Quantum Communication

A thesis submitted in fulfillment
of the requirements for the degree of
Master of Science

by

Ilan Kremer

supervised by
Prof. Noam Nisan

Computer Science Department
The Hebrew University of Jerusalem
Jerusalem, Israel

2 March 1995

First, I would like to thank my advisor Noam Nisan. During the two years I have been working with Noam, he has been a most costructive influence on me, teaching me how to think and write in a clear way. I would also like to thank my my fellow students, who during this time had to suffer hearing me lecturing my ideas; this goes especially to Amnon Tashma who hasn't recovered yet. My parents deserve a special thank, not only for getting me up to this point, but also for the help they have given me with the difficult task of writing this thesis in English. Finally I would like to thank my wife Ruthie for working around my short "mental going to work" periods, unlike the others she will have to continue living with me.

Contents

1 General Introduction

1.1 The modified Church-Turing thesis

The concept of "computation" is as old as the history of science itself. Philosophers, mathematicians and physicists were concerned with the issue of "computation" and its relation to the real word. With the construction of computers during the past 50 years this issue gained in importance considerably. A land-mark development in this field was the work of Allan Turing who defined the so called "Turing Machine" which is not a machine but the basic computational model in Computers Science [16]. A subsequent development was the formulation of the "Church-Turing thesis" [5] which established the general application of the "Turing Machine". This thesis claims that any "computational process" can be simulated by a Turing machine. It should be remarked that the concept of "computational process" in this context is a logical process.

With the rapid development of Computer Science a modified version of this thesis appeared. The "modified Church-Turing thesis" claims that any "reasonable computational process" can be efficiently simulated by a probabilistic Turing machine. The main differences in the modified thesis are:

- The simulation was required to be efficient (polynomial slowdown).
- The computational model "Turing Machine" was extended and a "Probabilistic Turing Machine" was defined.
- By the term "reasonable computational process" we mean also physical computation process i.e. analog computation. The initial state of a physical system codes the input of the computation while the final state codes the output. The term "reasonable" means that this process can be realized, for example only limited accuracy is allowed.

This thesis is not a mathematical statement therefore there exists no rigorous proof for it. Nevertheless scientists used to consider this thesis almost as an axiom. However, in recent years with the definition of quantum computation models (which will be discussed in section 1.2) doubts concerning the validity of this thesis developed.

In this context it should be remarked that there are two major questions connected with the relation of physics and computer science.

- The computational difficulties in the prediction of the behavior of physical systems.
- The availability of more power computers based on physical principles different from those on which present computers are based.

Most studies including this one whose subject is the second subject deal with discrete physical systems. The reason for it is the difficulty involved in the construction of continuous computational models in the real case when outside noise is present and the accuracy attainable in measurements is limited.

1.2 Quantum computation

There has been a rapid technological development in the field of computers during the past 50 years, which made it possible to solve many computational problems in an efficient way. There remain however, many important problems for which still no efficient solutions are available today. Assuming that the evolution continues at the current rate, it is to be expected that some of these problems will resist an efficient solution without the application of entirely new approaches. This fact led scientists to seek for non-conventional computational models. One of the most interesting directions in this field is quantum computation. Since the discovery of quantum mechanics, it has been found that the the laws of probability in quantum mechanics are inherently different from that encountered in the conventional probabilistic models. It was believed that by applying quantum computation models many until then intractable problem may become solvable. At the same time electronic components of the computers became smaller and smaller to the extent that these components themselves become subjected to the laws of quantum mechanics. As a result the electronics used in computers is quantum electronics. This fact led to the conviction that computers based on quantum principles belong to the real world, and we should alter our way of thinking and adapt it to the rules of quantum physics.

Feymann [7] was the first to ask what effect the behavior of quantum systems would have on computation. He claimed that it is intrinsically expensive to simulate quantum systems using classical computational methods, and that by using "quantum computers" this problem could be overcome. Subsequently Deutsch[6] formulated a quantum computational model, QTM (Quantum Turing Machine), and wondered whether the quantum computer has capabilities beyond that of the classical Turing machine. This question has attracted great interest (see [3][18][14]). The most remarkable result in this field was obtained by Peter Shor [13] who has shown that using Deutsch's model one can solve the problem of factorization, i.e. finding the prime factors of a natural number. This problem has a long history in the field of mathematics, it is widely believed that there exists no efficient conventional algorithm for this problem. Many cryptographic methods are based on this belief. Shor found a fast quantum algorithm for the above problem, but at present the question whether Shor's result refutes the thesis of Church-Turing is still open because:

- The proof that there does not exist an efficient algorithm for the factorization problem based on Turing machine is still outstanding
- It has not been shown that QTM can be realized in a concrete form.

Together with an interest in the above questions, there arose simultaneously an interest in quantum cryptography. Cryptography is a field in computer science which deals with the problem of safe transfer of information in a coded form. It has undergone rapid development in the past 20 years. As mentioned before most methods in this field are based on computational difficulties in various mathematical problems (e.g.- factorization problem). Although the existence of these difficulties is widely accepted, it has not yet been proven. These methods thereby lack a strong mathematical foundation.

The special field of quantum cryptography deals with the transfer of coded information using quantum principles. The proof of the safety of these methods is based on quantum mechanics; it therefore has a more solid theoretical foundation. Because the quantum devices involved are relatively simple, several models based on the transfer of quantum coded information have already been realized (see [2][4][8]).

1.3 Quantum communication

The quantum communication model is based on the communication model of Yao which was presented in his paper [17]. This model (the classical one) deals with the issue of communication by considering a situation in which two players A, B wish to evaluate a function $f(x, y)$. The input x is known only to A, and y is known only to B. In order to compute the function they have to communicate using some protocol. The resource in which the model is interested is the amount of communication needed for this purpose. In this context we have to mention Shanon's information theory, which also deals with the issue of transferring information and compare between the two models. Roughly speaking the main difference between this model and the well known Information theory of Shanon is that information theory deals with the question of *how* to send messages (how to overcome problems of noise, faulty links, etc.). The communication model on the other hand is concerned with the problem of *what* to send (i.e. design protocols). The motive in constructing this model was the wish to analyze computational models. This model proved to be successful in the area of computational complexity and many results were obtained by considering this model. Moreover extensive research whose main subject was communication was conducted in the field of computer science. The reason for this is the importance of the abstract notions *communication* and *information* in computers.

The quantum communication model deals with the information transferred in a quantum system. The model considers a quantum system divided into 3 parts A, B, C where A, B are the parts which communicate via C. Similarly to the classical model we deal with a situation in which some input x is coded in A and y in B, we are interested in the amount of information/communication needed to be transferred by a quantum time evolution process until the value $f(x, y)$ can be determined. Yao in his paper [18] which deals with quantum computation was the first (and in fact the only) researcher to mention quantum communication. He proved a quantum communication lower-bound of $loglog(n)$ for the *majority* function (a result which is generalized in this paper) and showed how to obtain a lower bound for quantum computational models using the above result. My primary motive in examining such a model was to compare quantum computational models with the classical models (TM, probabilistic TM...). More specifically I wanted to investigate whether the well known lower bounds for classical computation which are derived from communication hold true for quantum models of computation. Another motivation is an interest in communication for itself. As technology gets more and more sophisticated we have to start looking at quantum devices as a means for transferring information (see [2][4][8]).

In this thesis a complete and formal definition of the quantum communication model is given. Afterwards we show how to simulate probabilistic protocols by using quantum communication. This includes:

- Showing how to amplify the probability of success
- Showing how to simulate a probabilistic protocol using a quantum protocol.

We then continue by comparing quantum communication to probabilistic communication. We show that some basic lower bounds for probabilistic communication also hold for quantum communication.

- There exists at most a maximal exponential gap between deterministic communication and quantum communication (this is a generalization of the *majority* result which appears in Yao's paper)
- Most boolean functions $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ need $\Omega(n)$ quantum communication (this is the trivial upper bound).
- We present Yao's lower bound of $\Omega(n)$ for the inner-product mod 2 function. This gives an explicit function which needs $\Omega(n)$ quantum communication (this result was not previously published and has been e-mailed to me by Yao)

Still the question whether the probabilistic lower bound for the $DISJOITNESS$ function holds for the quantum case is left open. Moreover the more general question **whether quantum communication is more powerful than probabilistic communication?** is left open.

In section 6 we concentrate on the relation between one-round quantum communication and one-way probabilistic communication (the issue of one-round randomized communication is discussed in [11]). We follow the technique of complete problems:

- We define a complete problem for the class of functions whose one-way randomized communication is *polylog*.
- We define a complete problem for the class of functions whose one-way quantum communication is *polylog*, we discuss the relation between this problem and the complete problem for the randomized case.
- We define a second complete problem for the quantum case, and give 1-round randomized protocol for a special case of this problem.

The question of **whether 1-round quantum communication more powerful than 1-round probabilistic communication** is left open.

2 Basics

2.1 Quantum mechanics

2.1.1 Background

In this section we shall try to give a brief description of the the way Quantum Mechanics Theory views a "simple" system. This section is not intended to "teach" Quantum Physics, for this purpose the reader should refer to [15]. In the first subsection we give the postulates on which the mathematical description of a quantum system is based. These postulates should be treated as axioms. They are the foundations of Quantum Mechanics, and they come from observing physical phenomena and help us to formalize the mathematical description. Because this section when it comes separately can sound a bit vague we shall give a mathematical description of a very simple system (which physics name 'spinor'), and demonstrate how these principles are implemented.

2.1.2 The postulates of quantum mechanics

When we describe a physical theory we have to give answers to several questions, thereby giving a basis for the mathematical description. The postulates provide us answers to the following questions:

- What is the description of a state of a quantum system at a given time?
- Given the state, how can we predict the results of the measurements of various physical quantities? (i.e what is the meaning of a state).
- How can the state of the system at an arbitrary time t be computed when the state at time t_0 is known?

Quantum Mechanics describes a system by specifying a vector space (Hilbert space). The following six postulates define a discrete system.

1. At a fixed time t_0, the state of a physical system is defined by specifying a normalized vector belonging to the state space $\mathcal{V}$. The standard notation for this vector is the Dirac notation $|\psi(t_0)>$.
2. Every measurable physical quantity $\mathcal{A}$ is described by a linear operator A acting in $\mathcal{V}$, this operator is called observable.
3. The only possible result of the measurement of a physical quantity $\mathcal{A}$ is one of the eigenvalues of the corresponding observable A.
4. When the physical quantity $\mathcal{A}$ is measured on a system in state $|\psi(t_0)>$, the probability of obtaining the eigenvalue a_n of the observable A is:

$$Pr(a_n) = |<u_n|\psi(t_0)>|^2$$

Where $|u_n>$ is the normalized eigenvector of A, associated with the eigenvalue a_n. $<u_n|\psi(t_0)>$ denotes the inner product of the vectors $|u_n>, |\psi(t_0)>$.
The above formula is true when this eigenvalue is non-degenerate. The formula for the general case (degenerate eigenvalue) is a simple generalization of the previous one.

$$Pr(a_n) = \sum_{i=1}^{g_n} | <u_n^i|\psi(t_0)> |^2$$

Where g_n is the degree of degeneration, and $\{|u_n^i>\}$ is an orthonormal basis of the eigensubspace associated with the eigenvalue a_n.

5. After performing a measurement on the system, and getting a result a_n, the system is no longer in its previous state but in a state corresponding to the eigenvalue a_n

6. The time evolution of the state vector $|\psi(t_0)>$ is governed by the Schroedinger equation:

$$ih\frac{d}{dt}|\psi(t)>= H(t)|\psi(t)>$$

Where $H(t)$ is the observable associated with the total energy of the system. If this observable is independent of time, and time is discrete this equation can be written as:

$$|\psi(t=1)>= U|\psi(t=0)>$$

Where U is an unitary transformation in $\mathcal{V}$.

2.1.3 Spinor: an example of a 2 dimenssional system

In this section shall describe a simple system in the framework of quantum mechanics.

Suppose there is a system that can be described by a two dimension space, which has an orthonormal basis :

$$|e_0>= \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |e_1>= \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

This system is usually called spinor. We associate with these two vectors two states of the system S_0, S_1, which we name <u>basic states</u>. We associate with the state of the system at time t a vector of the form:

$$|\psi(t)> = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \qquad where\ |\alpha|^2 + |\beta|^2 = 1$$

Suppose the state at time t_0 is :

$$|\psi(t=0)> = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

The observable we have is :

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

which has two eigen values $a_0 = 0, \ a_1 = 2$. with the corresponding eigen vectors:

$$|v_0> = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad |v_1> = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

If we measure in time t_0 the physical quantity $\mathcal{A}$ that corresponds to A, we will get result a_0 with probability:

$$| <v_0|\psi(t=0)> |^2 = |\frac{1}{\sqrt{2}}|^2 = 0.5$$

The state will become $|v_0>$. If we do not measure and wait one second, the time transformation is given by:

$$U = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

The state will become:

$$|\psi(t=1)> = U|\psi(t=0)> = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

Then if we measure A we will get result a_0 with probability :

$$|(|v_0>, |\psi(t=1)>)|^2 = 0$$

2.2 Computational complexity

Computational complexity is a mathematical branch of Computer Science which deals with the analysis of difficulties one encounters in the calculation of functions. The purpose of the present section is to discuss various approaches used in solving problems in Computational Complexity. These differ in several aspects from those used in other areas of mathematics. In the following, we shall describe some of their essential features. For a more detailed discussion in this subject see [12].

