Master Erasmus Mundus in Photonics Engineering, Nanophotonics and Biophotonics

Europhotonics

MASTER THESIS WORK

QUANTUM COMPUTING BY INTERROGATION

Ivan Šupić

Supervised by Dr. Antonio Acín and Dr. Matty J. Hoban
(ICFO – The Institute of Photonic Sciences)

Presented in Barcelona, on 9th September 2014

Registered at

Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona

Abstract

Quantum information theory forms a bridge between the foundations of quantum mechanics and its promising practical potentials. The extensive theoretical research has been conducted during the last decades and the applications such as quantum key distribution and quantum computing promise to provide real technological breakthroughs. Many experimental groups worldwide are working on the practical implementation of the concepts from the quantum information theory. In the recent years a lot of attention has been paid to the certification of truly quantum behavior of some device, claimed to be quantum. This certification can be done by using some established interactive proof in which classical user just by commanding to a quantum device (prover) can make himself sure that the device works as advertised and moreover can actually use it for some particular task. When it comes to quantum computing one has to be sure that a machine uses quantum mechanics for the computational process. Two interesting interactive proofs are developed so far. One of them uses specific computation model, Measurement-based quantum computing, which is performed on a class of highly entangled states, graph states. The difficulty with the implementation of this interactive proof is a big number of provers required for the process. In this thesis we present some initial results of the attempts to construct interactive proof that would use smaller number of provers. Difficulties of the two-prover interactive proof and guidelines for the future research are presented as well as the new three-prover interactive proof for quantum device performing computation by doing single qubit unitary operations.

Acknowledgements

After finishing my master program I have to thank to some people for having their part in making it as good as it has been. On the first part I would like to thank to my master thesis supervisors Antonio Acin and Matty Hoban. Thanks to prof. Acin for letting me do my thesis in his group in ICFO, for good will and useful discussions on the topic. There are not enough words to express gratitude to Matty Hoban. During these last months I had this great opportunity to learn so much from him. Thanks for this exciting project, for the enthusiasm, invested time, the patience, everything. I had a great reference for the future research. I am also grateful to the whole Quantum information theory group in ICFO, for the pleasant and very motivating atmosphere they have created. Particularly I would like to thank to my office-mate Paul Skrzypczyk for answering my questions "out of nowhere" and help with SDP and to Ariel Bendersky for the help with the last part of my thesis and useful suggestions. Outside of the group I would like to thank to my friends Marko Spasenovic and Marko Andrijasevic for their help during the writing process.

This thesis is the final part of the Europhotonics master program. I am very grateful to all those who made this master program possible. I would like to single out professors Ramon Vilaseca and Crina Cojocaru from UPC for all the help and enthusiasm. At the end I have to mention those being there with me and for me from the start of this master program, my 16 fellow colleagues coming from all over the world and thank them for all the great time we have spent together during the last two years.

Contents

Chapter 1

Introduction

Since its beginnings quantum theory has been very successfully incorporated into all branches of physics. After the initial success of quantum mechanics, it fused with other theories and the emergence of quantum field theory, quantum electrodynamics and quantum optics boosted our understanding of the ways the nature functions. After demonstrating its power in describing different natural phenomena, the question about its application in computing and information theory naturally arose. If computing and information processing are physical phenomena then quantum theory should help us understand these processes in more details. This way of thinking was shown to be fruitful. Past decades brought discoveries showing a clear advantage of quantum computing and quantum information theory compared to their classical counterparts. Quantum teleportation, Super-dense coding, Quantum key distribution and Shor's factorization algorithm are just some of the breakthroughs. All these achievements are theoretical and substantial efforts are being made to implement quantum information theory in practice. In 2011 it was claimed that the first quantum annealer has been built [JAG+11], but the scientific community is not convinced [SSSV14, Aar13]. In the end, how can we know if some machine is really a quantum computer and not some superpowerful network of classical computers? One would be ready to pay a lot of money for such an exclusive computer, so it would be good if there would be some procedure able to verify the "quantumness". One possibility is to ask this computer to solve some problem which a quantum computer can easily solve, but is hard for a classical computer. The example of such a problem might be the factorization of an integer number. Up to now there is no classical algorithm able to solve this problem efficiently (in terms of time), yet a quantum computer can use Shor's algorithm to find a solution efficiently. The approach with this kind of problems, however, is not appropriate. Some classical supercomputer might simply get lucky and solve the problem quickly. Another possibility is to develop a suitable classical algorithm. We might know that for now there is no such an algorithm, but there is no proof that it cannot exist. In other words the success of this method relies on a conjecture that some specific problem cannot be solved efficiently on a classical computer. The method we would like to have should be able to check if a computer actually performs quantum operations, and ideally should rely only on the correctness of quantum mechanics.

This master thesis deals with the effort to find the best possible method for the certification of the claim that some computer is quantum. The current state of the affairs will be described (Chapter 1), as well as some new results (Chapter 2). The main idea is to construct an interactive proof that we could use not just to check if we work with a real quantum computer but also to perform some computations. Two kinds of interactive proofs are developed so far and the detailed description is given in section 1.7. Section 1.1 will introduce quantum bits, gates and circuits, as the main building blocks of a quantum computer. The most important quantum algorithms, demonstrating the potentials of quantum computing are given in section 1.2. Section 1.3 describes a special class of quantum states, graph states, that represent

a resource for Measurement-based quantum computing, described in section 1.4. This model of quantum computing is important for us because it has been used in one of the interactive proofs. Section 1.6 describes self-testing, the procedure that can be used in interactive proofs for certifying quantum states and quantum operations. This procedure relies on nonlocality of the examined system, so the concept of a quantum nonlocality, as genuinely non-classical effect, is explained beforehand in section 1.5.

1.1 Quantum bits, gates and circuits

The main building blocks of every quantum circuit are:

- *qubits*, representing the states,
- *unitary operations*, providing dynamics and
- *quantum measurements* enabling the readout.

A quantum computer uses quantum bits or qubits for the computation process. As the name suggests a qubit is a quantum counterpart of a bit. As one bit can have one of the two values (0 or 1) the qubit is a quantum system which can be in one of the two states, denoted as vectors in the corresponding Hilbert space: $|0\rangle$ and $|1\rangle$.[1] The important difference between a bit and a qubit is that the latter can also be in any normalized superposition of two states: $\alpha|0\rangle + \beta|1\rangle$, such that $\alpha^2 + \beta^2 = 1$. A qubit can be any physical system that has two possible states, for example a photon that can have vertical ($|V\rangle \equiv |0\rangle$) or horizontal ($|H\rangle \equiv |1\rangle$) polarization, an electron with two spin states ($|\text{up}\rangle$ and $|\text{down}\rangle$) or two different energy levels... Qubits are often represented as points on the Bloch sphere that has a unit radius. Having in mind how $|0\rangle$ and $|1\rangle$ are defined on the Bloch sphere every qubit can be written in the form:

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle \tag{1.1.1}$$

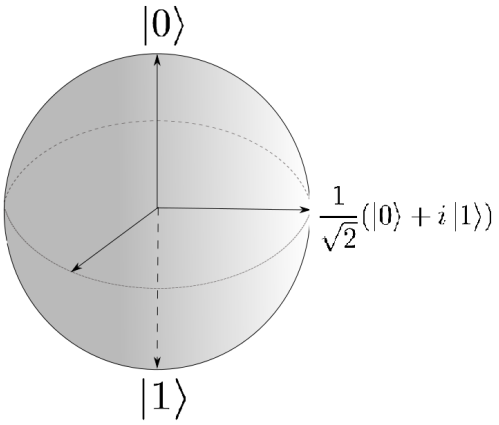


Figure 1.1: Bloch sphere. Vectors $|0\rangle$ and $|1\rangle$ are eigenvectors of the Pauli Z, so the axis which they define is z-axis. The equatorial plane is xy-plane. The vector $\frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$ as the eigenvector of Pauli Y shows the positive direction of the y-axis.

Since a qubit can be any point on the Bloch sphere one may conclude that it can be in one of infinitely many possible states. This is true, but since its Hilbert space is two-dimensional,

[1] Here we have used Dirac notation, in which all the vectors are called "kets" and are denoted as $|k\rangle$. Dual vectors are called "bras" and denoted as $\langle b|$. In this notation the scalar product can be nicely written in the bra-ket notation $\langle b|\,k\rangle$.

upon special measurement it will collapse to one of the two possible states, depending on the measurement basis. The most widely used bases are are:

- the computational basis $\{|0\rangle, |1\rangle\}$,
- and the Fourier basis $\left\{\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)\right\}$ also written as $\{|+\rangle, |-\rangle\}$ respectivelly.

A Hilbert space of two qubits is a four-dimensional tensor product of individual particle Hilbert spaces and the elements of the computational basis are $|00\rangle, |01\rangle, |10\rangle, |11\rangle$[2]. This notation is easily generalized to the case of more dimensions. Elements of the Fourier basis for an N-dimensional system have the following form

$$|e_k\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |x_j\rangle \, e^{2\pi jk/N} \tag{1.1.2}$$

where $|x_j\rangle$ labels the elements of the computational basis.

At this point it can be useful to introduce the concept of entanglement. This concept has no classical counterpart. A global state of a composite system is entangled if it cannot be written as a product of states of individual subsystems [HHHH09]. A simple example can be given in the case of two qubits, with the four famous Bell states or EPR pairs:

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{1.1.3}$$

$$\left|\Phi^-\right\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) \tag{1.1.4}$$

$$\left|\Psi^+\right\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) \tag{1.1.5}$$

$$\left|\Psi^+\right\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) \tag{1.1.6}$$

More detailed discussion about entanglement and related effects will be made in the following sections (1.3, 1.4, 1.5).

The next part of a quantum circuit to be described is the counterpart of a classical gate. In quantum circuits gates are represented by unitary operations. A unitary operation is bijective, length-preserving and always allows for a definition of the reverse operation (invertible). All unitaries U satisfy the condition $UU^\dagger = \mathbb{1}$, where $U^\dagger$ is the Hermitian conjugate of U. The commonly used single-qubit unitaries (gates) are the Pauli operations (X, Y, Z and $\mathbb{1}$), Hadamard (H), phase gate (S) and $\pi/8$ gate (T). These gates can be represented in the matrix form:

$$\begin{gathered} X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}; \qquad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}; \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}; \qquad \mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\ H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}; \qquad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}; \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix} \end{gathered} \tag{1.1.7}$$

In the computational basis X acts as a logical NOT or a bit-flip, while Z acts as a phase-gate, when acting on $|1\rangle$ it introduces the phase -1. In the Fourier basis these gates act in a reverse way (X as a phase gate, Z as a bit-flip). The Hadamard gate can be seen as a conversion from the computational to the Fourier basis and vice versa. Every single-qubit gate can be

[2] For the multiqubit states the shortened notation will be used: $|ab\rangle = |a\rangle \otimes |b\rangle$

represented as a rotation on the Bloch sphere. Furthermore, using the Euler decomposition, any rotation can be decomposed into three rotations around two mutually orthogonal axes. The form of the rotations around $\hat{x}, \hat{y}$ and $\hat{z}$ axes are given in the following equation:

$$U_x(\theta) = \begin{pmatrix} \cos\theta/2 & -i\sin\theta/2 \\ -i\sin\theta/2 & \cos\theta/2 \end{pmatrix} \qquad U_y(\theta) = \begin{pmatrix} \cos\theta/2 & -\sin\theta/2 \\ \sin\theta/2 & \cos\theta/2 \end{pmatrix} \qquad U_z(\theta) = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{-i\theta/2} \end{pmatrix} \tag{1.1.8}$$

Not many things can be done with single qubit gates. Much more interesting things, moreover everything that we need for quantum computing, can be done if two-qubit gates can be performed. The commonly used and the most useful two-qubit gates are the so-called controlled gates. These gates perform the certain unitary on the second qubit (denoted as the target) conditioned on the state of the first qubit (denoted as the control). One example is the controlled-NOT or $CNOT$, for which as its name suggests, the applied unitary is the bit-flip (Pauli X in the computational basis). Its matrix representation in the computational basis is:

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{1.1.9}$$

Other controlled two-qubit gate, whose application will be given in the next section, is the controlled-Z or simply CZ, which introduces phase-flip of the target in case the control qubit is in the state $|1\rangle$. Its matrix representation in the computational basis is given in the following equation:

$$CZ = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \tag{1.1.10}$$

The concept of controlled gates is easily generalized to the case of many qubits. Just one qubit is the target, while all the others are control qubits, with a gate performed only in the case all of them being in the state $|1\rangle$.

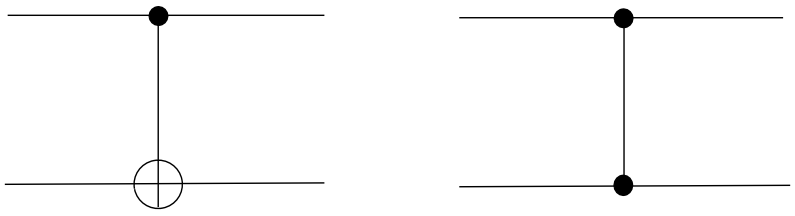


Figure 1.2: The symbols for the gates CNOT and CZ are on the left-hand and righ-hand side of the figure, respectively

When talking about quantum circuits the question that naturally arises is: what is the universal set of gates, sufficient to perform any quantum circuit? It has been proven that any

gate can be constructed from $CNOT$ and some number of single qubit gates. That means that for performing a quantum circuit on an arbitrarily large number of qubits we need just two-qubit and single-qubit gates. In their textbook Nielsen and Chuang give the following universal set of gates $\{H, S, CNOT, T\}$ [NC10], where T is the Toffoli gate, acting on 3 qubits. Boykin et al. showed that already $\{H, S, CNOT, T\}$ is good for universal quantum computing [BMP$^+$00]. An interesting result was obtained by Shi, namely the set $\{CNOT, K\}$, where K is a single qubit real gate such that K^2 does not preserve the computational basis, is a universal set of gates [Shi03]. This last

1.2 Quantum algorithms

A natural motivation for the building of a quantum computer would be its ability to solve some problems faster than a classical computer does or, even better, to solve some problems unsolvable with the use of just classical resources.[3] The improvement in the time necessary to solve some particular problems was given in historically important algorithms developed by Deutsch, Deutsch-Josza, Simon and Grover, but the tour de force of quantum computing is Shor's algorithm for the factorization of prime numbers.

Before giving a brief overview of the most famous quantum algorithms, it is good to explain the concept of quantum oracle due to its important role in these algorithms. A quantum oracle U_f for a specific function f, as used here is a black box which when given the input $|x\rangle |y\rangle$ returns the output $|x\rangle |y \oplus f(x)\rangle$. If the input is $|e_0\rangle |0\rangle$, where $|e_0\rangle = \frac{1}{\sqrt{N}} \sum_x |x\rangle$ is the first element of the Fourier basis, the output of the oracle will be $\frac{1}{\sqrt{N}} \sum_x |x\rangle |f(x)\rangle$. If this state is measured in the Fourier basis just one call of the quantum oracle is sufficient to obtain some relation between many values of f. The other important concept for further discussion is the query complexity of a certain problem. In the quantum oracle scenario it corresponds to the number of times we need to call the oracle in order to solve the problem. Classically it corresponds to the number of times one needs to do any calculation (potentially using some classical oracle).

- *Deutsch algorithm (1985)* [Deu85]. This was the first algorithm which showed the existence of a problem that a quantum computer can solve faster than a classical one. Because of that the algorithm bears a certain historical significance, but the problem itself doesn't have any practical significance. Namely, the algorithm is checking if some function $f : \{0,1\} \to \{0,1\}$ is constant or not. A classical computer would have to perform two operations to solve the problem: first to compute $f(0)$ and then $f(1)$. This means that the classical query complexity for this problem is 2. A quantum computer can just apply one quantum oracle to the input $|+\rangle |-\rangle$. If $f(0) = f(1)$ the output would be $(-1)^{f(0)} |+\rangle |-\rangle$, while in the case when $f(0) \neq f(1)$ the output is $(-1)^{f(0)} |-\rangle |-\rangle$. Thus, a readout in the Fourier basis alone distinguishes these two cases and the quantum query complexity for this problem is 1.

- *Deutsch-Josza algorithm (1992)* [DJ92]. Richard Josza and David Deutsch adopted Deutsch algorithm to a somewhat more complicated problem. In this case there is a function $f : \{0,1\}^n \to \{0,1\}$ which is either constant (all 2^n inputs give the same output value) or balanced (the number of inputs that give the output 0 is equal to the number of inputs that give the output 1). The problem is to determine if the function is constant or balanced. In order to deterministically solve the problem a classical computer would have to compute function f for $2^{n-1} + 1$ different inputs, so the classical query complexity for this problem is $2^{n-1} + 1$. A quantum computer can apply a quantum oracle for the input

[3] Though the latter is not likely to happen because it would violate the Church-Turing hypothesis.

$|e_0\rangle|-\rangle$, where $|e_0\rangle = \frac{1}{\sqrt{2^n}}\sum_{x=1}^{2^n}|x\rangle$ is the first element of the Fourier basis. The output of the quantum oracle will be:

(a) $(-1)^{f(0)}|e_0\rangle|-\rangle$ if the function is constant;
(b) $(-1)^{f(0)}|e_1\rangle|-\rangle$, where $|e_1\rangle = \sum_{x=1}^{2^n}(-1)^{f(x)}|x\rangle$, if the function is balanced.

The states $|e_0\rangle$ and $|e_1\rangle$ are orthogonal so for determining if the function is balanced or constant it is sufficient to measure the output in the Fourier basis. Since the quantum oracle was called just once the quantum querry complexity for this problem is equal to 1. In this case a quantum computer shows exponential improvement over a classical one.