In order to investigate the difficulties of computing some function f we have to specify some *computational model* which is a mathematical model (e.g. Turing machines, boolean circuits). Having defined a particular model, 'Algorithm' is the method of computing a desired function in this model. We have to specify in the model the various resources required in the computational process (the number of steps, memory requirements, etc..). These resources determine the various measures of the "cost of the Algorithm" which presents the central issue in Computational Complexity. The "cost of the Algorithm" is normally calculated for the worst case situation ("worst input"). In most cases we deal with boolean functions $f : \{0,1\}^n \to \{0,1\}$. We assume that f is defined for every n. We are interested in the assymptotical behaviour of the cost of the Algorithm when $n \to \infty$. The cost of the best possible Algorithm for a function ("cheapest" Algorithm) defines the complexity of the function.

For every "Computational Model" a probabilistic variant can also be defined. This can be done in two ways which in may cases are equivalent:

1. Define a "random Algorithm" as an Algorithm which uses "random steps".

2. Define a "random Algorithm" as constructing a probability distribution over deterministic Algorithms.

The cost of the "randomizes Algorithm" or the reliability of the "randomized Algorithm" in computing the function are measured by averaging over the random steps, or alternatively over the distribution of the Algorithms. It should be remarked that these results refer in most cases to the worst case input. (Note that we do not make any assumptions regarding a specific distribution over the inputs).

Complexity theory categorizes functions into classes according to their complexity. Its aim is to find relations among the different complexity classes. An important method to determine relations between two classes $\mathcal{A}$ and $\mathcal{B}$ is to find a complete function f (complete problem), which is a function $f \in \mathcal{A}$, and to which we can reduce every function f' belonging to $\mathcal{A}$ (by reducing f' to f we mean tranforming a prblem of computing $f'(x)$ to a problem of computing $f(y)$). Thus by proving $f \in \mathcal{B}$, it implies that $\mathcal{A} \subseteq \mathcal{B}$.

2.3 Communication complexity

In this section we will introduce Yao's model of communication complexity, which was introduced in his paper [18]. In subsection 1 we give definitions for the deterministic case. In subsection 2 we present the probabilistic variant. In subsection 3 we list known results in the classical(non-quantum) model which are related to the "quantum results". For further reading the reader should refer to [9][10])

2.3.1 Model definition

Yao's Communication model is a computational model in which we are only interested in the resource of communication. Thus, the notion of *Algorithm* is replaced by the notion of *Protocol*. The communication model deals with the following situation:
Two players *Alice* and *Bob* wish to compute the value of a function $f : X \times Y \to Z$, where X, Y, and Z are arbitrary sets, on a given pair of inputs $x \in X$ and $y \in Y$. The difficulty in the computation is due to the fact that only *Alice* knows x, and only *Bob* knows y. *Alice* and *Bob* are allowed to *communicate* by sending messages (bits, or strings of bits) between themselves according to some *protocol* P. The *cost* of P on a given input (x, y) is the number

of bits sent by *Alice* and *Bob* together for the input. The overall cost of P is defined as the worst case cost over all inputs. The *communication complexity* of a function f is defined asthe minimum cost over all protocols that compute f. In the following we formalize the notions described above.

A protocol P over domain $X \times Y$ with range Z is a set of functions $\{f_i\} : \{0,1\}^* \to \{0,1\}^*$.

- A function f_i with an odd index: $i \in \{1, 3, 5..\}$ represents the i-th message which is sent by *Alice*:

$$f_i(x, previous\ messages) = i'th\ message$$

- A function f_i with an even index: $i \in \{2, 4, 6..\}$ represents the i-th message which is sent by *Bob*

$$f_i(y, previous\ messages) = i'th\ message$$

The outcome of applying protcol P on an input x, y can be computed in a recursive form.

1. message $M_1 = f_1(x)$
2. message $M_2 = f_2(y, f_1(x))$
3. message $M_3 = f_3(x, f_1(x), f_2(y, f_1(x)))$

   $\vdots$

For every pair (x, y) this sequence must be finite. The last message which is sent is M_l. l is known to each of the players by his input and by the messages sent. The output of the protocol named $P(x, y)$ is defined as the last message M_l. The cost of P on input x, y is the total length of the messages sent on this input. The cost of P is $max_{x,y}(cost\ of\ P\ on\ input\ x, y)$ P is said to compute $f : X \times Y \to Z$ if $\forall x, y\ \ P(x, y) = f(x, y)$

We now define the deterministic communication complexity for a function $f : X \times Y \to Z$.

Definition 1 *$D(f)$ (deteministic communication complexity) is defined as the minimum cost over all deteministic protocols that compute f*

In most cases we shall deal with the case X, Y being $\{0,1\}^n$ ($n - bits$ strings) and $Z, \{0,1\}$. In the following, some protocols are shown:

Example 1 A *and* B *are given each a n-bits string and they wish to know if they have the same strings, the function f is therefore defined as:*

$$EQ(x, y) = \begin{cases} 0 & if\ x = y \\ 1 & otherwise \end{cases}$$

For this problem (and for all other problems) there exists a trivial protocol in which A *sends its input x to* B. B *computes $EQ(x, y)$ and tells* A *the answer. This protocol requires n+1 bits . One can prove that for this problem there doesn't exists a cheaper protocol. Thus,* $D(EQ) = \Theta(n)$

There are cases where there are cheaper protocol:

Example 2 A *and* B *are given n-bits strings* x, y, *their task is to determine wether the number of 1's in* x, y *is even or odd.*

$$PARITY(x,y) = \begin{cases} 0 & \textit{if (num of 1's in } x \textit{ + num of 1's in } y\textit{) is even} \\ 1 & \textit{otherwise} \end{cases}$$

Here it is easy to see that they need to exchange only two bits of information and that they don't need to use the trivial protocol (to send the input). A *sends to* B $((num\ of\ 1's\ in\ x)\ mod\ 2)$, B *computes* $(num\ of\ 1's\ in\ x\ +\ num\ of\ 1's\ in\ y) mod 2)$, *and tells* A *the answer. In this case* $D(PARITY) = \Theta(1)$

2.3.2 The probabilistic variant

As for almost any other computational model when we add the power of randomness we get interesting results . But what do we mean when we say a *random protocol*?
Alice and *Bob* have an access to some private random strings r_A, r_B so they can choose their messages at random. In the following we formalize the notions described above.

A random protocol P over domain $X \times Y$ with range Z is a set of functions $\{f_i\}$: $\{0,1\}^* \rightarrow \{0,1\}^*$.

- A function f_i with an odd index: $i \in \{1, 3, 5..\}$ represents the i-th message which is sent by *Alice*:

$$f_i(x, r_A, previous\ messages) = i - th\ message$$

- A function f_i with an even index: $i \in \{2, 4, 6..\}$ represents the i-th message which is sent by *Bob*

$$f_i(y, r_B, previous\ messages) = i - th\ message$$

r_A, r_B are random strings drawn from the uniform distribution over $\{0,1\}^k$ for some arbitrary k. The outcome of applying protcol P on an input x, y can be computed in a recursive form.

1. message $M_1 = f_1(x, r_A)$
2. message $M_2 = f_2(y, r_B f_1(x))$
3. message $M_3 = f_3(x, r_A, f_1(x), f_2(y, f_1(x)))$

$\vdots$

For every (x, y) this sequence must be finite. The last message which is sent is M_l. l is known to each of the players by his input, by his private random string and by the messages sent. The cost of P on input x, y is the <u>expected</u> total length of the messages sent on input x, y. The cost of P is $max_{x,y}(cost\ of\ P\ on\ input\ x, y)$ We say that a *random protocol P*

- Computes f with $\epsilon - error$ if $pr_{r_A,r_B}((M_l \neq f(x,y)) \leq \epsilon)$
- Computes f with $zero - error$ if $pr_{r_A,r_B}(M_l \neq f(x,y)) = 0)$

Definition 2
- *$R^{\epsilon}(f)$ is defined as the minimum cost over all random protocols that compute f with $\epsilon - error$*
- *$R^0(f)$ is defined as the minimum cost over all random protocols that compute f with $zero - error$*

In the following $R(f)$ denotes $R^{\frac{1}{3}}(f)$. Next we give a probabilistic protocol for EQ function that costs only O(log(n)) bits.

Example 3

- *Alice chooses prime number p at random from the first n^2 primes and sends:*

$$(x \ mod \ p), p$$

- *Bob checks whether*

$$y \ mod \ p = x \ mod \ p.$$

and tells Alice the answer.

The probability of mistake is probability that

$$x \neq y \ \ and \ \ (x \ mod \ p \ ) = (y \ mod \ p)$$

This happens iff $x \neq y$ and p is among the divisors of $x - y$. There are at most $log(x-y) < n$ divisors. so the probability of mistake is less than n^{-1}. For the claculation of the number of bits sents we recall a known fact from Number Theory which states that the first n^2 primes are in the range $[1 \ldots n^3]$, thus the length of the Alice's message is $O(log(n))$ bits. We conclude that $R(EQ) = O(log(n))$

2.3.3 Some basic results

We list here some known results in Communication complexity, and their intuitve meaning. One of the reasons for doing so is that in section 3 the quantum versions of these results appear. Moreover the proofs in the quantum case are many times modifications of the proofs for these results. For formal proofs of these result the reader may refer to [10].

1. $\forall \ f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\} \quad D(f) \leq n$ - For every function there is always the trivial protcol which is for A to send its input to B, this costs linear communication.

2. $\forall\, f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ $R(f) \leq D(f)$- Random communication is "stronger" than deterministic communication.

3. $\forall\, f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ $R(f) \geq log(D(f))$- There is an exponential gap at the most between random communication and deterministic communication.

4. For most of the functions $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ $\quad R(f) \geq \Omega(n)$ - Even though random commnication is "stronger" than deterministic communication, it is not so "strong". Most of the functions need linear communication which is the cost of the trivial protcol.

5. Define the function :

$$IP_2(x,y) = \sum_i x_i \cdot y_i (mod 2) \quad x, y \in \{0,1\}^n.$$

   For this function we know:

$$R(IP_2) \geq \Omega(n)$$

   Even though we know that most of the function need linear communication, it is not clear that we can give an explicit "expensive" function. This result tells us that a certain function is "expensive".

3 Definition of the model

3.1 Overview

In this section we define the model of quantum communication. We shall use notions from linear algebra, for example: tensor product. A description of the terms and the mathematical notations used appears in the appendix.

The model of quantum communication deals with the complexity of the time evolution of many particle systems (spinors). It is based on the analysis of the transfer of information within the system. For this purpose we divide the system into three parts: $\mathcal{A}, \mathcal{B}$ and $\mathcal{C}$. $\mathcal{A}$ and $\mathcal{B}$ are entities which communicate with each other, they correspond to Alice and Bob in Yao's model. Communication is transferred via $\mathcal{C}$. We regard this system as a model for computing boolean functions $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$. The initial state of the system codes the input of the function: $x \in \{0,1\}^n$ is coded in $\mathcal{A}$ and $y \in \{0,1\}^n$ is coded in $\mathcal{B}$. The final state codes the value of $f(x,y)$. The coding is done by the state of one of the particles (spinors). In terms of quantum mechanics, we obtain a random variable $\in \{0,1\}$, by measuring the state of the particle. The value of the random variable will be $f(x,y)$ with high probability. The process of computation (called the protocol) consists of a series of unitary transformations (as in the quantum computation model QTM). Each unitary transformation can change either the state of the pair of components $\mathcal{A},\mathcal{C}$ or that of $\mathcal{B},\mathcal{C}$. It is implied that there is no direct interaction between $\mathcal{A}$ and $\mathcal{B}$. The amount of communication which is transferred is equal to the number of unitary transformations times the number of particles in $\mathcal{C}$. This quantity is called the cost of the protocol. We shall say that a protocol P computes the function f, if for every pair of values x, y the protocol changes the state of the system starting with a certain initial states coding x, y and ending in a final state coding $f(x, y)$. The quantum communication complexity of a function f is then defined as the minimal cost required for a protocol which computes f. In the next section a more rigorous and formal definition of these terms will be given.

3.2 States

We give here the mathematical description of the states of the system as vectors in some vector space. Throughout this section we use the Dirac notation $|v>$ for vectors.

Definition 3 *Let H be the vector space C^{2^m}. A* basic state *is defined as a unit vector $\vec{e}_i$ in this space.*

We identify the numbers in the interval $[0 \cdots 2^m - 1]$ with the binary strings $\{0,1\}^m$, according to their binary representation.

Notation 1 *Let H be the vector space C^{2^m}, and let x be a binary string of length m. We denote by $|x>$ the basic state defined as :*

$$|x>_i = \begin{cases} 1 & i = x \\ 0 & otherwise \end{cases}$$

If the length of the string x is k where $k \le m$, we denote by $|x>$ the vector $|0^{m-k} \circ x>$ where $\circ$ means con-cation, and 0^{m-k} means the string of $m - k$ zeroes.

Definition 4 *Let H be the vector C^{2^m}. A* general state *is defined as a vector in this space, $\vec{v}$, where $|\,\vec{v}\,|_2 = 1$ (Its L_2 norm equals 1).*

The system is to be divided into three parts $\mathcal{A},\mathcal{B}$ and $\mathcal{C}$, Where $\mathcal{A}$ and $\mathcal{B}$ consist of m particles (spinors) and $\mathcal{C}$ of c particles. Each part can be described by a vector space. The mathematical description of the system is then the tensor product of these spaces.(see appendix)

Notation 2 *Denote by $H_{m,c}$ the vector space $H_A \otimes H_C \otimes H_B$, where H_A, H_B are the vector spaces C^{2^m}, and H_C is C^{2^c}.*

A special case often treated in this paper is $c = 1$.