- *Grover's quantum search algorithm (1996)* [Gro96, Gro01]. In his original paper Lov Grover indicates that the algorithm he presents can be used for a search of a particular phone number from a phone directory containing N phone numbers. Classically one would have to check number by number and at worst case $N-1$ queries would be needed for finding the right one. Quantum operations, as Grover points out, enable one to check multiple entries "at once" and with adjusting phases make the successful searches interfere constructively. The algorithm enables a quantum computer to find the right entry in $O(\sqrt{N})$ queries, representing quadratic improvement over a classical machine. To formalize the problem assume that there is a function $f : \{1, ..., N\} \to \{0, 1\}$, but just one of N inputs gives the output 1. The problem is to find the exact input $|x_1\rangle$ giving the output 1. The algorithm starts on the same way as previous, by preparing the input $|e_0\rangle|-\rangle$ and applying the quantum oracle. Again, the output of the oracle will be $\frac{1}{\sqrt{2}}\sum_{x=1}^{N}(-1)^{f(x)}|x\rangle|-\rangle$. This state can be seen as the action of the reduced oracle V_f on the input qubit $|e_0\rangle$, reduced oracle being $V_f = \mathbb{1} - 2\sum_{x:f(x)=1}|x\rangle\langle x|$. After this first reduced oracle another one is applied: $W = \mathbb{1} - 2\sum_{x:f(x)=1}|e_0\rangle\langle e_0|$. Successive applications of V_f and W rotate the originally input state $|e_0\rangle$ towards the wanted state $|x_1\rangle$. The process is called the amplitude amplification and the number of applications of the pair (V_f, W) necessary for rotating $|e_0\rangle$ as close as possible to $|x_1\rangle$ is $k = \left\lfloor \frac{\pi}{4}\sqrt{N} \right\rfloor$[4]. After k applications of the pair of reduced oracles (V_f, W) the output is measured in the computational basis, the readout being the desired x_1. It has been proven that Grover's algorithm is optimal and that quantum search cannot be done with a smaller number of querries [Zai97]

- *Simon's algorithm (1994)* [Sim94]. This algorithm solves an artificial problem but its significance lies in the fact that it is the precursor for Shor's algorithm. Suppose there is a function: $f : \{0,1\}^n \to \{0,1\}^n$, such that $f(x) = f(y)$ if and only if $y = x \oplus s$ where $s \in \{0,1\}^n$ ($\oplus$ represents addition modulo 2). Simon's algorithm deals with the problem of finding s. It is a period-finding algorithm, since s is a period of the function f. Classically this problem is very difficult since one would have to randomly choose vectors $x \in \{0,1\}^n$ and calculate $f(x)$ until finds two that give the same output. Probabilistically the function has to be called $O(2^{n-1})$ times in order to solve the problem. Simon's algorithm enables finding the period s in n steps, which represents an exponential improvement. The first step is to prepare the state $|e_0\rangle_A|0\rangle_B$. After calling the oracle U_f the output will be $\frac{1}{\sqrt{2^n}}\sum(|x\rangle + |x+s\rangle)_A|f(x)\rangle_B$. If the output B is measured in the computational basis, the output A will collapse to the state $\frac{1}{\sqrt{2^n}}\sum(|x\rangle + |x+s\rangle)$. In the next step each qubit of the output A is measured in the Fourier basis. The result of this measurement gives one relation between the elements of vector s. Repeating this procedure n times gives n equations with n unknown variables (elements of s), so the vector s can be uniquely determined.

[4] $\lfloor x \rfloor$ is the "floor" function of x and it returns the biggest integer smaller or equal to x

- *Shor's algorithm (1994)* [Sho94, Sho97]. The RSA public key crypstosystem relies on the fact that the factorization of a large number is classically a very difficult problem. There are no classical algorithms factoring large numbers efficiently and the best classical query complexity for factoring a number n is $O(\exp(n^{\frac{1}{3}}\log^{\frac{2}{3}} n))$. In 1994, Peter Shor developed an algorithm for probabilistic factorization of the number n in $O(n^2 \log n \log\log n)$ queries giving exponential improvement. The algorithm consists of a classical and quantum part. The classical part does the reduction of a factoring problem to a period-finding problem, which, as shown above, can be efficiently solved using a quantum computer. The algorithm consists of the following steps [NC10]:

  (a) Check if the number is even and return 2 if it is;

  (b) Check if it is an exponential of some number s and return that number if it is;

  (c) Randomly choose positive a smaller than N and find $\gcd(a, N)$ (greatest common divisor). If $\gcd(a, N) = 1$ return a;

  (d) *quantum part* Use quantum algorithm to find the period of function $a^x \mod N$;

  (e) If r is even and $a^{r/2} \neq -1 \pmod N$, then check if $\gcd(a^{r/2}-1, N)$ and $\gcd(a^{r/2}+1, N)$ are the factors. If not, algorithm fails and it has to be repeated.

1.3 Graph states

In this section one important class of quantum states, the so-called graph states will be introduced. These states have been shown to be useful in a handful of applications, primarily in quantum computing [BR01], quantum error correction [SW02, Sch02] and exploring of multipartite entanglement [CL07, HDB05].

The underlying structure of a graph state is, naturally, a graph.

- **Definition 1.1.** A graph G is composed of two sets:
  - a set V of vertices;
  - a set $E \subset V \times V$ of edges, such that $(u, u) \notin E$ and $(v, u) \in E$ if $(u, v) \in E$.

Two vertices u and v are adjacent if $(u, v) \in E$. The neighbors of a vertex are all vertices adjacent to it. A set of neighbors of the vertex v is labeled with N_v. An adjacency matrix A of a graph G containing N labeled vertices is $N \times N$ matrix whose elements are $A_{ij} = 1$ if vertices i and j are neighbors and $A_{ij} = 0$ otherwise.

A graph is connected if any of its two vertices are the beginning and the end of the sequence of adjacent vertices. A cycle is a sequence of adjacent vertices starting and ending with a same vertex. If a set of vertices of some graph can be divided into two groups each of which has no two or more adjacent vertices in it, a graph is bipartite. It is easy to show that the set of all bipartite graphs is disjoint to the set of all odd cycles [Wes00].

In quantum information theory a graph state is related to a set of qubits, each representing a vertex of the corresponding graph. Every qubit is initially prepared in the state $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. The corresponding graph state is obtained by applying CZ gate to all pairs of qubits connected with an edge . For example, the graph state of just two connected qubits is

$$CZ[|+\rangle\,|+\rangle] = CZ[\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)] = \frac{1}{\sqrt{2}}(|0+\rangle + |1-\rangle) \tag{1.3.1}$$

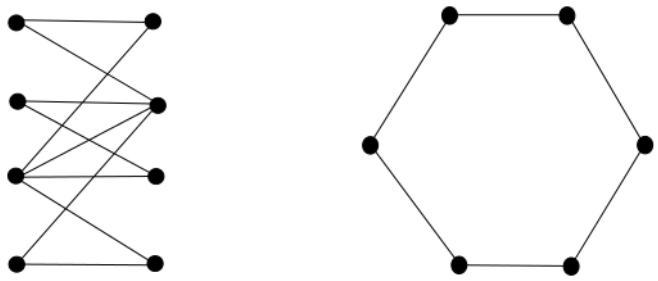


Figure 1.3: Connected graphs. On the left-hand side is a bipartite graph, while on the right-hand side is a cycle

which is locally equivalent to one of the Bell states. Thus, the CZ gate is used to create entanglement. Similarly the 3-qubit graph state (1-2-3) corresponds to the famous three-partite maximally entangled state, the GHZ state (Greenberger-Horne-Zeilinger) [GHZ89]:

$$CZ[|+\rangle |+\rangle |+\rangle] = \frac{1}{\sqrt{2}}(|0+0\rangle + |1-1\rangle) \tag{1.3.2}$$

In the literature the term cluster state sometimes is used interchangeably with the term graph state, but in more narrow sense cluster states represent a sub-class of graph states, whose qubit/vertex structure is a square lattice.

Graph states are also characterized by their stabilizer group. Stabilizer operator S_v for the the space $\mathtt{S}$ of states $|\psi\rangle$ is defined as:

$$S_v |\psi\rangle = |\psi\rangle , \qquad \forall |\psi\rangle \in \mathtt{S} \tag{1.3.3}$$

Stabilizers for a certain state form a group, since the product of two stabilizers is also a stabilizer. One stabilizer for a graph state in the form of tensor product can be generated by applying Pauli X operator on certain qubit, Pauli Z operators on each of its neighbors and identity to all the other (non-neighboring) qubits. Thus, the stabilizer group for a graph state is:

$$S_v = \left\{ X_v Z^{N_v} | v \in V \right\} \tag{1.3.4}$$

In the previous equation a certain notation rule is introduced. Namely, when some operator A acts on a set of qubits Q it is written as A^Q. Stabilizers form an Abelian group, since all the stabilizers commute. The correspondence between a graph state and a stabilizer group is unique.

In a laboratory a graph state can be obtained by the Ising-type interaction, that entangles two neighboring qubits. In this sense underlying graph can be seen as a summary of the interaction history of the particles. For more details on graph states see [HDE+05].

1.4 One-way quantum computation

A very important application of the introduced graph states is One-way quantum computing (1WQC) which is one example of Measurement-based quantum computing (MBQC in further text) [BR01]. The circuit model is not the only formalism describing quantum computation. One-way quantum computation is an alternative, in some ways better adjusted to the actual physical realization of quantum computation. A resource for 1WQC is a graph state, which is a highly entangled multipartite state. Information is written onto the graph state and in order to perform some operation each qubit of the graph state has to be measured in a certain basis. The result of a measurement can determine a base in which a measurement of the next qubit will be performed, so in this way measurements are temporarily ordered. Results of measurements are random, but if the whole state is entangled measurement results on it are correlated. This allows processing of an information throughout a graph state. The last remaining qubit will be in the state determined by the results of measurements performed on all previous qubits and followed by appropriate post-processing it represents the desired output. In this way different quantum computing operations are determined by the structure of underlying graph state and the sequence of performed measurements. The origin of the name "measurement-based" is apparent, while "one-way" comes from the irreversible nature of the computing process: a measurement done on qubits disentangles the graph state, making subsequent qubits collapse into the eigenstates of the measurement projector. There is one important distinction to be made between two types of qubits, those that are actually measured are called *physical qubits* or *cluster qubits*, and those that carry the information being processed are called *logical qubits*. It can be said that 1WQC is the process of teleportation of logical qubits.