Notation 3 *Let x, y, z be strings of length m, c, m respectively. Denote by $|x, y, z>$ a basic state in $H_{m,c}$. The state will be the tensor product of three basic states from H_A, H_C, H_B:*

$$|x, y, z> = |x> \otimes |y> \otimes |z>$$

$H_{m,c}$ is itself a space of dimension 2^{2m+c}. In terms of the notation 1 $|x,y,z\rangle$ is $|x\circ y\circ z\rangle$. To clarify this point let us consider the following example:

Example 4 *Let x be the string 01 y be 1, and z the string 10. Let H_A, H_C, H_B be spaces of dimensions 4, 2, 4 respectively. The unit vectors $|01\rangle, |1\rangle, |10\rangle$ from H_A, H_C, H_B are :*

$$|x\rangle = |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad |y\rangle = |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad |z\rangle = |10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

In $H = H_A \otimes H_C \otimes H_C$ $|01,1,10\rangle$ is a vector of dimension $4\times 4\times 2 = 32$.

$$|01,1,10\rangle = |01\rangle \otimes |1\rangle \otimes |10\rangle$$

In terms of H , $|01,1,10\rangle$ is actually $|01110\rangle$ implying that it is the unit vector $\vec{e}_{14=01110}$ According to our first notation we can drop the leading zeroes in $|01110\rangle$ and write it in the form $|1110\rangle$.

We have now the mathematical description of a system built from 3 parts as a tensor product of three vector spaces, and the mathematical description of its basic states as unit vectors. The general states of this system are vectors whose L_2 norm equals 1. It follows from the definition of tensor product the set of vectors $\{|x,y,z\rangle \mid x,z \in \{0,1\}^m\ y \in \{0,1\}^c\}$ is an orthonormal basis for $H_{m.c}$ and we can deduce:

Corollary 1 *$\forall$ general state $\vec{v} \in H_{m,c}$ there exist a unique set of coefficients $\{a_{x,y,z}\}$ s.t:*

$$\vec{v} = \sum_{x,y,z} a_{x,y,z}|x,y,z\rangle \quad where \quad \sum_{x,y,z} |a_{x,y,z}|^2 = 1$$

3.3 Measurements

Next we will define the notion of measurement.

Definition 5 *Let H be a vector space of dimension 2^N, $S \subseteq \{0,1\}^N$, and let $\vec{v}$ be a general state in H*
A measurement *R_S of $\vec{v}$ is defined as random variable with values {0,1} s.t :*

1 $pr(R_s(\vec{v}) = 1) = |proj_{H'}\vec{v}\,|^2$
where $H' = span\{|i\rangle \mid i \in S\}$

2 $pr(R_s(\vec{v}) = 0) = |proj_{H''}\vec{v}\,|^2$
where $H' = span\{|i\rangle \mid i \notin S\}$

It follows from the fact that $|\,\vec{v}\,|^2 = 1$ that this definition is valid, i.e. :

1. $|proj_{H'}\vec{v}\,|^2, |proj_{H''}\vec{v}\,|^2 \geq 0$

2. $|proj_{H'}\vec{v}\,|^2 + |proj_{H''}\vec{v}\,|^2 = 1$

The probability of getting the value 1 in our measurement is the square norm of the projection of $\vec{v}$ on some subspace. It can be written in the following way.

Proposition 1

$$pr(R_S(\vec{v}) = 1) = \sum_{i \in S} |\vec{v}_i|^2$$

Definition 6 *Let i be a $(2m + c)$ bit string which can be written as a concatenation of 3 strings of length m, c, m :*

$$i = x \circ y \circ z$$

A measurement R_s in $H_{m,c}$ is said to measure a bit b in *H_A if*

$$S \quad = \quad \{i = x \circ y \circ z \mid b = 1, b \text{ is a bit in } x\}$$

Similarly, we can define the measurements of bits in H_C or H_B

Example 5 *Let us take H as H_m. There is only one measurement of bit in H_C which is R_S where :*

$$S = \{i | i = x \circ 1 \circ z\}$$

We shall call this measurement as R_{com}, because it measures a bit in the communication space. If H is the general case i.e. $H_{m,c}$, we define R_{com} as the measurement of the first bit in H_c. It implies :

$$S \quad = \quad \{i = x \circ y \circ z \mid \text{the first bit in } y \text{ is } 1\}$$

3.4 Quantum protocol

Definition 7 *Let U be a unitary transformation on $H_{m,c}$. U is said to* act on *$H_A \otimes H_C$ if there exists a set of $2^{2(m+c)}$ coefficients $a_{x,y,x',y'}$ s.t for every $x, z \in \{0,1\}^m$, $y \in \{0,1\}^c$*

$$U|x, y, z\rangle = \sum_{x',y'} a_{x,y,x',y'} |x', y', z\rangle$$

Intuitively this means that U does not alter H_B because the z part remains unchanged. These transformation can transfer information from H_A to H_C and/or transfer information from H_C to H_A. Similarly we define a transformation that acts on $H_C \otimes H_B$.

Example 6 • *The transformation of moving a bit b from H_C to H_B is an example of an unitary transformation that acts on $H_C \otimes H_B$, it is defined in the following way:*

$$\forall x_A, x_B, b \quad U|x_A,\ b,\ x_B\rangle = |x_A,\ 0,\ b \circ x_B\rangle$$

We will denote this transformation as $U_{move\ (C,B)}$.
(Recall that according to our notation $|x> = |0^{m-k} \circ x>$ *)*

- *The transformation of copying a bit* b *from* H_C *to* H_B *is also an example of an unitary transformation which acts on* $H_C \otimes H_B$*, it is defined in the following way :*

$$\forall x_A, x_B, b \quad U|x_A,\ b,\ x_B> = |x_A,\ b,\ b \circ x_B>$$

We will denote this transformation as $U_{copy\ (C,B)}$.

Definition 8 *A* quantum protocol P *is a sequence of unitary transformations on* $H_{m,c}$ *:*

$$U_A^1, U_B^2, U_A^3, U_B^4, ..U_A^{l-1}, U_B^l$$

$\{U_A^i\}$ *are unitary transformations which act on the space* $H_A \otimes H_C$, $\{U_B^i\}$ *are unitary transformations that act on* $H_C \otimes H_B$ *. The cost of a protocol* P*,* $C(P)$ *is defined as :*

$$l \times log(dim(H_C)) \quad = \quad l \times c$$

Notation 4 1. *Denote by* $\vec{P}^{x,y}$ *the vector :* $U_B^l \cdot U_A^{l-1} \cdot U_B^{l-2} \cdot U_A^3 \cdots U_B^2 \cdot U_A^1|x, 0, y>$

2. *Denote by* $P(x,y)$ *the probability of getting* 1 *when measuring the first bit in* H_c : $pr(R_{com}(\vec{P}^{x,y}) = 1)$

Next we define the meaning of the statement that a protocol "computes" the function f.

Definition 9 *The protocol* P *is said to* compute f with accuracy ϵ *if:*

$$\forall x, y \quad pr(R_{com}(\vec{P}^{x,y}) = f(x,y)) \geq 1 - \epsilon$$

For a protocol P which computes f with accuracy ϵ we have

1. $f(x,y) = 1 \Rightarrow P(x,y) \geq 1 - \epsilon$
2. $f(x,y) = 0 \Rightarrow P(x,y) \leq \epsilon$

We are able now to define the notion of quantum communication complexity.

Definition 10 *The* quantum communication complexity $Q^\epsilon(f)$ *is defined as the minimum cost over all quantum protocol that compute* f *with accuracy* ϵ*. In the following,* $Q^{1/3}(f)$ *will simply be denoted as* $Q(f)$

4 Discussion

4.1 Technical lemmas

In this section we shall deal mainly with technical results. These results will show that it is possible to perform simple operations using the quantum model which are comparable to those performed when using the classical model. Below is a series of lemmas each following from the preceding one. The last lemma shows that we are able to amplify probabilistic results by repeated application of a quantum protocol. The essence of the proof is to demonstrate how a repetition of the quantum protocol can be performed. For this purpose we shall show how the protocol can preserve the initial inputs. The proofs are given in the appendix, as they involve numerous details.

Definition 11 *Let H be a vector space of dimension 2^{n+k} , U be a unitary transformation acting on H . U is said to preserve $\{0,1\}^n$ if there exists a set of coefficients $\{a_{x,i}\}$ s.t for every $x \in \{0,1\}^n$:*

$$U|x> = \sum_{i \in \{0,1\}^k} a_{x,i} |i \circ x>$$

The next lemma shows that we can find a wide range of transformations, which preserve $\{0,1\}^n$.

Lemma 1 *Let X be $\{0,1\}^n$, Y be $\{0,1\}^k$.*
For every two sets sets $\{a_{x,i}\}\{b_{x,i}\}$ satisfying that for every $x \in \{0,1\}^n$:

$$\sum_{i \in \{0,1\}^k} |a_{x,i}|^2 = \sum_{i \in \{0,1\}^k} |b_{x,i}|^2 = 1$$

there exists an unitary transformation U acting on a space of 2^{n+k} dimensions s.t for every $x \in \{0,1\}^n$:

$$U \sum_{i \in \{0,1\}^k} a_i^x |i \circ x> = \sum_{i \in \{0,1\}^k} b_i^x |i \circ x>$$

It follows that we can simulate any boolean function $f : \{0,1\}^n \to \{0,1\}$ using transformations which preserve $\{0,1\}^n$

Corollary 2 *Let H be a 2^{n+1} dimensional space. For every function $f : \{0,1\}^n \to \{0,1\}$ there exists an unitary transformation U in H space, that "simulates" f and preserves X i.e. :*

$$\forall x \in \{0,1\}^n \;\; U|x> = |f(x) \circ x>$$

Proof: The conditions :

$$\forall x \in X \;\; U|x> = |f(x) \circ x>$$

are a special case of the previous lemma.

$$a_{x,i} = \begin{cases} 1 & i \text{ is a string of zeroes} \\ 0 & otherwise \end{cases} \qquad b^{x,i} = \begin{cases} 1 & i \text{ is } f(x) \\ 0 & otherwise \end{cases}$$

□

Notation 5 *Let us denote by U_f an unitary transformation that simulates f*

Definition 12 *Let P be a protocol acting on $H_{m,c}$ for some function $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$. P will be called a protocol which* preserves the input*, if the following holds:*

$$\forall x, y \in \{0,1\}^n \quad \vec{P}^{x,y} = \sum_{i,j,k} a_{i,j,k} |i \circ x, j, k \circ y>$$

Lemma 2 *For every protocol P in $H_{m,c}$ that computes a function $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ with accuracy ϵ there exists a protocol P' which acts on $H_{m+n,c}$ that computes f with accuracy ϵ , and preserves the input, moreover $C(P) = C(P')$.*

In the classical model, given a protocol which computes a function $f : X \times Y \rightarrow \{0,1\}$ with probability $1-\epsilon$, it is always possible to construct an alternative protocol which will repeat the original protocol three times and will return the "majority result". The probability of the success of the new protocol will be identical with the probability of getting at least two "heads" when flipping three times a coin with parameter $1-\epsilon$:

$$3(1-\epsilon)\epsilon^2 + (1-\epsilon)^3 > 1-\epsilon$$

The price of the new protocol is three times that of the original one, which implies:

$$\forall f, \epsilon \quad R^{(1-\epsilon)^3 + 3\epsilon(1-\epsilon)^2}(f) \leq 3 \cdot R^{1/2-\epsilon}(f)$$

By repeating c times it follows that:

$$\forall f, \epsilon \quad R^{e^{\Omega(-c\epsilon^2)}}(f) \leq c \cdot R^{1/2-\epsilon}(f)$$

The same technique can be applied also in the quantum case. The main point is to prove a repetition lemma, in the appendix we prove a 2-times repetition lemma:

Lemma 3 *Let f_1 and f_2 be boolean functions $\{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$. In addition let h be a boolean function $\{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ defined as $h(x,y) = g(f_1(x,y), f_1(x,y))$ for some function $g : \{0,1\}^2 \rightarrow \{0,1\}$. If there exist two quantum protocols P_1, P_2 which compute f_1, f_2 respectively with accuracy ϵ then there exists a third quantum protocol P with the following properties:*

- *P computes h with accuracy $\epsilon' = pr_{r_1,r_2}[g(r_1,r_2) \neq h(x,y)]$ where r_1, r_2 are two* <u>*independent*</u> *Bernoulli variables ($\in \{0,1\}$) with the property: $pr[r_1 = f_1(x,y)] = pr[r_2 = f_2(x,y)] = 1-\epsilon$*

- $C(P) = C(P_1) + C(P_2)$

By repeating the protocol c times with $f_1, ...f_c = f$ and $h = maj(f_1, ...f_c)$ it follows that:

Theorem 1

$$\forall f, \epsilon \quad Q^{e^{\Omega(-c\epsilon^2)}}(f) \leq c \cdot Q^{1/2-\epsilon}(f)$$

4.2 Quantum communication as compared to probabilistic communication

The purpose of this section is to prove that we can simulate a probabilistic protocol by using a quantum protocol. We start by simulating a deterministic protocol.

Lemma 4 $\forall f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\} \quad Q(f) \leq O(Dcc(f))$

Proof: We assume in the deterministic protocol that alternate players send single bits. This way of communication increases the cost by a factor of 2, at most. Suppose we have a deterministic protocol for f, of cost l.