The best way to explain the process is to give one simple example. Let us see how one of the widely-used gates in Quantum circuit model, the Hadamard gate, is modeled in 1WQC. The action to be simulated is:

$$\begin{aligned} \text{input:} \quad & |\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \\ \text{output:} \quad & H\,|\psi\rangle = (\alpha+\beta)\,|0\rangle + (\alpha-\beta)\,|1\rangle \end{aligned} \tag{1.4.1}$$

For that purpose we need the two-qubit graph state. The input state, $|\psi\rangle$, on which the Hadamard will act is encoded in the one of the qubits (qubit 1). The other qubit (qubit 2) is, as prescribed, in the state $|+\rangle$ and the CZ gate is applied in order to construct the graph state:

$$|\psi_{g2}\rangle = \frac{1}{\sqrt{2}}(\alpha\,|0+\rangle + \beta\,|1-\rangle) \tag{1.4.2}$$

In order to simulate the Hadamard gate it is enough to perform an X-basis measurement on qubit 1. The action of this measurement on the state can be written in the following way:

$$X_1\,|\psi_{g2}\rangle = \frac{1}{\sqrt{2}}\,|x_1\rangle\,(\alpha\,|+\rangle + (-1)^{x_1}\beta\,|-\rangle) \tag{1.4.3}$$

where x_1 is the result of the measurement of the observable X on qubit 1. When the result is 0 the measured qubit will collapse into the state $|+\rangle$, while the result 1 means that the qubit collapses to the state $|-\rangle$ (in further text s_i is the result of the measurement of the observable S on qubit i). After measuring qubit 1 in the X-basis, the state $|\psi\rangle$ originally encoded in qubit 1 is teleported to qubit 2 with the Hadamard applied. However, besides the desired Hadamard, there is an unwanted Pauli X in the case when the result of the measurement of qubit 1 is 1 (it collapsed to the state $|-\rangle$):

$$X_1\,|\psi_{g2}\rangle = |x_1\rangle \otimes X^{x_1}((\alpha+\beta)\,|0\rangle + (\alpha-\beta)\,|1\rangle) = |x_1\rangle \otimes X^{x_1}H\,|\psi\rangle \tag{1.4.4}$$

Pauli operators whose action depends on the previous measurements are unavoidable in 1WQC (they appear due to the randomness of the measurement process). Thus, 1WQC requires postprocessing to cancel-out the action of these, so-called, byproduct operators.

With the explained simulation of the Hadamard gate development of the simulation of some other important gates is much easier. In the following text simulations of the identity, rotation around z-axis and a general rotation on a cluster state will be presented.

- A very small change is required for the simulation of the rotation around z-axis. It can be done on the same two-qubit graph state, but qubit 1 has to be measured in a different basis. If the rotation $U_z(\phi)$ for an angle ϕ around z-axis has to be performed the measurement basis of qubit 1 has to be $(1/\sqrt{2}(|0\rangle \pm e^{(i\phi)} |1\rangle))$ where ϕ is a real parameter. After measuring qubit 1, qubit 2 will be in the state:

$$\alpha |+\rangle + (-1)^{x_1} e^{-i\phi} \beta |-\rangle \tag{1.4.5}$$

  where $x_1 = 0$ corresponds to collapsing of qubit 1 into the state $\frac{1}{\sqrt{2}}(|0\rangle + e^{i\phi} |1\rangle)$, while $x_1 = 1$ corresponds to collapsing into the perpendicular state $\frac{1}{\sqrt{2}}(|0\rangle - e^{i\phi} |1\rangle)$. However, the state of qubit 2 has coefficients corresponding to the z-rotation but in the "wrong", Fourier basis. The applied gate obtained upon the measurement of qubit 2 is:

$$X^{x_1} H U_z(\phi) |\psi\rangle \tag{1.4.6}$$

  This result has a somewhat more complex structure in comparison to the simulation of the Hadamard gate, because besides the byproduct operator X whose action depends on the result of the measurement of qubit 1, an additional Hadamard appears regardless of the measurement result. Of course that adequate postprocessing would cancel-out this Hadamard, but if one wants, an 1WQC simulation of $U_z(\phi)$ gate without Hadamard can be done by utilizing a linear graph state consisting of 3 qubits, where qubit 1 is measured in the basis $(\frac{1}{\sqrt{2}}(|0\rangle \pm e^{(i\phi)} |1\rangle))$, qubit 2 in the X-basis and qubit 3 is the output qubit measured in the Z-basis. Of course that this simulation introduces another byproduct operator (Z in this case) which acts on the logical qubit conditionally on the measurement result of qubit 2. To summarize, if a 3-qubit graph state is used to simulate $U_z(\phi)$, the gate that is actually performed is

$$X^{x_2} Z^{z_1} U_z(\phi) |\psi\rangle \tag{1.4.7}$$

- The simulation of Hadamard and the alternative 3-qubit simulation of $U_z(\phi)$ can give intuition for applying the identity gate. The additional Hadamard in the 2-qubit simulation of $U_z(\phi)$ was canceled-out by adding one qubit before the output and measuring it in the X-basis. One qubit measured in the X-basis before measuring the output qubit simulates exactly Hadamard, so it can be considered that the 3-qubit $U_z(\phi)$ simulation consisted of patched gates, one performing $U_z(\phi)$ with unwanted Hadamard and the other performing one more Hadamard which gives $H^2 U_z(\phi) = U_z(\phi)$. This gives the idea that the Hadamard applied twice will give the identity, and that is exactly how it is done. On the graph state of 3-qubits, the first two are measured in the X-basis which gives $H^2 = I$ acting on a logical qubit.

- One more result can be developed from the simulation of $U_z(\phi)$ and it is 1WQC performing a general rotation (meaning any one-qubit unitary). It is a widely known result that any rotation in the Euler representation can be decomposed as a product of three rotations around two perpendicular axes: $U_z(\gamma)U_x(\beta)U_z(\alpha)$. Having this in mind any general

rotation can be simulated on the 5-qubit linear graph state with qubits measured in the following bases:

$$\begin{aligned} &1 : X \\ &2 : (\frac{1}{\sqrt{2}}(|0\rangle \pm e^{i\alpha}|1\rangle)) \\ &3 : (\frac{1}{\sqrt{2}}(|0\rangle \pm e^{i\beta}|1\rangle)) \\ &4 : (\frac{1}{\sqrt{2}}(|0\rangle \pm e^{i\gamma}|1\rangle)) \\ &5 : Z \end{aligned} \tag{1.4.8}$$

This can be seen as a decomposition in which measurements on qubits 1 and 2 are teleporting the action of $U_z(\alpha)$ without Hadamard, qubits 3 and 4 rotations $U_z(\beta)$ and $U_z(\gamma)$, respectively, with Hadamard. Thus $U_z(\beta)$ is sandwiched between two Hadamards which represents $U_x(\beta)$ (the following identity can easily be checked: $HU_z(\beta)H = U_x(\beta)$).

In previous discussion it was implicitly assumed that the simulation of a gate that can be decomposed as a product of elementary unitary transformations is performed just as a succession of simulations of elementary unitaries on a linear graph state. It completely follows the logic of 1WQC, because a computation is done as a teleportation of logical qubit(s) and on a linear graph state this teleportation is trivial. A solution that would use a different underlying graph, for example a tree-graph (Fig. 1.4) with multiple inputs would not do the job, because after performing the first unitary by measuring one of the inputs, the graph state would be changed and furthermore all subsequent measurements would "overwrite" the teleported logical qubit. The linear graph, in the other way, just patches all elementary unitary transformations. The output of the first elementary unitary operation is the input for the subsequent and the sequence continues until the full gate is simulated, by a final measurement of the output qubit.

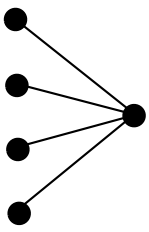


Figure 1.4: The tree-graph

The explained procedure is easily generalized to the case of any quantum circuit consisting of an arbitrary number of quantum gates. A circuit is simulated by simply patching all the gates. In the first section it has been pointed out that a universal set of gates must include two-qubit gates, such as $CNOT$. So, for doing universal quantum computing in 1WQC these gates must be incorporated in the process. This is done, and just the procedure for performing $CNOT$ in 1WQC will be given here, for the proof and the detailed description are out of the scope of this introduction [RBB03]. For the realization of the $CNOT$ gate a graph state consisting of 15 qubits is needed (Fig. 1.5). The graph state is prepared in a standard way, encoding input in qubits 1 (control) and 9 (target) and then performing CZ on qubits that are connected (1-2, 2-3, 3-4, 4-5, 5-6, 6-7, 4-8, 8-12, 9-10, 10-11, 11-12, 12-13, 13-14, 14-15). The output qubits

are 7 and 15. Qubits 1, 9, 10, 11, 13 and 14 are measured in the X-basis, while 2, 3, 4, 5, 6, 8 and 12 are measured in the Y-basis. The output qubits are, as usual, measured in the computational Z-basis. Since the realization of the $CNOT$ includes 13 measurements before the output, the byproduct operator will consist of up to 13 Pauli operators acting on logical qubits, depending on results of all the measurements. The byproduct operator acting on the control output is $X^{y_2+y_3+y_5+y_6}Z^{x_1+y_3+y_4+y_5+y_8+x_9+x_{11}+1}$, while $X^{y_2+y_3+y_8+x_{10}+y_{12}+x_{14}}Z^{x_9+x_{11}+x_{13}}$ will act on the target qubit. For the details see [RBB03].