- In stage 1 A sends a bit b_1 to B which is a function of x:

$$b_1 = f_1(x)$$

- In stage 2 B send a bit b_2 which is a function of y and b_1

$$b_2 = f_2(y, b_1)$$

- In the final stage l, B sends to A the result, $b_l = f(x, y)$, where

$$b_l = f_l(y, b_1, \ldots, b_{l-1})$$

The quantum protocol which simulate the deterministic one is defined in the following way:

- Define U_1^0 as $U_{copy(1,c)} \cdot U_{f_1}$. Note that:

$$\begin{aligned} U_1^0 |x, 0, y> &= U_{copy(1,c)} \cdot U_{f_1} |x,\ 0,\ y> = U_{copy(1,c)} |b_1 \circ x,\ 0,\ y> = \\ &= |b_1 \circ x,\ b_1,\ y> . \end{aligned}$$

- Define U_2^1 as $U_{copy(2,c)} U_{f_2} U_{move(c,2)}$. Note that:

$$\begin{aligned} U_{copy(2,c)} \cdot U_{f_2} \cdot U_{move(c,2)} |x,\ b_1,\ y> &= U_{copy(2,c)} \cdot U_{f_2} |x,\ 0,\ b_1 \circ y> \\ &= U_{copy(2,c)} |x,\ 0,\ b_2 \circ b_1 \circ y> = |x,\ b_2,\ b_2 \circ b_1 \circ y> \end{aligned}$$

- Define U_2^l as $U_{f_l} U_{move(c,2)}$

So P is:

$$P = U_{f_l} U_{move(c,2)}, \ldots, U_{copy(2,c)} U_{f_2} U_{move(c,2)}, U_{copy(1,c)} U_{f_1}$$

It follows that:

$$\vec{P}^{x,y} = |b_l \circ \ldots \circ b_1 \circ x,\ b_l, b_l \circ \ldots \circ b_1 \circ y> =$$
$$|b_{l-1} \circ \ldots \circ b_1 \circ x,\ f(x,y),\ b_l \circ \ldots s \circ b_1 \circ y> = \vec{v}_{final}$$

which implies that:

1. $f(x,y) = 1 \Rightarrow P(x,y) = 1$
2. $f(x,y) = 0 \Rightarrow P(x,y) = 0$

□

A similar proof can be given in the probabilistic case:

Theorem 2 $\forall f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\} \quad Q(f) \leq O(R(f))$

Proof: We assume in the deterministic protocol that alternate players send single bits. Suppose we have a deterministic protocol for f, of cost l. A has input x and a random string r_A drawn from the uniform distribution over $\{0,1\}^k$ for some k. Similarly B has y and r_B. Given x, r_A and y, r_B, the players apply a deterministic protocol.

- In stage 1 A sends a bit b_1 to B which is a function of x, r_A:

$$b_1 = f_1(x, r_A)$$

- In stage 2 B send a bit b_2 which is a function of y, r_B and b_1

$$b_2 = f_2(y, r_B, b_1)$$

- In the final stage l, B sends to A the result, b_l, where

$$b_l = f_l(y, r_B, b_1, \ldots, b_{l-1})$$

The probability of b_l being $f(x,y)$ is given by:

$$pr_{r_A, r_B}(b_l = f(x,y)) = 1/2^{2k}(number\ of\ strings\ r_A, r_B\ where\ b_l = f(x,y))$$

For the quantum protocol which simulates the probabilistic we define two transformation: Define U^1_{rand} as the unitary transformation such that

$$\forall\ x, b, y \quad U^1_{rand}|x, b, y> = \sum_{r_A \in \{0,1\}^k} (1/2^{k/2})\ |r_A \circ x, b, y>$$

Similarly we can define U^2_{rand}, these transformation are unitary transformations (see 1). Intuitively these transformation choose random strings from $\{0,1\}^k$. The quantum protocol is defined in the following way:

$$P = U_{f_l} U_{move(c,2)}, \ldots, U_{copy(2,c)} U_{f_2} U_{move(c,2)} U^2_{rand}, U_{copy(1,c)} U_{f_1} U^1_{rand}$$

It follows that:

$$\vec{P}^{x,y} = \sum_{r_A, r_B} (1/2^k) \, |b_{l-1} \circ \ldots \circ b_1 \circ r_A \circ x, \; b_l, \; b_l \circ \ldots s \circ b_1 \circ r_B \circ y> = \vec{v}_{final}$$

Which implies:

$$\begin{aligned} pr(R_{com}(\vec{P}^{x,y}) = f(x,y)) &= 1/2^{2k}(number\ of\ strings\ r_A, r_B\ where\ b_l = f(x,y)) \\ &\geq 1 - \epsilon \end{aligned}$$

Note that $C(P) = l$ □

5 Lower bounds

5.1 Overview

In section 4 it was shown that quantum communication is as cheap as the probabilistic communication. In this section we shall see that some lower bounds to the probabilistic communication also hold in the quantum case. In part 1 we shall prove two theorems. The first one will show that there exists at most an exponential gap between deterministic communication and quantum communication. The second will prove that most of that boolean functions $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ require linear quantum communication. These two theorems follow from the description of a quantum protocol by a set of matrices. Part 2 is based on a theorem by Yao. This theorem proves the existence of a linear lower bound for a specific function. In this section we shall deal only with protocols for H_m. The results can be generalized for general spaces $H_{m,c}$. All these results will follow from the analysis of quantum protocols which is presented in the next section.

5.2 Quantum protocol analysis

Lemma 5 *For every protocol P of cost l and for every input $x, y \in \{0,1\}^n$.*

$$\vec{P}^{x,y} = \sum_{j \in \{0,1\}^l} \vec{v}^j_A \otimes |b_j> \otimes \vec{v}^j_B$$

where $\vec{v}^j_A, \vec{v}^j_B$ are vectors from H_1, H_2 respectively (their L_2 norm not being necessarily 1), and b_j is the last bit in j, more over:

1. *The set* $\{\vec{v}_A^j\}$ *depends on the protocol* P *and the input* x
2. *The set* $\{\vec{v}_B^j\}$ *depends on the protocol* P *and the input* y

Proof: The proof is by induction on the length of the protocol :
l=0:

$$\vec{P}^{x,y} = |x,0,y> = |x> \otimes |0> \otimes |y>$$

induction step:
Let P be a protocol of cost l:

$$P = U_1^1, U_2^2, U_1^3, U_2^4, ..U_1^{l-1}, U_2^l$$

Assume by induction that:

$$U_1^{l-1} \cdots U_2^2 \cdot U_1^1 |x,0,y> = \sum_{j \in \{0,1\}^{l-1}} \vec{v}_A^j \otimes \vec{v}_c^j \otimes \vec{v}_B^j$$

where the set $\{\vec{v}_A^j\}$ can be computed using P and x, $\{\vec{v}_B^j\}$ can be computed using P and y. The next transformation is U_2^l, because U_2^l acts on $H_C \otimes H_B$ we can write:

$$U_2^l\ (\vec{v}_A^j \otimes \vec{v}_C^j \otimes \vec{v}_B^j) = \vec{v}_A^j \otimes \vec{w}^j$$

where $\vec{w}^j$ is some vector in $H_c \otimes H_2$, define:

$$\begin{array}{ll} \vec{w}_0^j = proj_{H_0} \vec{w}^j & where\ H_0 = span\{|0> \otimes |z> \ \mid\ |0> \in H_C,\ |z> \in H_B\} \\ \vec{w}_1^j = proj_{H_1} \vec{w}^j & where\ H_1 = span\{|1> \otimes |z> \ \mid\ |1> \in H_C,\ |z> \in H_B\} \end{array}$$

and it t follows that $\vec{w}^j = \vec{w}_0^j + \vec{w}_1^j$ and that $\vec{w}_0^j$ can be written as $|0> \otimes \vec{v}_0^j$ for some vector $\vec{v}_0^j$ in H_2. Also $\vec{w}_0^j$ can be written as $|1> \otimes \vec{v}_1^j$ for some vector $\vec{v}_1^j$ in H_2, so we get:

$$\begin{aligned} U_2^l \cdot U_1^{l-1} \cdots U_2^2 \cdot U_1^1 |x,0,y> &= U_2^l \sum_{j \in \{0,1\}^{l-1}} \vec{v}_A^j \otimes \vec{v}_c^j \otimes \vec{v}_B^j = \sum_{j \in \{0,1\}^{l-1}} \vec{v}_A^j \otimes (\vec{w}_0^j + \vec{w}_1^j) = \\ \sum_{j \in \{0,1\}^{l-1}} \vec{v}_A^j \otimes (|0> \otimes \vec{v}_0^j + |1> \otimes \vec{v}_1^j) &= \sum_{j \in \{0,1\}^{l}} \vec{v}_A^j \otimes \vec{v}_c^j \otimes \vec{v}_B^j \end{aligned}$$

The set $\{\vec{v}_A^i\}_{i \in \{0,1\}^l}$ didn't change through the induction step, it just doubled it self by repeating each vector twice, so if the previous set could be computed from P and x so is this set. The set $\{\vec{v}_B^i\}_{i \in \{0,1\}^l}$ changed according to the previous set and the last transformation, so if the previous set was a function of P and y so is this set. □

Definition 13 *Let* P *be a protocol of length* l*, denote by* $\{0,1\}_1^l$ *the set of* l *bit strings with the last bit being 1. The matrices* M_A^x*,* M_B^y *of dimension* $2^{l-1} \times 2^{l-1}$ *for* P *are defined as :*

$$M_A^x(i,j) = (\vec{v}_A^i, \vec{v}_A^j)\ \ ,\ \ M_B^y(i,j) = (\vec{v}_B^i, \vec{v}_B^j) \qquad i,j \in \{0,1\}_1^l$$

where the sets of vectors $\{\vec{v}^i_A\}$, $\{\vec{v}^i_B\}$ *are identical to those appearing in the previous lemma,* $(\vec{v}^i_A, \vec{v}^j_A)$ *denotes inner product*

The set of matrices can describe the protocol, and it is important to notice that:

Observation 1 *1. The set* M^A_x *depends only on* x *and on the quantum protocol* P.

2. The set M^B_y *depends only on* y *and on* P.

Observation 2 *The matrices* M^x_A, M^y_B *are hermit and positive semi-definite.*

Proof: If we define the matrix V^x_A as the matrix where the vectors $\{\vec{v}^i_A\}$ are its rows, and V^y_B as the matrix where $\{\vec{v}^i_B\}$ are its rows we get that

$$M^x_A = V^x_A \cdot V^{x\dagger}_A \quad , \quad M^y_B = V^y_B \cdot V^{y\dagger}_B$$

And the observation follows. □

We can also assume the entries in these matrices are real:

Lemma 6 *For every quantum protocol* P*, with sets of matrices* M^x_A M^y_B*, that computes* $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ *with accuracy* ϵ*, there exists a quantum protocol* P' *s.t:*

1 $C(P') = 2C(P)$.

2 P' *computes* f *with* ϵ *accuracy.*

3 The matrices M'^x_A M'^y_B *for* P' *are real matrices.*

The proof appears in the appendix and is based on [3].

Lemma 7 $P(x,y) = (M^x_A, M^y_B)$

Proof: By lemma 5 the final state is:

$$\vec{v}_{final} = \sum_{j \in 0,1^l} \vec{v}^j_A \otimes \vec{v}^j_c \otimes \vec{v}^j_B$$

The probability of measuring 1 is:

$$pr(R_{com}(\vec{v}_{final}) = 1) = |\sum_{j\in\{0,1\}^l_1} \vec{v}^j_A \otimes \vec{v}^j_c \otimes \vec{v}^j_B|^2 = \sum_{i,j\in\{0,1\}^l_1} (\vec{v}^i_A \otimes |1\rangle \otimes \vec{v}^i_B \ , \ \vec{v}^j_A \otimes |1\rangle \otimes \vec{v}^j_B)$$

$$= \sum_{i,j\in\{0,1\}^l_1} (\vec{v}^i_A, \vec{v}^j_A) \cdot (\vec{v}^i_B, \vec{v}^j_B) = \sum_{i,j\in\{0,1\}^l_1} M^x_A(i,j) M^y_B(i,j) = (M^x_A, M^y_B)$$

□

Observation 3 *If the accuracy of the entries in the matrices is limited to* $2^{-O(l)}$*, then the following holds for a protocol* P *of cost* l*:*

$$|P(x,y) - \sum_{i,j} M^x_A(i,j) M^y_B(i,j)| \leq 2^{-\theta(l)}$$

5.3 Basic lower bounds

In this subsection we shall prove two theorems that follow from the previous subsection.

Theorem 3 *For most of the functions* $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$

$$Q(f) \geq n/2$$

Proof: The argument is based on counting. There are $2^{2^{2 \cdot n}}$ different functions $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$. There are at most $2^{2 \cdot 2^n \cdot O(l) \cdot 2^l}$ different protocols of cost l (in order to describe anyone of them we can write the $2 \cdot 2^n$ matrices). It follows that there are at most $2^{0(n \cdot 2^{3/2 \cdot n})}$ different functions with a protocol of cost $l = n/2$. (It should be remarked that if a function has a protocol of cost $l \leq n/2$ it has also a protocol of cost $l = n/2$) □

Theorem 4 $\forall f \;\; Q(f) \geq \Omega(\log Dcc(f))$

Proof: The proof is based on a simulation of a quantum protocol of cost l by a deterministic one-round protocol of cost $l \cdot 2^l$. Suppose we have a quantum protocol P for f, and the cost of the protocol is l. Using the amplification lemma (see 1) we can assume:

1. $f(x,y) = 1 \Rightarrow P(x,y) \geq 3/4$
2. $f(x,y) = 0 \Rightarrow P(x,y) \leq 1/4$

The probability that the protocol's answer on x, y is 1 is :

$$P(x,y) = (M_A^x, M_B^y)$$

We construct a 1 round deterministic protocol for f.

1. A computes the matrix M_x^A by himself, (for a given P , M_x^A depends only on x) and sends M_x^A, but only with $2^{-O(l)}$ accuracy.
2. B computes the matrix M_y^B by himself, (for a given P , M_y^B depends only on y) and subsequently computes:

$$P(x,y) = (M_x^A, M_y^B)$$

If $P(x,y) \geq 2/3$ B responds with the answer 1, if $P(x,y) \leq 1/3$ he responds with 0.

The cost of the protocol is the size of the description of the matrix M_x^A which is $\mathrm{O}(l \times 2^l)$. □

Example 7 *The equality function is defined as :*

$$EQ(x,y) = \begin{cases} 1 & x = y \\ 0 & otherwise \end{cases}$$

Where $x, y \in \{0,1\}^n$, the deterministic lower bound is $\Omega(n)$, so we get a lower bound of $\Omega(log(n))$. Because, on the other hand $R(EQ) = \theta(log(n))$ this is a tight bound, meaning that also $Q(EQ) = \theta(log(n))$ and we conclude that $Q(EQ) = R(EQ)$.

5.4 Discrepancy lower bound

This section shows that a specific technique for proving lower bounds in probabilistic communication can be generalized also to the quantum case. For this purpose we shall define the following terms.

Definition 14

1. *The communication matrix M_f for some boolean function $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ is a $2^n \times 2^n$ matrix where:*

$$M_f(i,j) = f(i,j) \qquad i,j \in \{0,1\}^n$$

2. *A rectangle R is a subset of $\{0,1\}^n \times \{0,1\}^n$ s.t:*

$$R = S \times T \qquad S,T \subseteq \{0,1\}^n$$

3. *The discrepancy of f according to a probability distribution μ is defined as:*

$$Disc_\mu(f) = max_R |Pr_\mu[(f(x,y) = 1) \cap (x,y) \in R] - Pr_\mu[(f(x,y) = 0) \cap (x,y) \in R]|$$

   where the maximum is taken over all rectangles R.

At this stage we introduce some new notations:

Notation 6

1. *Denote by $f^{-1}(1)$ the set $\{(x,y)|f(x,y) = 1\}$ and by $f^{-1}(0)$ the set $\{(x,y)|f(x,y) = 0\}$*
2. *Denote by μ_1 the probability $pr_\mu((x,y) \in f^{-1}(1))$ and by μ_0 the probability $pr_\mu((x,y) \in f^{-1}(0))$.*
3. *Denote by μ_{1R_i}, $Pr_\mu[(f(x,y) = 1) \cap (x,y) \in R]$ and by μ_{0R_i}, $Pr_\mu[(f(x,y) = 0) \cap (x,y) \in R]$*

A known technique for getting lower bound in probabilistic communication uses the following theorem:

Theorem 5 *If there exists some distribution μ over $\{0,1\}^n \times \{0,1\}^n$ with the property $Disc_\mu(f) \leq 2^{-\Omega(t)}$ then*

$$R(f) \geq \Omega(t)$$

The quantum version of this theorem can be written in a similar form:

Theorem 6 *If there exists some distribution μ over $\{0,1\}^n \times \{0,1\}^n$ with the property $Disc_\mu(f) \leq 2^{-\Omega(t)}$ then*

$$Q(f) \geq \Omega(t)$$

For the purpose of proving this theorem we shall need two lemmas:

Lemma 8 *For every quantum protocol P solving f with accuracy ϵ and for every distribution μ satisfying:*

$$|\mu_1 - \mu_0| \leq \epsilon$$

the following inequality holds :

$$\sum_{(x,y)\in f^{-1}(1)} \mu(x,y)P(x,y) - \sum_{(x,y)\in f^{-1}(0)} \mu(x,y)P(x,y) \geq 1/2 - 2\epsilon$$

Proof:

$$(x,y) \in f^{-1}(1) \Rightarrow P(x,y) \geq 1-\epsilon$$

$$(x,y) \in f^{-1}(0) \Rightarrow P(x,y) \leq \epsilon$$

$$\Rightarrow \sum_{(x,y)\in f^{-1}(1)} \mu(x,y)P(x,y) - \sum_{(x,y)\in f^{-1}(0)} \mu(x,y)P(x,y) \geq$$
$$(1-\epsilon)\mu_1 - \epsilon\mu_0 = 1 - \epsilon(\mu_0+\mu_1) == \mu_1 - \epsilon \geq 1/2 - 2\epsilon$$

□

Lemma 9 *For every two sets of vectors $\{\vec{a_i}\}_{i=1}^m$ of dimensiond, $\{\vec{b_i}\}_{i=1}^m$, whose components $\in [-1,1]$ and are given with an accuracy of $\frac{1}{d_1}$ where $d_1 = poly(d)$, there exist two sets of vectors $\{\vec{a_i'}\}_{i=1}^m$, $\{\vec{b_i'}\}_{i=1}^m$ of dimension $L = poly(d)$ which satisfy the following:*

1 $\forall i,j \quad (a_i', b_j') = (a_i, b_j)$

2 $\{\vec{a_i'}\}_{i=1}^m$ *are non-negative vectors.*

3 The non-positive envies in $\{\vec{b_i'}\}_{i=1}^m$ *are in fixed places (for every* i*).*

4 All the non-zeroes entries in $\{\vec{a_i'}\}_{i=1}^m$ *and* $\{\vec{b_i'}\}_{i=1}^m$ *equal* $\pm\frac{1}{d_1}$.

(The proof appears in the appendix)
Proof:(MAIN THEOREM)

Suppose we have a quantum protocol P for the function f of cost t we will show $t \geq \Omega(k)$: The probability of P answering 1 on the pair (x, y) is given by the inner product :

$$P(x,y) = (M_A^x, M_B^y)$$

We view the matrices M_A^x, M_B^y as vectors of dimension 4^{t-1}, (to indicate a coordinate in these matrices we use only one index). We can apply the transformation of the previous lemma (we can assume that the numbers in the vectors are written with $\frac{1}{2^{\theta(t)}}$ accuracy). The dimension of the new vectors becomes $L = 2^{\theta(t)}$. The probability of the quantum protocol answering 1 on the pair x, y :

$$P(x,y) = \sum_i (\vec{M}_A^x)_i (\vec{M}_B^y)_i = (\vec{M}_A^x, \vec{M}_B^y) = (\vec{M'}_A^x, \vec{M'}_B^y)$$

If we define the sets:

$$S_i = \{x \in X \mid i \in support(\vec{M}_A'^x)\}\ ,\ T_i = \{y \in Y \mid i \in support(\vec{M}_B'^y)\}$$

we can write:

$$P(x,y) = (\vec{M'}_A^x, \vec{M'}_B^y) = \sum_i (\vec{M}_A'^x)_i (\vec{M}_B'^y)_i = 1/2^{\theta(t)} \cdot \sum_{1 \leq i \leq L} \psi_i \chi_{S_i}(x) \chi_{T_i}(y)$$

where ψ_i is a sign, $\chi_{S_i}(x)$ is the characteristic function for $S_i \subseteq X$, and $\chi_{T_i}(y)$ is the characteristic function for $T_i \subseteq Y$. Recalling the fact that $\{0,1\}^n \times \{0,1\}^n$ is also a rectangle which implies $|\mu_1 - \mu_0| \leq \epsilon$ It follows from lemma (Lemma 5.4) that :

$$\sum_{(x,y) \in f^{-1}(1)} \mu(x,y) P(x,y) - \sum_{(x,y) \in f^{-1}(0)} \mu(x,y) P(x,y) \geq 1/2 - 2\epsilon$$

Introducing the expression for $P(x, y)$ we obtain:

$$\sum_{(x,y) \in f^{-1}(1)} \mu(x,y) \sum_{1 \leq i \leq L} \psi_i \chi_{S_i}(x) \chi_{T_i}(y) - \sum_{(x,y) \in f^{-1}(0)} \mu(x,y) \sum_{1 \leq i \leq L} \psi_i \chi_{S_i}(x) \chi_{T_i}(y) \geq 1/2 - 2\epsilon$$

By rearranging the summation we obtain:

$$1/2^{\theta(t)} \sum_{1 \leq i \leq L} \psi_i \ \Big(\sum_{(x,y) \in f^{-1}(1)} \mu(x,y) \chi_{S_i}(x) \chi_{T_i}(y) - \sum_{(x,y) \in f^{-1}(0)} \mu(x,y) \chi_{S_i}(x) \chi_{T_i}(y) \ \Big) \geq 1/2 - 2\epsilon$$

By this it can be viewed as sums and differences over L different rectangles $\{R_i = S_I \times T_i\}_{i=1}^{L}$.

$$1/2^{\theta(t)} \cdot \sum_{1 \leq i \leq L} \psi_i (\mu_{1R_i} - \mu_{0R_i}) \geq 1/2 - 2\epsilon$$

We can conclude that there must exist some rectangle R_i where

$$\mu_{1R_i} - \mu_{0R_i} \geq 2^{\theta(t)} \cdot (1/2 - 2\epsilon)/L \geq 1/(3L)$$

We assumed:

$$\forall i \;\; \mu_{1R_i} - \mu_{0R_i} \leq 2^{-\Omega(k)}$$

It follows that:

$$L \geq 2^{\Omega(k)} \Rightarrow t \geq \Omega(k)$$

□

Corollary 3 *(YAO) The $inner - product\ mod2$ function is defined as:*

$$IP_2(x,y) = \sum_{i \in [1..n]} x_i y_i \, mod2$$

It is known([]) that for the uniform distribution the following holds:

$$\forall i \;\; \mu_{1R_i} - \mu_{0R_i} \leq 2^{-\Omega(k)}$$

By this theorem we obtain

$$Q(IP_2) \geq \Omega(n)$$

which implies:

$$Q(IP_2) = R(IP_2) = \theta(n)$$

6 Complete problems

This section deals with the relation between quantum and probabilistic communication complexity by using the concept of a complete problem. In the following we shall limit the discussion to 1-round protocols only. A 1-round protocol is a protocol in which A sends a message based on its input to B, and based on this message and its input B computes the value of the function $f(x,y)$. Subsequently, B sends this value to A. A convenient representation in this case of complete problems will be inner product of vectors.

The definition of complete problems in the quantum and the probabilistic case will shed light on the relation between the two modes of communication. The definition of a complete problem in the quantum case will provide us with an explicit form of a function. The proof of either a lower bound or an upper bound to the probabilistic communication of this function will yield immediately a theorem regarding the relation between 1-round probabilistic and 1-round quantum communication complexity.

- In subsection 1 we present a fairly simple function which is complete for the class of boolean functions whose 1-round probabilistic communication complexity is $polylog(n)$. First we need to define *completeness* in this context. This is done using *rectangular reductions*, which were introduced in [1].
- In subsection 2 we shall use the definitions of subsection 1 for the quantum case and define a complete problem for the class of boolean functions whose 1-round quantum communication complexity is $polylog(n)$.
- In subsection 3 we shall define another complete problem for the quantum case. We shall present an efficient probabilistic protocol for a special case of of the quantum complete problem.

6.1 A complete problem for 1-round randomized complexity

Definition 15 *let* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *be an arbitrary function.*

- *A* deterministic 1-round communication protocol *P for f is a pair of functions $P_A : \{0,1\}^n \to \{0,1\}^c$, and $P_B : \{0,1\}^c \times \{0,1\}^n \to \{0,1\}$. The output of P on input (x,y) is $P(x,y) = P_B(P_A(x), y)$. The cost of P is c.*
- *A* probabilistic 1-round communication protocol *is a pair of functions $P_A : \{0,1\}^n \times \{0,1\}^{\rho_A} \to \{0,1\}^c$, and $P_B : \{0,1\}^c \times \{0,1\}^n \times \{0,1\}^{\rho_B} \to \{0,1\}$. The output of P on input (x,y), the (private) random coin tosses of Alice, $r_A \in \{0,1\}^{\rho_A}$, and the (private) random coin tosses of Bob, $r_B \in \{0,1\}^{\rho_B}$, is $P(x,y,r_A,r_B) = P_B(P_A(x,r_A), y, r_B)$. The cost of P is c. The* probabilistic 1-round communication complexity *of f, $R^{A\to B,\epsilon}(f)$, is defined to be the cost of the best randomized private-coin 1-round communication protocol P for which $Pr[P(x,y,r_A,r_B) \neq f(x,y)|] < \epsilon$, where the probability is taken over the random coin tosses r_A and r_B.*

Definition 16 *Let X, X', Y, Y' and Z be arbitrary sets. Let $f : X \times Y \to Z$, and let $f' : X' \times Y' \to Z$. A* rectangular reduction *from f' to f is a pair of functions $g_1 : X' \to X$, $g_2 : Y' \to Y$, which satisfy the following conditions:*

1. $\forall x' \in X',\ y' \in Y',\ f'(x',y') = f(g_1(x'), g_2(y'))$.
2. $\forall x' \in X', |g_1(x')| = 2^{polylog(|x'|)}$.
3. $\forall y' \in Y', |g_2(y')| = 2^{polylog(|y'|)}$.

If there exists a rectangular reduction from f' to f then we say that f' reduces to *f and we denote this by $f' \propto f$.*

Functions can be classified according to their communication complexity, similarly to the way in which they are classified according to their computational complexity. A *complete* function for a given communication complexity class is defined as follows.