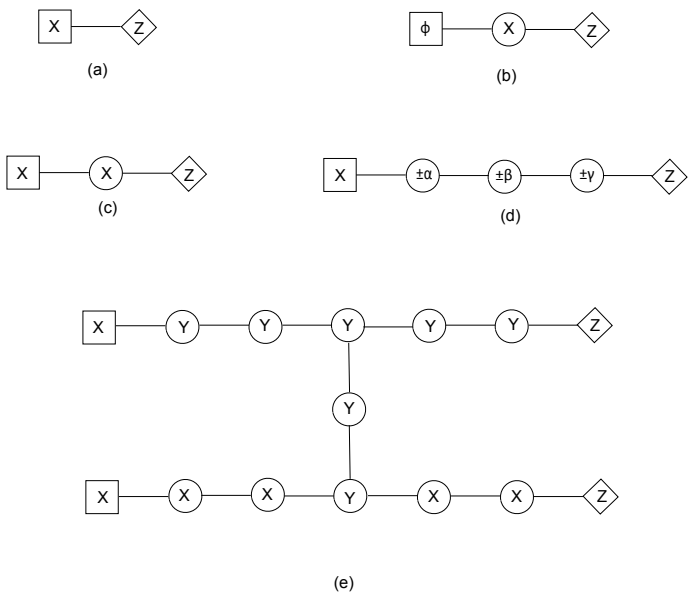


Figure 1.5: The symbols for the implementation of different gates in 1WQC. The square represents the input qubit and the diamond is the output qubit. The gates on the figure are: (a) Hadamard; (b) rotation around z-axis; (c) identity; (d) arbitrary rotation; (e) CZ

In the previous paragraphs the procedure for the realization of the universal set of gates on a cluster state is given. Still, a few clarifications are left to be made before ending this brief description of the principles of 1WQC. The fact that this model of computation is based on the measurement which in quantum mechanics has random outcomes manifests in the appearance of byproduct operators (Pauli X and/or Z) acting after the each realized gate. In order to perform a successful realization of any quantum circuit or even some non-trivial rotation the explained patching of simple unitaries has to be done. As byproducts act after each unitary, the post-processing can be performed just if all byproduct operators are propagated to the end of the simulated circuit. This propagation can be very simple if the realized unitaries belong to the Clifford group. Operators from the Clifford group map an operator from the Pauli group to the same or different Pauli operator. If a Clifford operator is C and a Pauli P, the following relation is satisfied:

$$CP = P'C \tag{1.4.9}$$

General rotations do not belong to the Clifford group and are alterred upon the propagation of byproduct Pauli operators:

$$U_s(\alpha,\beta,\gamma)X = XU_s(\alpha,-\beta,\gamma) \tag{1.4.10}$$

$$U_s(\alpha,\beta,\gamma)Z = ZU_s(-\alpha,\beta,-\gamma) \tag{1.4.11}$$

With this in mind it can be concluded that measurements in bases of operators belonging to the Clifford group can be done at the same time, but measurements not corresponding to a

Clifford operation have to be performed in a basis that depends on the outcomes of previous measurements. All computations involving a realization of at least one measurement not belonging to the Clifford group in 1WQC model are time-ordered and adaptive. In the circuit model a computation is performed in a number of steps in time that corresponds to a number of gates constituting the circuit. In 1WQC all measurements can be done at the same time if a computation involves only Clifford operations.

Before starting a short discussion about actual realization of 1WQC in a laboratory, we will give some useful remarks about the simplest measurements, those performed in X and Z bases. Measuring in the Z-basis effectively takes the qubit out of the graph state and applies Z to neighboring qubits if the measured qubit is projected into the state $|1\rangle$. Measurements in X or Y bases take a role of quantum wires for propagation of a logical qubit through the graph state. Also, computation on a linear graph state (or any linear subgraph) will not be altered if an even number of qubits is added and measured in the X-basis (as pointed out this represents the identity gate). This will be important in further sections.

It is already mentioned that one advantage of the 1WQC is that the way it is performed resembles the actual implementation in a very straightforward way. First, the set of physical objects with two different states is necessary for qubits. Besides that, an interaction able to couple these two states and to introduce a phase difference of π for the role of CZ gate is needed. It should be possible to switch on and off this interaction depending on the underlying graph structure [BB06]. An optical lattice might be used for the construction of a quantum computer performing 1WQC. Essentially it is a standing wave formed by interference of two counterpropagating laser fields. The periodic potential formed in this way can be used for the trapping of cold atoms. Depending on its internal state ($|0\rangle$ or $|1\rangle$) atoms will see different potentials with shifted maxima and minima and thus different optical lattices. By moving the lattice that $|1\rangle$ sees it can acquire the phase -1 only when it interacts with the other cold atom which is in the state $|0\rangle$. This effect can be used for the realization of CZ gate. Optical lattices are good because CZ can be applied on many neighboring atoms simultaneously, resulting in the preparation of a quite big cluster state. It has been proven that a cluster state can be used for universal MBQC, by measuring redundant qubits in the Z-basis and producing a graph state necessary for the computation. Authors pointed out, up to now, insurmountable obstacle for using optical lattices in the process of MBQC and that is inability of measuring and manipulating single atoms. Other possible solutions are linear optics [Kok07] and cavity QED implementation of the one-way quantum computer [BK05].

1.5 Nonlocality

For future discussion of self-testing and interactive proofs the concept of nonlocality is of the particular importance. Nonlocality is the phenomenon that does not have a counterpart in classical theory. The discussion about nonlocality and its place in quantum theory started almost 80 years ago. Albert Einstein had objections to the Copenhagen interpretation of quantum mechanics. After some failed attempts to show that the interpretation of the theory is wrong he tried to prove that it is incomplete. In their seminal paper "Can Quantum-Mechanical description of physical reality be considered complete?" published in 1935 [EPR35] Einstein, Podolsky and Rosen used an entangled state to prove incompleteness of quantum mechanics. Based on the paradox they presented in the paper, the existence of some kind of local hidden variable was proposed. The knowledge of this hidden variable determines the outcome of an experiment and rules out the probabilistic nature of quantum mechanics. It is due to ignorance of a hidden variable that quantum mechanics cannot give better predictions than those on a probabilistic level. A theory that would describe hidden variables would give deterministic predictions on

outcomes of measurements. According to the arguments used in the famous EPR paper two properties are necessary for a complete theory and they showed that in quantum mechanics one excludes another, thus it must be incomplete. The first property of a good theory aspiring to be complete is locality, meaning that outcome of one measurement cannot affect outcome of the other spatially separated measurement. The other one is that the theory has to be realistic (sometimes referred to as counterfactual-definite), meaning that properties of a system exist independently of a measurement, they are predetermined. Einstein himself described nonlocality as "the spooky action on distance" while realism he described as a belief that "the Moon exists even when we are not looking at it" [EPR35, Pai79].

John Bell started from these two assumptions and derived the famous Bell's theorem which allows to formulate a Bell inequality that must be satisfied in every local and realistic theory, often called a local hidden variable theory (LHV) [Bel04]. The main consequence of the paper is that quantum mechanics is either nonlocal or nonrealistic due to the fact that an entangled state violates the Bell's inequality. There are many different Bell's inequalities, corresponding to different systems. One very simple derivation of one particular Bell inequality is proposed by Maccone [Mac13]. Suppose Alice and Bob every day get from Charlie the identical gift: the set of three marbles (denoted by X, Y and Z) in a nontransparent box. Each marble can be black or white, and Alice and Bob can at a time play with just one marble. They both choose one of the three marbles (X, Y and Z) from their boxes and compare the color. Since they like to play with marbles but also like statistics after a year of playing they formed a set of probabilities for getting the marble of the same color. They denoted as $P_{same}(X, Y)$ the probability that when Alice takes marble X and Bob marble Y they will play with two marbles of the same color. $P_{same}(X, Z)$ is the probability that marbles X and Z will have the same color and so on. Since they get the same gift every day it is clear that $P_{same}(X, X) = P_{same}(Y, Y) = P_{same}(Z, Z) = 1$. The Bell inequality gives the relation for other three probabilities and it is:

$$P_{same}(X, Y) + P_{same}(X, Z) + P_{same}(Y, Z) \geq 1 \tag{1.5.1}$$

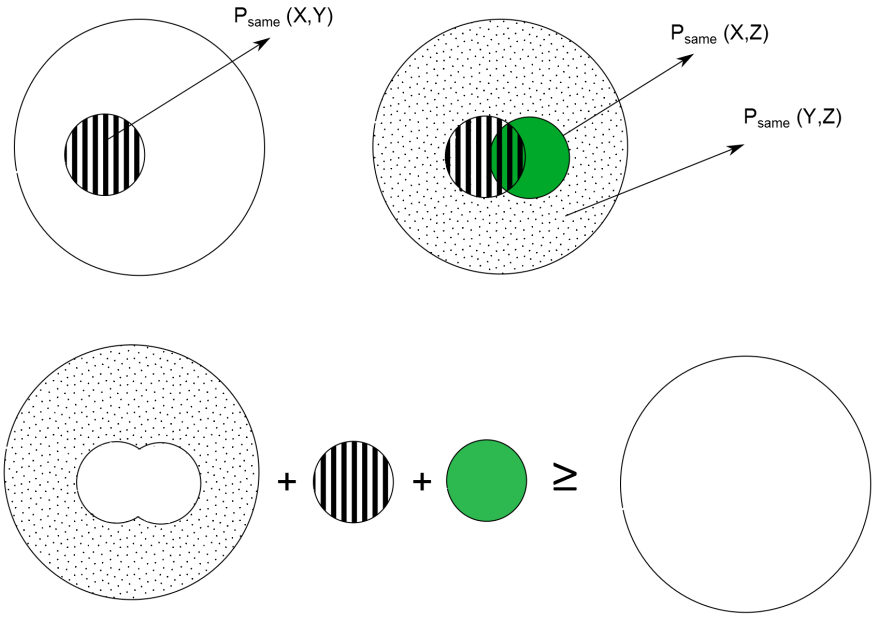


Figure 1.6: A graphical representation of the Bell inequality (1.5.1). The area of the every big circle is 1. The area of the striped circle is $P_{same}(X, Y)$, of the green one $P_{same}(X, Z)$, while the dotted part of the circle has the area $P_{same}(Y, Z)$.