Definition 17 *We say that a function f is* complete *for a communication complexity class $\mathcal{C}$ if the following two conditions hold:*

1. $f \in \mathcal{C}$;

2. $\forall f' \in C$, $f' \propto f$.

Let $INP_{1,\infty}(\cdot,\cdot)$ be the following inner product function. $INP_{1,\infty}(\vec{p},\vec{q}) = (\vec{p},\vec{q})$, where $\vec{p}$ and $\vec{q}$ are n-dimensional vectors which have the following properties: $\|\vec{p}\|_1 \leq 1$ ($\sum_i |p_i| \leq 1$), and $\|\vec{q}\|_\infty \leq 1$ ($\max_i |y_i| \leq 1$). Let the *partial* boolean function $INP^{par}_{1,\infty}(\cdot,\cdot)$ be defined as follows:

$$INP^{par}_{1,\infty}(\vec{p},\vec{q}) \stackrel{\text{def}}{=} \begin{cases} 1 & INP_{1,\infty}(\vec{p},\vec{q}) \geq 2/3 \\ 0 & INP_{1,\infty}(\vec{p},\vec{q}) \leq 1/3 \end{cases}$$

Theorem 7 *$INP^{par}_{1,\infty}$ is complete for the class of boolean functions whose 1-round communication complexity is $polylog(n)$.*

Proof: Without loss of generality, we assume that $\vec{q}$ is non-negative, and that $\vec{p}$ is a probability vector, i.e., $\vec{p}$ is non-negative, and $\|\vec{p}\|_1 = 1$. We start by describing a 1-round randomized protocol for computing $INP_{1,\infty}$, whose cost is $O(\log n)$. Clearly, it follows that $INP^{par}_{1,\infty}$ belongs to the class of boolean functions whose 1-round communication complexity is $polylog(n)$. Alice repeats the following process k times, where k is a constant. She chooses an index i with probability p_i and sends it to Bob. For the ℓ'th repetition of this process, let X_ℓ be the value of the q_i corresponding to the index i sent by Alice. Bob then outputs the average of the X_ℓ's. The X_ℓ's are random variables which take values in $[0,1]$, and whose expect value is $\sum_{j=1}^{n} p_j \cdot q_j$, the inner product between $\vec{p}$ and $\vec{q}$. Applying Chernoff bounds, if $k = O(1/\epsilon_2^2 \log(1/\epsilon_1))$, then with probability at least $1-\epsilon_1$, the absolute value of the difference between the average of the X_ℓ's, and $(\vec{p},\vec{q})$ is at most ϵ_2. For constant ϵ_1 and ϵ_2, the cost of the protocol is thus $O(\log n)$.

Next, we describe a rectangular reduction from any given $f : X \times Y \to \{0,1\}$ for which $R^{A\to B}(f) = polylog(n)$, to $INP^{par}_{1,\infty}$. If $R^{A\to B}(f) = polylog(n)$ then there exists a 1-round communication protocol P for f that has the following properties. For every $x \in X$, Alice's side of the For every $x \in X$, Alice's side of the protocol defines a probability distribution over all messages of length c, where $c = polylog(n)$, and for each such message, and for every $y \in Y$, Bob's side of the protocols determines a probability of outputting 1. For $1 \leq i \leq 2^c$, let M_i denote the i'th message in some arbitrary enumeration of the messages Alice can send Bob. Let $p_i(x)$ be the probability that Alice sends the message M_i to Bob given that her input is x, and let $q_i(y)$ be the probability that Bob outputs 1 given that he received the input y, and that Alice sent him the message M_i. Thus, using the notations from Definition 16, we define $g_1(x)$ to be $\vec{p}(x)$, and $g_2(y) = \vec{q}(y)$. The dimension of both vectors is 2^c which is $2^{polylog(n)}$, and we let each coordinate be written with exponential precision, using $O(n)$ bits.

It remains to be shown that $f(x,y) = INP^{par}_{1,\infty}(\vec{p}(x), \vec{q}(y))$. By definition of $\vec{p}$ and $\vec{q}$,

$$Pr[P(x,y)=1] = \sum_{i=1}^{2^c} p_i \cdot q_i \pm o(2^{-n}). \qquad (1)$$

Since $Pr[P(x,y) = 1]$ should be greater than 2/3 if $f(x,y) = 1$ and smaller than 1/3 otherwise, the claim follows. □

6.2 A complete problem for 1-round quantum communication

Definition 18 *A* 1-round quantum protocol P *acting on* $H_{m,c}$ *is a quantum protocol* P *of the form*

$$P = U_1^1, U_2^2$$

Definition 19 *The* quantum 1-round communication complexity *is defined as:*

$$Q^{A \to B, \epsilon}(f) = min\{C(P)| \ P \ is \ 1-round \ quantum \ protcol \ which \ computes \ f \ with \ accuracy \ \epsilon\}$$

Denote $Q^{A \to B, 1/3}(f)$ *by* $Q^{A \to B}(f)$.

For the purpose of analyzing 1-round quantum protocol, we will analyze the pair of matrices M_A^x, M_B^y after applying 1-round protocol:

$$\begin{aligned} \vec{P}^{x,y} &= U_2^2 \cdot U_1^1 |x, 0^c, y\rangle = U_2^2 \sum_i \vec{u}^i \otimes |i\rangle \otimes |y\rangle = \sum_i \vec{u}^i \otimes \sum_j |j\rangle \otimes \vec{v}^{ij} \\ &= \sum_i \vec{u}^i \otimes \Big(\sum_{last \ bit \ in \ j \ is \ 1} |j\rangle \otimes \vec{v}^{ij} + \sum_{last \ bit \ in \ j \ is \ 0} |j\rangle \otimes \vec{v}^{ij} \Big) \end{aligned}$$

(where 0^c means the string of c zeroes)
And we have as shown before (see Lemma 2)

1. The matrices M_A^x, M_B^y are Hermite and positive definite.

2. $P(x,y) = (M_A^x, M_B^y)$.

But for this case we also have that :

3 tr(M_A^x)=1 : This we conclude by observing that the set $\{\vec{u}^i\}$ is a set of vectors with the following property:

$$\sum_i (\vec{u}^i, \vec{u}^i) = 1$$

4 All the eigen values of $M_B^y \quad \leq 1$. This we can conclude by observing that the set $\{\vec{v}^{ij}\}$ is a set of vectors with the following property:

$$\forall\, i\ (\vec{v}^{ij}, \vec{v}^{ij}) \leq 1$$

Let $MIP_{1,\infty}$ be the following inner product function $(M1, M2)$, where $M1$ is an $n \times n$ matrix with following properties:

1 hermit

2 tr$(M1)$ =1

3 positive definite (all its eigen values ≥ 0)

and $M2$ is an $n \times n$ matrix with following properties:

1 Hermite

2 all its eigenvalues $0 \leq \lambda_i \leq 1$

Let the partial function $MIP^{par}(.,.)$ be defined as follows:

$$MIP^{par}_{1,\infty}(M1, M2) \stackrel{\text{def}}{=} \begin{cases} 1 & MIP_{1,\infty}(M1, M2) \geq 2/3 \\ 0 & MIP_{1,\infty}(M1, M2) \leq 1/3 \end{cases}$$

Theorem 8 *$MIP^{par}_{1,\infty}$ is complete for the class of boolean functions whose 1-round quantum communication complexity is $polylog(n)$.*

Proof: We describe a rectangular reduction from any given $f : X \times Y \to \{0,1\}$ for which $Q^{A\to B}(f) = polylog(n)$, to $INP^{par}_{1,\infty}$. Given a 1-round quantum protocol for a function f whose cost is $polylog(n)$ define the rectangular reduction:

1. $M1(x) = M_A^x$

2. $M2(y) = M_B^y$

Next we shall show how to approximate $MIP_{1,\infty}(M1, M2)$ using a 1-round quantum protocol of cost $polylog(n)$, this implies a quantum protocol for $MIP^{par}_{1,\infty}(M1, M2)$. For this purpose we shall need two lemmas (proofs appear in the appendix):

Lemma 10 *If a matrix M is:*

1. *positive semi-definite*

2. *hermit*

3. *tr(M) = 1*

then there exists a set of vectors $\{\vec{u}^i\}$ *s.t.*

1. $M_{i,j} = (\vec{u}^i, \vec{u}^j)$
2. $\sum_i (\vec{u}^i, \vec{u}^i) = 1$

Lemma 11 *For every matrix* $M_{n\times n}$ *which is :*

1 *hermit*

2 *with eigenvalues* $0 \leq \lambda_i \leq 1$

there exists an orthogonal projection matrix $P_{2n\times 2n}$ $(P = P^\dagger = P^2)$ *of the form:*

$$P = \begin{pmatrix} M & X \\ X^\dagger & I - M \end{pmatrix}$$

Corollary 4 *For every matrix* $M_{n\times n}$ *which is:*

1 *hermit*

2 *with eigenvalues* $0 \leq \lambda_i \leq 1$

there exist a set of orthogonal vectors $\vec{v}^i$ *each being written as sum of two orthogonal vectors* $\vec{v}^{i0} + \vec{v}^{i1}$ *with the following properties:*

1. $\forall i, j \ \ (\vec{v}^i, \vec{v}^j) = \delta_{i,j}$
2. $\forall i \ \ \vec{v}^i = \vec{v}^{i0} + \vec{v}^{i1}$
3. $\forall i, j \ \ (\vec{v}^{i0}, \vec{v}^{j1}) = 0$
4. $\forall i, j \ \ (\vec{v}^{i1}, \vec{v}^{j1}) = M(i, j)$

Since the quantum protocol we present for $MIP_{1,\infty}(M1, M2)$ preserves its input $M1, M2$ we shall sometimes omit $M1, M2$ from the description of the states. The quantum protocol for $MIP_{1,\infty}(M1, M2)$ is defined as:

1. Let $\{\vec{u}^i\}$ the set of vectors which satisfy: $M1_{i,j} = (\vec{u}^i, \vec{u}^j)$, $\sum_i (\vec{u}^i, \vec{u}^i) = 1$. Define U_1^1 as the unitary transformation with the following property:

$$U_1^1 |M1, 0^c, M2\rangle = \sum_i \vec{u}^i \otimes |i\rangle \otimes |M2\rangle$$

2. Let $\vec{v}^i$ be a set of vectors which satisfy the conditions of corollary 4. Define U_2^2 as the unitary transformation with the following property

$$U_2^2 \sum_i \vec{u}^i \otimes |i\rangle \otimes |M_2\rangle = \sum_i \vec{u}^i \otimes (|0\rangle \otimes \vec{v}^{i0} + |1\rangle \otimes \vec{v}^{i1})$$

By this we obtain:

$$\vec{P}^{x,y} = \sum_i \vec{u}^i \otimes (|0> \otimes \vec{v}^{i0} + |1> \otimes \vec{v}^{i1})$$

which implies:

1. $M_A^x(i,j) = (\vec{u}^i, \vec{u}^j) = M1_{i,j}$
2. $M_B^y(i,j) = (\vec{v}^{i1}, \vec{v}^{j1}) = M2_{i,j}$

And the probability of getting 1 when we measure is :

$$P(x,y) = (M_A^x, M_B^y) = (M1, M2)$$

□

The relation between the complete problem for the quantum case and for the probabilistic case follows form the following observations:

- The diagonal of a semi-positive-definite Hermite matrix whose eigen-values sum up to 1 is a probability vector.
- The diagonal of a semi-positive-definite Hermite matrix whose eigen-values ≤ 1 is positive vector with $L_\infty \leq 1$

Thus, the diagonals of the matrices which are the input for $MIP_{1,\infty}^{par}$ have the form of the input for $INP_{1,\infty}^{par}$. It is to be emphasized that these characteristics and the inner-product of the matrices are invariant to unitary transformations. Moreover, if one of the matrices is diagonal it follows that the inner-product of the matrices is identical with the inner product of their diagonals. Thus, we can deduce that if the players could apply the same unitary transformations to their matrices and by this to diagonalize one of the matrices then they could apply the probabilistic protocol for $INP_{1,\infty}^{par}$.

6.3 Inner product

In this section we deal with another version of a complete problem for the quantum case: Let $LEN(\vec{u}, \{\vec{v}_i\}_{i=1}^k)$ be defined as:

$$LEN(\vec{u}, \{\vec{v}_i\}_{i=1}^k) \stackrel{\text{def}}{=} \sum_i (\vec{u}, \vec{v}_i)^2$$

where $\vec{u}$ is a n dimension vector whose L_2 norm equals 1 and $\{\vec{v}_i\}_{i=1}^k)$ are $k \leq n$ orthonormal vectors. Let the partial function $Len^{par}(\vec{u}, \{\vec{v}_i\}_{i=1}^k)$ be defined as follows

$$Len^{par}(\vec{u}, \{\vec{v}_i\}_{i=1}^k) \stackrel{\text{def}}{=} \begin{cases} 1 & LEN(\vec{u}, \{\vec{v}_i\}_{i=1}^k) \geq 2/3 \\ 0 & LEN(\vec{u}, \{\vec{v}_i\}_{i=1}^k) \leq 1/3 \end{cases}$$

Next we show by that:

Theorem 9 *$Len^{par}(\vec{u}, \{\vec{v}_i\}_{i=1}^{k})$ is complete for the class of boolean functions whose 1-round quantum communication complexity is polylog(n).*

Proof:

- We construct a quantum protocol for LEN^{par} by showing a reduction from LEN^{par} to MIP^{par}, define:

  1. $M1_{i,j} = \vec{u}_i \vec{u}_j^*$
  2. $M2_{i,j} = \sum_l \vec{v}_i^l \vec{v}_j^{l^*}$

  This implies:

$$\begin{aligned}(M1, M2) &= \sum_{i,j} \vec{u}_i \cdot \vec{u}_j^* \sum_l \vec{v}_i^l \cdot (\vec{v}_j^l)^* = \sum_l (\sum_i \vec{u}_i \cdot \vec{v}_i^l)(\sum_j \vec{u}_j \cdot \vec{v}_j^l)^* \\ &= \sum_l |(\vec{u}, \vec{v}^l)|^2 = LEN(\vec{u}, \{\vec{v}^i\}_{i=1}^k)\end{aligned}$$

  We conclude that there exists a quantum protocol for LEN^{par} of cost $polylog(n)$.

- Next we show how to reduce $MIP^{par}(M1, M2)$ to $LEN^{par}(\vec{u}, \{\vec{v}^i\})$. From the fact that if a matrix M is:

  1. hermit
  2. All its eigenvalues equal 0 except λ_i which equals 1.

  there a exist a vector $\vec{u}$ with the following properties:

  1. $M_{i,j} = \vec{u}_i \vec{u}_j^*$
  2. $(\vec{u}, \vec{u}) = 1$

  We conclude that :

  1. there exists a set of orthonormal vectors $\{\vec{u}^l\}$, and a probability vector $\vec{\lambda}$ s.t:

$$M1_{i,j} = \sum_l \vec{\lambda}_l \cdot \vec{u}_i^l \cdot (\vec{u}_j^l)^*$$

  2. there exists another set of orthonormal vectors $\{\vec{v}^l\}$ s.t:

$$M2_{i,j} = \sum_l \vec{v}_i^l \cdot (\vec{v}_j^l)^*$$

1,2 imply:

$$(M1, M2) = \sum_l \lambda_l \; LEN(\vec{u^l}, \{\vec{v}^i\})$$

By this we conclude that given a protocol to LEN^{par} one can pick a random l and compute $LEN^{par}(\vec{u^l}, \{\vec{v}^i\})$. Repeating this process a constant number of times and taking the majority result will give a protocol to $MIP^{par}(M1, M2)$.

□

We continue by giving an efficient probabilistic protocol for the problem of approximating the inner product of two n-dimensional vectors whose L_2 norm is bounded by 1. which is a special case of LEN ($k = 1$). More precisely, we define:

$$INP_{2,2}^{par}(\vec{u}, \vec{v}) \stackrel{\text{def}}{=} \begin{cases} 1 & |(\vec{u}, \vec{v})|^2 \geq 2/3 \\ 0 & |(\vec{u}, \vec{v})|^2 \leq 1/3 \end{cases}$$

Theorem 10 $R^{A \to B}(INP_{2,2}^{par}) = polylog(n)$.

Proof: We describe a 1-round randomized communication protocol for approximating with an additive constant the inner product of two vectors whose L_2 norm equal 1. Without loss of generality, we may assume that the vectors $\vec{u}$ and $\vec{v}$ are positive. [1]

Let ℓ denote the length of the representation of each coordinate in $\vec{u}$ and $\vec{v}$, where we may assume that $\ell = polylog(n)$. Alice and Bob transform their vectors into sums of binary vectors. More explicitly, let

$$\vec{u} = \sum_{j=1}^{\ell} 2^{-j+1} \vec{\mu}^j \;\; , \;\; \vec{v} = \sum_{j=1}^{\ell} 2^{-j+1} \vec{\nu}^j \qquad \forall i, j \; \vec{\mu}_i^j, \vec{\nu}_i^j \in \{0, 1\} \;\; , \tag{2}$$

where $\vec{\mu}_i^j$ equals u_{ij}, the j'th bit in the binary representation of the i'th coordinate of $\vec{u}$, u_i, and $\vec{\nu}_i^j$ is defined similarly.

We next make the following two key observations.

1. $(\vec{u}, \vec{v}) = \sum_{j,k=1}^{l} 2^{-(j+k)+2} (\mu^j, \nu^k)$;
2. $\forall j, \; 1 \leq j \leq l, \; sup(\vec{\mu}^j), sup(\vec{\nu}^j) \leq 2^{2j-2}$, where for a vector $\vec{r}$, $sup(\vec{r})$, (the *support* of $\vec{r}$), is the number of non-zero coordinates in $\vec{r}$.

The first observation can be verified through the following sequence of equalities.

$$(\vec{u}, \vec{v}) \;\; = \;\; \sum_{i=1}^{n} u_i \cdot v_i \tag{3}$$

[1]If this is not the case then let $\vec{u} = \vec{u}^+ + \vec{u}^-$, and $\vec{v} = \vec{v}^+ + \vec{v}^-$, where $\vec{u}^+$ and $\vec{v}^+$ are positive vectors and $\vec{u}^-$ and $\vec{v}^-$ are negative vectors. Then $(\vec{u}, \vec{v}) = (\vec{u}^+, \vec{v}^+) + (-\vec{u}^-, -\vec{v}^-) + (\vec{u}^+, -\vec{v}^-) + (-\vec{u}^-, \vec{v}^+)$.

$$
\begin{aligned}
&= \sum_{i=1}^{n}\sum_{j=1}^{l}\sum_{k=1}^{l} 2^{-j+1}u_{ij} \cdot 2^{-k+1}u_{ik} && (4)\\
&= \sum_{j,k=1}^{l} 2^{-(j+k)+2} \sum_{i=1}^{n} u_{ij} \cdot v_{ik} && (5)\\
&= \sum_{j,k=1}^{l} 2^{-(j+k)+2} \cdot (\mu^j, \nu^k). && (6)
\end{aligned}
$$

The second observation is true since following the first observation:

$$
\begin{aligned}
(\vec{u}, \vec{u}) &= \sum_{j,k} 2^{-(j+k)+2} \cdot (\vec{\mu}^j, \vec{\nu}^k) && (7)\\
&\geq \sum_{j} 2^{-2j+2} \cdot sup(\vec{\mu}^j) \ , && (8)
\end{aligned}
$$

but $(\vec{u}, \vec{u}) \leq 1$ and hence in particular $\forall j,\ 2^{-2j+2} \cdot sup(\vec{\mu}^j) \leq 1$.

Let $a_{j,k} \stackrel{\text{def}}{=} 2^{-(j+k)+2}(\vec{\mu}^j, \vec{\nu}^j)$. Alice and Bob compute each $a_{j,k}$ separately, and then sum them all up. The number of pairs (j,k) is $polylog(n)$, and hence it remains to show how these values can be computed each with $polylog(n)$ bits of communication, $1/polylog(n)$ accuracy, and with confidence at least $1 - 1/polylog(n)$.

We separate the discussion into two cases: $j \leq k$, and $j > k$.

- $\boldsymbol{j \leq k}$: Alice repeats the following process $polylog(n)$ times. She picks a coordinate i in $sup(\vec{\mu}^j)$, uniformly, and at random, and sends i to Bob. Clearly, the corresponding coordinate of $\vec{\nu}^k$, ν_i^k, is a $\{0,1\}$ random variable whose expectation is:

  $$\frac{(\vec{\mu}^j, \vec{\nu}^k)}{|sup(\vec{\mu}^j)|} \ .$$

  Since this process is repeated $polylog(n)$ times, if we apply Chernoff bounds, we get that with high probability, the average value of the ν_i^k's approximates $(\vec{\mu}^j, \vec{\nu}^k)/|sup(\vec{\mu}^j|)$ within an additive error of $1/polylog(n)$. Alice also sends Bob the size of $sup(\vec{\mu}^j)$, and Bob then multiplies the average of the ν_i^k's by $sup(\vec{\mu}^j)$ and by $2^{-(j+k)+2}$ to get an approximation of $a_{j,k}$. With high probability $(1 - 2^{-polylog(n)})$, the error of this approximation is

  $$\frac{2^{-(j+k)+2} \cdot |sup(\vec{\mu}^j|}{polylog(n)} \ .$$

  Since $k \geq j$, and using the second observation above, we get that the error is bounded by $1/polylog(n)$, as required. The communication complexity is clearly $polylog(n)$.

- $\boldsymbol{j > k}$: This case can be divided into two sub-cases:

1. $2^{j-k} \leq polylog(n)$: Alice and Bob essentially follow the same protocol as described above for the case $j \leq k$. Since

$$2^{-(j+k)+2} \cdot |sup(\vec{\mu}^j)| \leq 2^{j-k},$$

the claim follows.

2. $2^{j-k} > polylog(n)$: in this case

$$\begin{aligned} a_{j,k} &= 2^{-(j+k)+2}(\vec{\mu}^k, \vec{\nu}^l) && (9)\\ &\leq 2^{-(j+k)+2}|sup(\vec{\nu}^k)| && (10)\\ &\leq 2^{k-j} \leq \frac{1}{polylog(n)}\ , && (11) \end{aligned}$$

and we can ignore all such pairs by assuming the corresponding $a_{j,k}$'s are 0.

□

7 Appendix

7.1 Some mathematical notations

The mathematical notions we will deal with throughout this work come from Linear Algebra. We will use standard notations:

1. Vectors- We use the notation of $\vec{v}$ to denote a vector, In some special cases we use Dirac notation $|x>$ to specify a vector.

2. Conjugate matrix - Given a matrix M we denote by $M^\dagger$ the conjugate matrix of M:

$$(M^\dagger)_{i,j} = (M_{j,i})^*$$

3. Inner product- We use the notation of $(\vec{v}^1, \vec{v}^2)$ to denote the inner product of two vectors $\vec{v}^1, \vec{v}^2$. Given two matrices M^1, M^2 we denote by $(\vec{M}^1, \vec{M}^2)$ the inner product of these matrices.

$$(\vec{M}^1, \vec{M}^2) = \sum_{i,j} M^1_{i,j} \dot{M}^2_{i,j}$$

4. Projection- Given a linear space H', a vector in this space $\vec{v}$ and a subspace H' of H' with an orthonormal basis $\{\vec{u}_i\}$. We denote by $proj_{H'}\vec{v}$ the vector which is the orthonormal projection of $\vec{v}$ on H':

$$proj_{H'}\vec{v} = \sum_i (\vec{v}, \vec{u}_i)\vec{u}_i$$

5. Tensor product- Given two vectors $\vec{v}^1, \vec{v}^2$ we denote by $\vec{v}^1 \otimes \vec{v}^2$ their tensor product.

7.2 Tensor Product

In the following section we describe briefly the notion of tensor product of vector spaces. The description is not intended to be an axiomatic definition of tensor product,rather a description of its properties, (see [15]). Tensor product spaces are used in quantum mechanics for the description of many particles systems . For the description of a single particle i we use a vector space H_i. For the description of a n particles system we use the vector space H which is the tensor product $H_1 \otimes \cdots \otimes H_n$.

Let H_1, H_2 be two vector spaces of dimensions n_1, n_2 respectively. We build the tensor product of these two spaces $H = H_1 \otimes H_2$ (a vector space of dimension $n_1 n_2$) in the following way :
To every pair of vectors $\vec{v_1}$ $\vec{v_2}$ we associate a vector $\vec{v} = \vec{v_1} \otimes \vec{v_2}$ in H. This association has the following properties:

1. It is linear with respect to multiplication with complex number :

$$(\lambda \vec{v_1}) \otimes \vec{v_2} \quad = \quad \vec{v_1} \otimes (\lambda \vec{v_2}) = \lambda(\vec{v_1} \otimes \vec{v_2})$$

2. It is distributive :

$$\vec{v_1} \otimes (\vec{v_2^1} + \vec{v_2^2}) = \vec{v_1} \otimes \vec{v_2^1} + \vec{v_1} \otimes \vec{v_2^2}$$
$$(\vec{v_1^1} + \vec{v_1^2}) \otimes \vec{v_2} = \vec{v_1^1} \otimes \vec{v_2} + \vec{v_1^2} \otimes \vec{v_2}$$

3. If a basis is chosen in each of the spaces H_1 H_2 $\{\vec{v_1^i}\}$ $\{\vec{v_2^j}\}$ then the set of vectors $\{\vec{v_1^i} \otimes \vec{v_2^j}\}$ constitutes a basis for H.

4. Not every vector $\vec{v}$ in H is a tensor product of two vectors from H_1, H_2 , but it can be written as a sum of tensor product of vectors.
$\vec{v} = \sum_{i,j} a_{i,j} \vec{v_1^i} \otimes \vec{v_2^j}$

5. The inner product of vectors, which are tensor products themselves is defined as :

$$(\vec{v}_1 \otimes \vec{v}_2, \vec{u}_1 \otimes \vec{u}_2) = (\vec{v}_1, \vec{u}_1) \cdot (\vec{v}_2, \vec{u}_2)$$

6. We can extend this definition of the tensor product of two spaces to tensor product of three spaces (or n spaces) and we define $H = H_1 \otimes H_2 \otimes H_3$ as $(H_1 \otimes H_2) \otimes H_3$

7.3 Appendix for section 4

Lemma **1** *1 Let* X *be* $\{0,1\}^n$, Y *be* $\{0,1\}^k$.