This inequality can be proven very easily by using Venn diagrams. There are 8 combinations of marbles: (white, white, white), (black, white, white), (white, black, white), (white, white, black), (white, black, black), (black, white, black), (black, black, white) and (black, black, black). On the 1st Venn diagram (Fig. 1.6) the area of the striped smaller circle represents $P_{same}(X,Y)$, while the area of the bigger circle is equal to 1, since it corresponds to the case when X and Y have either same or different color. On the second diagram the area of the green circle represents $P_{same}(X,Z)$ and its intersection with the striped circle represents the case $P_{same}(X,Y,Z)$. The area of all that is in neither green, nor striped circle represents $P_{same}(Y,Z)$ because if X is not of the same color as nor Y, neither Z then Y and Z must be of the same color. On the lower part of the Fig. 1.6 we see that the inequality is satisfied and only assumed thing is that marbles had colors before Alice and Bob took them out of the boxes (realism) and that the color of the Bob's marble does not depend on the color of the marble Alice took out that day, and vice versa (locality).

In this classical game-measurement a box can be assumed to be an object and its properties are colors of the marbles. The quantum analog of this case would be a bipartite state with measurements of three properties with requirement that the same properties are measured by both parties. This can be achieved for entangled states, with one observable being measured in three different bases. The measurement bases are:

$$\begin{aligned} &\text{X basis:} \quad |x_0\rangle = |0\rangle\,, |x_1\rangle = |1\rangle \\ &\text{Y basis:} \quad |y_0\rangle = \frac{1}{2}|0\rangle + \frac{\sqrt{3}}{2}|1\rangle\,, |y_1\rangle = \frac{\sqrt{3}}{2}|0\rangle - \frac{1}{2}|1\rangle \\ &\text{Z basis:} \quad |z_0\rangle = \frac{1}{2}|0\rangle - \frac{\sqrt{3}}{2}|0\rangle\,, |z_1\rangle = \frac{\sqrt{3}}{2}|0\rangle + \frac{1}{2}|1\rangle \end{aligned} \tag{1.5.2}$$

The entangled state $\left|\Phi^+\right\rangle$ can be written in one of the forms

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}\left(|x_0\rangle|x_0\rangle + |x_1\rangle|x_1\rangle\right) = \frac{1}{\sqrt{2}}\left(|y_0\rangle|y_0\rangle + |y_1\rangle|y_1\rangle\right) = \frac{1}{\sqrt{2}}\left(|z_0\rangle|z_0\rangle + |z_1\rangle|z_1\rangle\right) \tag{1.5.3}$$

$P_{same}(X,Y)$ is defined as the probability to get results x_0 and y_0 or x_1 and y_1 when measuring in the X and Y bases, and analog for the other pairs. From the form of $\left|\Phi^+\right\rangle$ it is obvious that $P_{same}(X,X) = P_{same}(Y,Y) = P_{same}(Z,Z) = 1$. Other probabilities can be obtained in the following way:

$$P_{same}(X,Y) = \frac{1}{2}|\langle x_0|\ y_0\rangle|^2 + |\langle x_1|\ y_1\rangle|^2 = 1/4 \tag{1.5.4}$$

$$P_{same}(X,Z) = \frac{1}{2}|\langle x_0|\ z_0\rangle|^2 + |\langle x_1|\ z_1\rangle|^2 = 1/4 \tag{1.5.5}$$

$$P_{same}(Y,Z) = \frac{1}{2}|\langle y_0|\ z_0\rangle|^2 + |\langle y_1|\ z_1\rangle|^2 = 1/4 \tag{1.5.6}$$

so their sum is:

$$P_{same}(X,Y) + P_{same}(X,Z) + P_{same}(Y,Z) = 3/4 < 1 \tag{1.5.7}$$

Thus, some entangled states, that exist in nature (can be produced in a physical process) but as entities are described just in quantum theory, violate Bell inequalities. The Bell inequality used here was originally derived by Preskill [Pre13], but there are many more forms of Bell's inequalities that separate local and nonlocal theories. One of the most used Bell's inequalities is the one developed by Clauser, Horne, Shimony and Holt, known as the CHSH inequality [CHSH69]:

$$|S| = \left|E(A,B) - E(A,B') + E(A',B) + E(A',B')\right| \leq 2 \tag{1.5.8}$$

This inequality describes outcomes of spatially separated measurements on a bipartite state, performed by Alice and Bob, A and A' being measurement settings on the Alice's side and B and B' on the Bob's. The possible outcome of the measurements A, A', B, B' is 0 or 1. $E(A, B)$ is the correlation, giving the expectation value of the product of operators A and B, where the outcome is set to 1 if outcomes of A and B are the same and -1 if they are different. S is called the CHSH expression. In a similar way as it was done for the Bell's inequality derived by Preskill, it can be shown here that the Bell state violates the CHSH inequality. For example for $|\Phi^+\rangle$ (1.1.4) with an appropriate set of measurements the CHSH expression takes the value $2\sqrt{2}$. In 1980 Tsirelson proved that there is an upper bound for the value quantum correlations can exhibit and for the CHSH expression it is exactly $2\sqrt{2}$ [Tsi80]. For maximally entangled states the CHSH expression constructed for appropriate measurements will reach the Tsirelson bound (Fig.1.7). In the context of CHSH one clear distinction between nonlocal and LHV theories is the form of correlations resulting from performed measurements. In a LHV theory the probability that measurements X and Y on spatially separated subsystems yield outcomes a and b, respectively, can be written in the form:

$$p(a, b|X, Y) = \int d\lambda q(\lambda)\, p_\lambda(a|X)\, p_\lambda(b|Y) \tag{1.5.9}$$

where λ is the local hidden variable that might be random, so $q(\lambda)$ is its probability distribution. In nonlocal theories common probabilities of the form $p(a, b|X, Y)$ cannot be decomposed on a product of individual probabilities $p(a|X)$ and $p(b|Y)$.

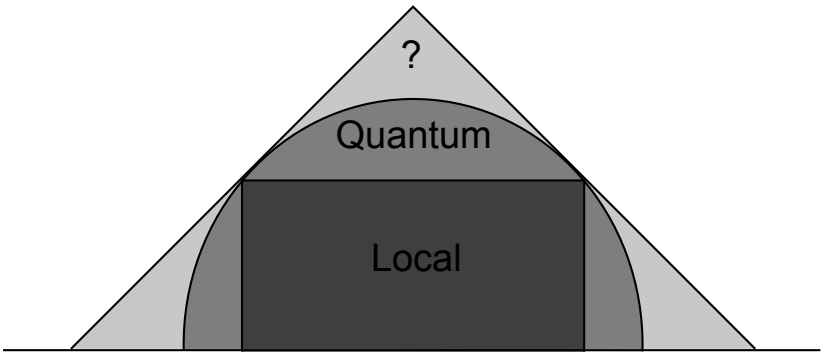


Figure 1.7: Classical correlations which do not violate the Bell inequalities are subset of the larger set of quantum correlations bounded by the Tsirelson bound. These quantum, non-local, correlations might be a subset of a larger set of correlations bounded only by the non-signaling principle

Today there are many experimental confirmations that nature allows the quantum mechanical violation of Bell inequality. The Bell test or Bell experiment must include a source of entangled particles (mainly photons). Two particles from an entangled pair propagate in different directions and are being measured in spatially separated laboratories. However, all these experiments contain some loopholes. The most important loopholes are the *detection* and *locality* loopholes.

- *Detection loophole.* Bell experiments are usually performed with photons, because of, compared to the other particles, fairly easy manipulation and detection process. Nevertheless the detectors used in laboratories are never perfect and not all photons involved in the experiment are being detected. The first thing one would do is to discard all the photons which are not detected and to calculate the measurement statistics based only

on the detected photons. This strategy involves the post-processing (discarding process) and it has been proven that by using post-selection one can fake the violation of a Bell's inequality while whole time operating in the purely local theory. [CH74, GLLL$^+$11, Pea70]

- *Locality loophole.* It is very important to preserve independence of measurement processes performed by two parties in a conventional Bell test. One has to be sure that there is no communication between parties that can introduce some kind of causality. This means that the measurement process has to be short enough that the signal from one party cannot travel to the other party and affect nor the choice of the measurement settings, neither the measurement outcome [Bel04].