For every two sets sets $\{a_{x,i}\}\{b_{x,i}\}$ satisfying that for every $x \in \{0,1\}^n$:

$$\sum_{i\in\{0,1\}^k} |a_{x,i}|^2 = \sum_{i\in\{0,1\}^k} |b_{x,i}|^2 = 1$$

there exists an unitary transformation U acting on a space of 2^{n+k} dimensions s.t for every $x \in \{0,1\}^n$:

$$U \sum_{i\in\{0,1\}^k} a_i^x |i \circ x\rangle = \sum_{i\in\{0,1\}^k} b_i^x |i \circ x\rangle$$

Proof: It is known that given a set of k orthonormal vectors. There exists an unitary matrix $M_{n\times n}$ s.t. these k vectors are part of its n rows. Similarly one can express that as given a set of k orthonormal vectors in a n dimension space, one can extend this set to be a basis. A careful looks of the conditions in this lemma show that they are of this form, thus the lemma follows immediately. □

LEMMA **2** *2 For every protocol P in $H_{m,c}$ that computes a function $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ with accuracy ϵ there exists a protocol P' which acts on $H_{m+n,c}$ that computes f with accuracy ϵ , and preserves the input, moreover $C(P) = C(P')$.*

Proof: Suppose we have a protocol P:

$$U_1^0, U_2^1, U_1^2, U_2^3, ..U_1^{l-1}, U_2^l$$

We will replace U_1^0 with a transformations that preserves its input meaning that if :

$$U_1^0|x,0,y\rangle = \sum_{i,b} a_{i,b}|i,b,y\rangle$$

We will build U'^0_1 with the conditions (using Lemma 1) :

$$U'^0_1|x,0,y\rangle = \sum_{i,b} a_{i,b}|i \circ x, b, y\rangle$$

Next we will show how to generalize this idea and replace U_1^l (we will handle U_2^l the same way). Suppose:

$$\begin{aligned} U_2^{l-1}\cdots U_1^0|x,0,y\rangle &= \sum_{i,b,j} a_{i,b,j}^{x,y}|i,b,j\rangle \\ \textit{and} \qquad U_1^l \sum_{i,b,j} a_{i,b,j}^{x,y}|i,b,j\rangle &= \sum_{i,b,j} a'^{x,y}_{i,b,j}|i,b,j\rangle \end{aligned}$$

we will require that :

$$U'^l_1 \sum_{i,b,j} a_{i,b,j}|i \circ x, b, j \circ y\rangle = \sum_{i,b,j} b_{i,b,j}|i \circ x, b, j \circ y\rangle$$

By (Lemma 1) we can find such ${U'}^l_1$ and it is easy to see that if U^l_1 acts on $H_1 \otimes H_c$ so does ${U'}^l_1$. By induction one can easily see that if :

$$\vec{P}^{x,y} = \sum_{i,b,j} a_{i,b,j}|i,b,j\rangle = v^{\vec{p}}_{x,y}$$

Then

$$\vec{P'}^{x,y} = \sum_{i,k,j} a_{i,k,j}|i \circ x,,y \circ j\rangle = \vec{v}^{p'}_{x,y}$$

It follows that C(P)=C(P') and that $R_{com}(\vec{v}^{p'}_{x,y}) = R_{com}(\vec{v}^{p}_{x,y})$ $\square$

LEMMA **3** *Let f_1 and f_2 be boolean functions $\{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. In addition let h be a boolean function $\{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ defined as $h(x,y) = g(f_1(x,y), f_1(x,y))$ for some function $g : \{0,1\}^2 \to \{0,1\}$. If there exist two quantum protocols P_1, P_2 which compute f_1, f_2 respectively with accuracy ϵ then there exists a third quantum protocol P with the following properties:*

- *P computes h with accuracy $\epsilon' = pr_{r_1,r_2}[g(r_1,r_2) \neq h(x,y)]$ where r_1, r_2 are two independent Bernoulli variables ($\in \{0,1\}$) with the property: $pr[r_1 = f_1(x,y)] = pr[r_2 = f_2(x,y)] = 1-\epsilon$*
- *$C(P) = C(P_1) + C(P_2)$*

Proof: Suppose we have a quantum protocol P_1, P_2 for $f_1 f_2$, we assume that these protocols preserve their inputs:

$$\vec{P}^{x,y}_1 = \sum_{i,j,k} a_{i,j,k}|x \circ i,\ k,\ y \circ j\rangle \quad , \vec{P}^{x,y}_2 = \sum_{i',j',k'} b_{i',j',k'}|x \circ i',\ k',\ y \circ j'\rangle$$

Define the protocol P as:

$$P = U_{move\ (2,c)}, U_g, U_{move\ (c,2)}, P_2, U_{move\ (c,2)}, P_1$$

Note that $C(P) = C(P_1) + C(P_2)$ this is because we can replace the last transformation in P_1 which is U^2_l with $U_{move\ (c,2)} \cdot U^2_l$, the same holds for $U_g, U_{move\ (2,c)}$. By applying P we get:

$$\vec{P}^{x,y} = \sum_{i,j,k} \sum_{i',j',k'} a_{i,j,k} b_{i',j',k'}|x \circ i \circ i', g(k,k'),\ y \circ j \circ j'\rangle$$

If we now use R_{com} we will measure a bit b with probability:

$$\sum_{i,j,k} \sum_{i',j',k'} |a_{i,j,k} \cdot b_{i',j',k'}|^2 \quad where\ \ g(k,k') = b$$

Since P_1, P_2 compute f_1, f_2 with accuracy ϵ we have that:

$$\sum_{i,j} |a_{i,j,f_1(x,y)}|^2 = \sum_{i',j'} |b_{i',j',f_2(x,y)}|^2 = 1 - \epsilon$$

and we conclude that P computes h with accuracy $\epsilon' = pr_{r_1,r_2}[g(r_1, r_2) \neq h(x, y)]$ where r_1, r_2 are two independent Bernoulli variables ($\in \{0,1\}$) with the property: $pr[r_1 = f_1(x, y)] = pr[r_2 = f_2(x, y)] = 1 - \epsilon$ $\square$

7.4 Appendix for section 5

LEMMA **4** *9 For every two sets of vectors* $\{\vec{a_i}\}_{i=1}^m$ *of dimension* d, $\{\vec{b_i}\}_{i=1}^m$, *whose components* $\in [-1, 1]$ *and are given with an accuracy of* $\frac{1}{d_1}$ *where* $d_1 = poly(d)$, *there exist two sets of vectors* $\{\vec{a'_i}\}_{i=1}^m$, $\{\vec{b'_i}\}_{i=1}^m$ *of dimension* $L = poly(d)$ *which satisfy the following:*

1 $\forall i, j \quad (a'_i, b'_j) = (a_i, b_j)$

2 $\{\vec{a'_i}\}_{i=1}^m$ *are non-negative vectors.*

3 The non-positive entries in $\{\vec{b'_i}\}_{i=1}^m$ *are in fixed places (for every i).*

4 All the non-zeroes entries in $\{\vec{a'_i}\}_{i=1}^m$ *and* $\{\vec{b'_i}\}_{i=1}^m$ *equal* $\pm\frac{1}{d_1}$.

Proof: We will perform two transformations each one will preserve the inner product. The first one makes the set of vectors $\{\vec{a''_i}\}_{i=1}^m$ non-negative, and negative entries in $\{\vec{b''_i}\}_{i=1}^m$ will be in fixed places: The second transformation makes the non-zeroes entries equal $\pm\frac{1}{d_1}$.

In the first transformation replace each entry with four entries, thus the vectors become of dimension $4d$. Denote by a_i^j the $j'th$ entry in $\vec{a_i}$ we replace this entry with $(a_i^j + 1, 1, a_i^j + 1, 1)$, denote by b_i^j the $j'th$ entry in $\vec{b_i}$ we replace this entry with $(b_i^j - 1, -b_i^j - 1, 1, 1)$:

$$\begin{pmatrix} \vdots \\ a_j^i \\ \vdots \end{pmatrix} \Rightarrow \begin{pmatrix} \vdots \\ a_j^i + 1 \\ 1 \\ a_i^j + 1 \\ 1 \\ \vdots \end{pmatrix} \quad , \quad \begin{pmatrix} \vdots \\ b_j^i \\ \vdots \end{pmatrix} \Rightarrow \begin{pmatrix} \vdots \\ b_i^j - 1 \\ -b_i^j - 1 \\ 1 \\ 1 \\ \vdots \end{pmatrix}$$

The inner product is preserved:

$$a_i^j \cdot b_i^j = (a_i^j + 1)(b_i^j - 1) + 1 \cdot (-b_i^j - 1) + 1 \cdot (a_i^j + 1) + 1 \cdot 1$$

Since the entries in $\{\vec{a_i}\}$ are $\in [-1, 1]$, the entries become non-negative. On the other hand negative entries in $\{\vec{b_i}\}_{i=1}^m$ are in coordinates $j's$ s.t. $j \ mod\ 4 = 1, 2$.

In the second transformation replace each entry with a matrix of $2d_1 * 2d_1$ cells, where in the (i, j) cell there are $i * j$ entries. We replace a_i^j ($j'th$ entry of $\vec{a_i}$) with a matrix where whole the cells are filled with zero-entries, except the $d_1 * a_i^j$ row where in each cell all the entries will be $\frac{1}{d_1}$. the new vector will be of $d_1^5 = poly(d)$ dimensions. Similarly we replace b_i^j with a matrix where whole the cells are filled with zero-entries, except the $d_1 * b_i^j$ column where in each cell all the entries will be $\frac{1}{d_1}$. The inner product is preserved because the inner product of two matrices is: $(\frac{1}{d_1})^2 * (d_1 * a_i^j) * (d_1 * b_i^j) = a_i^j \ b_i^j$ □

7.5 Appendix for section 6

Lemma 12 *If a matrix M is:*

1. *positive semi-definite*
2. *hermit*
3. $tr(M) = 1$

then there exists a set of vectors $\{\vec{u}^i\}$ s.t.

1. $M_{i,j} = (\vec{u}^i, \vec{u}^j)$
2. $\sum_i (\vec{u}^i, \vec{u}^i) = 1$

Proof: The proof follows from the fact that for a matrix M with properties 1,2 there exists a matrix X s.t $XX^\dagger = M$. We define $\vec{u}^i$ as the $i'th$ row of X which implies $M_{i,j} = (\vec{u}^i, \vec{u}^j)$. For diagonal entries we have $M_{i,i} = (\vec{u}^i, \vec{u}^i)$ and we conclude that $tr(M) = 1 \Rightarrow \sum_i (\vec{u}^i, \vec{u}^i) = 1$ □

Lemma 13 *For every matrix $M_{n \times n}$ which is :*

1 *hermit*

2 *with eigenvalues* $0 \leq \lambda_i \leq 1$

there exists an orthogonal projection matrix $P_{2n \times 2n}$ $(P = P^\dagger = P^2)$ of the form:

$$P = \begin{pmatrix} M & X \\ X^\dagger & I - M \end{pmatrix}$$

Proof: Consider the matrix T

$$\begin{pmatrix} M & X \\ X^\dagger & I - M \end{pmatrix}$$

It follows that $T = T^\dagger$. We have to verify that there exists a $n \times n$ matrix X s.t. $T^2 = T$. This is iff: $XX^\dagger = M - M^2$. M is a positive definite hermit matrix with eigenvalues smaller than 1 it follows that $M - M^2$ is a positive definite hermit matrix. Which implies that such X exists. □

References

[1] L. Babai, P. Frankl, and J. Simon. Complexity classes in communication complexity theory. In *27th Annual Symposium on Foundations of Computer Science*, pages 337–347, 1986.

[2] C. Bennet and G. brassard. The dawn of a new era for quantum cryptography: the experimental prototype is working. *SIGACT News*, 20:78–82, 1989.

[3] E. Bernstein and U. Vazirani. Quantum complexity theory. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on the Theory of Computing*, pages 11–20, 1993.

[4] C. Bennet F. Bessete G. brassard L. Salvail and J. Smolin. Experimental quantum cryptography. *Journal of Cryptology*, 5:3–28, 1992.

[5] A. Church. An unsolvable problem of elemantry number theory. *American journal of Mathematics*, 2(58):345–363, 1936.

[6] D. Deutch. Quantum theory, the church-turing principle and the universal quantum computer. In *Proc. Roy. Soc. Lond, Vol. A400*, pages 96–117, 1985.

[7] R. Feynman. Simulating physics with computers. In *International Journal of Theoretical Physics, Vol. 21, No. 6/7*, pages 467–488, 1982.

[8] C. Bennet C. Crepaeu R. Joza and D. Langlois. Quantum bit commitment and coin tossing protocols. In *34th Annual Symposium on Foundations of Computer Science*, pages 362–369, 1993.

[9] Bernard Korte and Laslo Lovasz. *Paths, Flows, and VLSI-LAYOUT*. Springer-Verlag, 1990.

[10] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Manuscript, 1995.

[11] I. Kremer N. Nissan and D. Ron. On one-round randomized communication complexity. In *Proceedings of the Twenty-Seventh Annual ACM Symposium on the Theory of Computing*, page unknown yet, 1995.

[12] Christos H. Papadimitriu. *Computational Complexity*. Addison-Wesley, 1994.

[13] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th Annual Symposium on Foundations of Computer Science*, page unknown yet, 1994.

[14] D. Simon. On the power of quantum computation. In *35th Annual Symposium on Foundations of Computer Science*, page unknown yet, 1994.

[15] Cohen Tanudji. *Quantum Mechanics*. Springer-Verlag, 1984.

[16] A.M. Turing. On computable numbers with an apllication to the entscheid- ungsproblem. *Proceeding London Mathematical Society*, 2(42):230–365, 1936.

[17] A. Yao. Some complexity questions related to distributive computing. In *Proceedings of the Eleventh Annual ACM Symposium on Theory of Computing*, pages 209–213, 1979.

[18] A. Yao. Quantum circuit complexity. In *34th Annual Symposium on Foundations of Computer Science*, pages 352–361, 1993.