Box 1.1 CHSH game

Nonlocal correlations can be examined through nonlocal games. One referee and two or more players play this game and players are rather cooperating than competing. The CHSH game is played between Eve (referee) and players Alice and Bob. Eve sends to Alice and Bob binary messages, x to Alice and y to Bob. By sending this messages Eve commands to Alice and Bob which measurement to perform on the qubit they both have in their possession. Alice and Bob report the obtained outcome via another binary message, a from Alice and b from Bob. Thus, in the CHSH game Eve communicates with both Alice and Bob, but two of them cannot communicate. The rule is that Alice and Bob win the game if

$$x \wedge y = a \oplus b \qquad x, y, a, b \in \{0, 1\} \tag{1.5.10}$$

If Alice and Bob have qubits that are classically correlated the best strategy for them is to always report 0. In that case the probability to win the game is 0.75. Indeed, $x \wedge y$ is equal to 0 in three out of four possible combinations. On the other side, it has been proven that if Alice and Bob hold an entangled pair of qubits the best strategy allows them to win the CHSH game with the probability 0.851 [CHSH69].

The optimal strategy for winning the CHSH game when Alice and Bob share an EPR pair, $|\psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + |11\rangle$, consists of measurements from the XZ-plane of the Bloch sphere. When Alice gets the input 0 she measures her qubit in the Z-basis, while upon getting the input 1 she measures her qubit in the X-basis. On the other hand, on the input 0 Bob responds by measuring his qubit in the $\frac{X+Z}{\sqrt{2}}$-basis, while on the input 1 he responds by measuring in the $\frac{X-Z}{\sqrt{2}}$-basis. It is straightforward task to calculate probabilities and make sure that this strategy allows Alice and Bob to win the CHSH game with probability 0.851.

In this section nonlocality for the bipartite states has been described. The situation gets much more complicated, and interesting at the same time, when it comes to multipartite states. The first idea for describing multipartite nonlocality would be adjusting the concept already developed for bipartite nonlocality. As one would expect there is no such one-to-one correspondence. In the bipartite case nonlocality can be achieved when measurement results obtained by one party depend on the measurement results obtained by the other party. For the multipartite case a novel form of nonlocality is certainly not exhibited by the states where measurement results of one party do not depend on any of the measurements results of the other parties. The question is if there is if there is some "different" nonlocality than the one manifested by just two

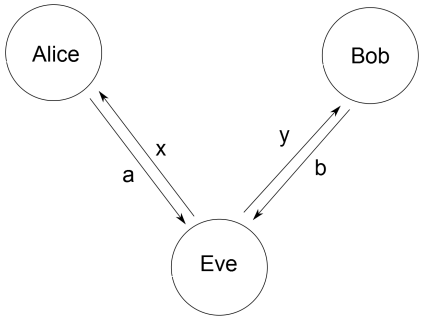


Figure 1.8: The CHSH game. The referee Eve exchanges the binary messages with Alice and Bob.

of all the parties. In 1987 Svetlichny introduced the concept of *genuine multipartite nonlocality* and derived an inequality for testing it [Sve87].

The genuine multipartite nonlocality for the tripartite case will now be explained and then the generalization for the n-partite case is straightforward. In the tripartite case where measurement settings of three parties are denoted as X, Y and Z and outcomes of the respective measurements as a, b and c the correlations $p\left(abc|XYZ\right)$ in a local model can be written as:

$$p\left(abc|XYZ\right)=\int d\lambda q\left(\lambda\right)p_{\lambda}\left(a|X\right)p_{\lambda}\left(b|Y\right)p_{\lambda}\left(c|Z\right) \tag{1.5.11}$$

where λ is a shared local hidden variable satisfying $\int d\lambda q(\lambda)=1$. If the correlations can be written in the form $p(ab|XY)p(c|Z)$ then when two parties are regarded as one the state can be written as a product, hence the third party is not correlated with the first two. If correlations between a and b are nonlocal the whole state is nonlocal. However, this kind of nonlocality is a little bit different than the one exhibited when all three parties are nonlocally correlated. In the aforementioned paper Svetlichny made a borderline. Genuine multipartite nonlocality is exhibited by states that cannot be written in the form:

$$\begin{aligned}p\left(abc|xyz\right)=&\int d\lambda q\left(\lambda\right)p_{\lambda}\left(ab|xy\right)p_{\lambda}\left(c|z\right)+\\&+\int d\mu q\left(\mu\right)p_{\mu}\left(bc|yz\right)p_{\mu}\left(a|x\right)+\\&+\int d\nu q\left(\nu\right)p_{\nu}\left(ac|xz\right)p_{\nu}\left(b|y\right)\end{aligned} \tag{1.5.12}$$

where $\int d\lambda q(\lambda)+\int d\mu q(\mu)+\int d\nu q(\nu)=1$. Physically this means that there is no bipartition across which correlations would be local, therefore all three parties must share some common nonlocal resource. Svetlichny introduced a Bell-type inequality whose violation implies the presence of the genuine multipartite nonlocality. Assume that three parties are involved and each party i can chose between two measurements to perform: X_i and Y_i. All the measurements can have two possible outcomes $a_i, b_i \in \{0,1\}$. Every probability distribution that can be written in the form (1.5.12) satisfies the following inequality:

$$<S_3>=<a_1a_2b_3+a_1b_2a_3+b_1a_2a_3-b_1b_2b_3+b_1b_2a_3+b_1a_2b_3+a_1b_2b_3-a_1a_2a_3>\leq 4 \tag{1.5.13}$$

This inequality can be written in terms of the CHSH expression for two parties [BBGL11], (due to the permutational symmetry it is not important which two):

$$S_3=Sb_3+S'a_3\leq 4 \tag{1.5.14}$$

where S is the CHSH expression while $S' = b_1b_2 + b_1a_2 + a_1b_2 - a_1a_2$ is obtained from S by permuting the choice of measurements. Equation (1.5.14) allows for the generalization of the Svetlichny inequality to the case of n parties [BBGL11]:

$$S_n = S_{n-1}b_n + S_{n-1}a_n \leq 2^{n-1} \tag{1.5.15}$$

1.6 Self-testing

Initial protocols for quantum key distribution (QKD) assumed trustworthy quantum devices (sources, measurement devices). In reality this could rarely be the case, so the conditions for trustworthiness were the weak point of those protocols. Devices can be prepared adversarially and do not necessarily work according to the specifications. Especially in the cryptographic scenario there is a big reason for all kinds of mistrust. However, it has been proven that this condition for some protocols can be relaxed. Device independent quantum key distribution can be performed even with adversarial preparation of quantum devices, and it relies on the correctness of quantum mechanics, with some protocols relaxing even this assumption [PAB$^+$09, ABG$^+$07]. This led to finding ways to describe other quantum communication and quantum computation processes in the context of adversarial quantum devices.

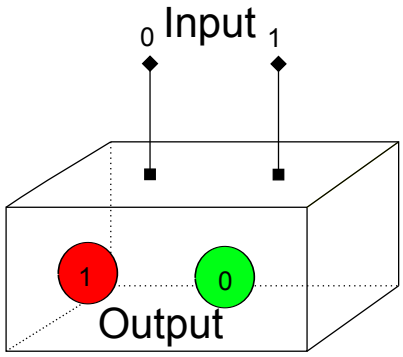


Figure 1.9: In self-testing a state and measurements are seen as a black box with a certain number of inputs. On this figure there are two outcomes for each input, and these outputs are modeled as light bulbs

We assume that a source of qubits with the specification is provided, as well as a set of devices claimed to perform specific measurements. However, devices are black boxes, and classical users performing computation, for example, do not have an insight into the actual nature of the source; it could be anything. *Self-testing* is a procedure that can be done in order to check if the given devices can be used for quantum computing or communication. Mayers and Yao [MY04] introduced self-testing of a singlet, a state equivalent to a maximally entangled pair of qubits, and basic set of measurements. After this pioneering work van Dam et al. [vDMMS00] developed a self-test for quantum gates, with assumption that dimension of the Hilbert space is known. Next step, the self-test of quantum circuits in a real Hilbert space was presented in the paper of Magniez et al. [MMMO06]. McKague and Mosca [MM10] extended the Mayers-Yao self-test on the measurements in the complex Hilbert space. Another important contribution is the McKague's self-test of graph states [McK11]. A bit different approach to self-testing can be found in the papers of Bancal et al. [BNS$^+$14], where the SWAP concept is used to achieve much

higher robustness compared to all earlier results and in the paper of Reichardt et al. [RUV13] that developes the self-test of the tensor product of many EPR pairs (more details in the next section).

After this short history of self-testing the more detailed description of the process will be given. As noted, the aim of the procedure is to confirm or deny that a source emits qubits that are in a state close to the claimed state and also that measurements performed are close to the specification. But, in the first place what is meant by "close to the state". Clearly, the self-test doesn't aim to confirm, for example, that a source emits precisely the EPR pair of photons, with entangled polarization. Even when photons are maximally entangled in terms of polarization, as physical objects they have more degrees of freedom. Thus, a successful self-test would be able to prove that a source emits qubits in a state that is a tensor product of a singlet and some state of arbitrary dimension that will be denoted as $|junk\rangle$. Here one very useful and important property of self-testing is implied: the dimension of a tested state is not assumed. This adds one more nice characteristic in that the self-test can always assume pure states; since dimension is not important, it is always possible to purify states by adding more dimensions [NC10]. To return to the "closeness" of the states; it is difficult to possess a singlet state, even assuming that it is in a tensor product with some multidimensional irrelevant state. This means that a good self-test will always involve some kind of robustness. In other words, it should be constructed on such a way that the claimed state $|\psi\rangle$ is not the only one that passes the test but also any state $|\phi\rangle$ satisfying $\| |\psi\rangle - |\phi\rangle \| < \epsilon$, where ϵ is some positive number close to 0.

For more detailed description of the self-testing we will need the following definitions:

- **Definition 1.2.** The term *reference experiment* refers to the specification of black-box that should be used, say, for some quantum computation. A reference experiment involves the supposed state $|\psi\rangle$ and the set of local measurement observables $M = \{A_1, A_2, ...\}$.

- **Definition 1.3.** The term *physical experiment* refers to how the black-box actually operates, including the actual state of the system $|\psi'\rangle$ and the actual measurements being performed $M' = \{A'_1, A'_2, ...\}$.

- **Definition 1.4.** An isometry is a linear operator $\Phi : A \to B$ that preserves the inner products. A local isometry on n subsystems is an isometry that can be written as a tensor product of n isometries acting on the individual systems:

$$\Phi = \Phi_1 \otimes \Phi_2 \otimes ... \otimes \Phi_n \tag{1.6.1}$$

Two crucial concepts in self-testing theory are those of *simulation* and *equivalence*.

- **Definition 1.5.** The physical experiment simulates the reference experiment if $\|A'_i |\psi'\rangle\| = \|A_i |\psi\rangle\|$ for every operator $A_i \in M$ and $A'_i \in M'$.

According to this definition simulation is a concept that refers to probabilities produced when certain measurements are performed.

- **Definition 1.6.** The physical experiment is equivalent to the reference experiment if there exists a local isometry Φ such that

$$\Phi(|\psi'\rangle) = |junk\rangle \otimes |\psi\rangle \tag{1.6.2}$$

$$\Phi(A'_i |\psi'\rangle) = |junk\rangle \otimes A_i |\psi\rangle \tag{1.6.3}$$

for all $A_i \in M$ and $A'_i \in M'$.

The aim of a self-test is to prove the equivalence of a physical and a reference experiment. A simulation in general case does not guarantee that quantum devices can be used and trusted, but existence of a local isometry between an actual and a specified state is a clear sign that results obtained by performing measurements on the physical experiment can be trusted. Since, probabilities produced upon measurement are the only thing a classical user can obtain during the self-testing procedure the idea is to find conditions under which it can be proved that simulation implies equivalence. A proof should culminate in a construction of the isometry. In many cases this isometry relies on the some kind of SWAP gate. A general SWAP gate can be constructed of two anti-commuting observables, usually physical counterparts of reference Pauli operators X and Z. To return one step backwards, these anti-commuting relations should be deduced from measurement probabilities.

Since the equivalence is deduced from probabilities, attention has to be paid to all the transformations of the reference experiment that preserve probabilities [McK11]. These transformations are:

- Local changes of basis.
- Additions of arbitrary number of ancillas.
- Changing the action of the observables outside of the support of the state.
- Locally embedding the state and operators in a larger (smaller) Hilbert space.
- Complex conjugation of all states and operators comprising the physical experiment.

All of these transformations have to be taken into account when performing a self-test, but the last one can cause real problems, about which there will be some more discussion in the following chapters.

The number of assumptions about a tested system incorporated in a self-test should be as small as possible. Here, it is already mentioned that many self-tests do not make assumptions about the dimension of a tested system. This appears to be very convenient, since all states can be assumed to be pure and all measurement operators can be assumed to be projectors. Nevertheless not a single self-test up to now could be free from any kind of assumption about the underlying black-box. For example, communication between parties having one part of a Bell pair, or one qubit from a graph state is forbidden. Also, black-boxes are usually assumed not to have any memory, the constraint that doesn't have to be realistic in all the cases.

Another result worth emphasizing is the self-test of an EPR pair presented in the paper of Bancal et al. [BNS$^+$14]. In this procedure, known as the SWAP-test, one collects the measurement outputs reported by black-boxes, calculates the CHSH inequality violation V and then uses NPA hierarchy [NPA08] to estimate the minimum fidelity between the physical and the reference state. The fidelity is optimized over all quantum states that give the CHSH inequality violation V. The robust bound of results obtained via the SWAP-test is much better than the corresponding bound obtained in the Mayers-Yao self-test. The SWAP-test has recently been used to self-test states other than EPR pair [PVN14, WCY$^+$14].

For the end of this subsection we will see how some particular reference experiments look like. The corresponding self-tests will be important for the development of interactive proofs.

- *RE1: Mayers-Yao self-test of a singlet* [MY04]: The reference experiment consists of an EPR pair: $|\Phi_+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ and the measurement observables X, Z and $\frac{X+Z}{\sqrt{2}}$

- *RE2: McKague self-test of a connected graph* [McK11]: Let $G = (E, V)$ be a connected graph with at least two vertices. Let $|\psi\rangle$ be the corresponding graph state with stabilizers $S_v, \quad v \in V$. For the chosen edge $(u, v) \in E$, the reference experiment consists of the state $|\psi\rangle$, the stabilizer measurements S_v and the measurements $Z_u Z^{N_u}$, $D_u Z^{N_u}$ and $D_u X_v Z^{N_v/\{u\}}$, where $D_u = \frac{X_u + Z_u}{\sqrt{2}}$.
- *RE3: McKague self-test of a connected graph with an odd cycle* [McK11]: Let $G = (V, E)$ be a connected graph containing an odd induced cycle $C = (V', E')$. Let $|\psi\rangle$ be the corresponding graph state with stabilizers S_v, $v \in V$. The reference experiment consists of the state $|\psi\rangle$, the stabilizer measurements S_v and the measurement $X^{V'} Z^{N(V')}$.

1.7 Interactive proofs

An interactive proof is a model of computation consisting of the exchange of messages between the two parties. The parties are denoted as the *verifier* and *prover*. The prover is a very strong computer, in our case a quantum one, but cannot be trusted. The verifier is a machine with limited computational abilities and its task is to make the prover solve some problem and become sure that the problem is solved, i.e. that the prover is not cheating. This description of an interactive proof fits well in our search for a process confirming that some computationally strong machine is actually a quantum computer. All one needs is a suitable interactive proof, consisting of a classical verifier and a supposedly quantum prover. The verifier needs to

- convince itself that the prover has quantum resources and that it is performing quantum operations and,
- use the prover to perform some calculations

We will present here two interactive proofs, one by Reichardt, Unger and Vazirani [RUV13] and the other by McKague [McK13]. In the paper describing the former it is argued that "one cannot distinguish between a quantum system that evolves as desired and a device that merely simulates the desired evolution using a classical computer" [RUV13]. Few reasons for this are mentioned: impossibility to gather statistics from repeated experiments because adversarial system with memory can deliberately deceive the verifier; the measurements collapse the state and moreover the dimension of a Hilbert space grows exponentially with the number of particles in the system while information accessible via measurement grows only linearly. The possible solution is straightforward: take more than one prover.

The paper of Reichardt, Unger and Vazirani (RUV in further text) [RUV13] presents the interactive proof in which a classical verifier interacts with two quantum provers. It is claimed that provers possess the tensor product of EPR pairs and can perform quantum operations. In the usual manner quantum provers have the names Alice and Bob, while the classical verifier is named Eve. In the self-testing procedure Eve plays sequential CHSH games with Alice and Bob (see Box 1.1). If really holding an EPR pair Alice and Bob will win a CHSH game with the probability ω, which corresponds to the Tsirelson's bound. Moreover this result is robust: if Alice and Bob win the game with the probability $\omega - \epsilon$ their strategy for playing the CHSH game is $O(\sqrt{\epsilon})$ close to the ideal strategy. A strategy for playing CHSH game includes the state in possession and the measurement operators to be performed. Reichardt et al. prove that if Alice and Bob when playing n sequential CHSH games win at least $n\omega(1 - \epsilon)$ times they have a state that is close to the tensor product of n EPR pairs (possibly tensored with an arbitrary ancilla) and perform measurements which are by norm close to the measurements applied in the ideal strategy for playing the CHSH game. This result is far from being obvious since in this scenario Alice and Bob have memory and their strategy generally can depend on the previous history. However, the only strategy that allows the provers to win $n\omega$ sequential CHSH

games is to use ideal strategy for playing n independent single CHSH games. The quantum dynamics is verified by performing the teleportation based quantum computation [GC99]. To conduct this type of computation both Alice and Bob need to perform specific measurements on their qubits. Eve prevents Alice and Bob to cheat when asked to perform computation by introducing two new tasks: the process tomography and the state tomography. In the state tomography Bob is asked to perform the same measurements he would do in the computational process, while Alice plays the usual CHSH game. Any inconsistency in reported outputs would imply that they are not to be trusted. In the process tomography Alice is asked to perform the same measurements she would do in the computational process while Bob just plays the CHSH game. In this way Alice cannot distinguish the CHSH game from the state tomography nor the process tomography from the computation. Bob, on the other hand cannot differentiate between the CHSH game and the process tomography, nor between the state tomography and the computation. For the full details see [RUV13] and [RUV12].

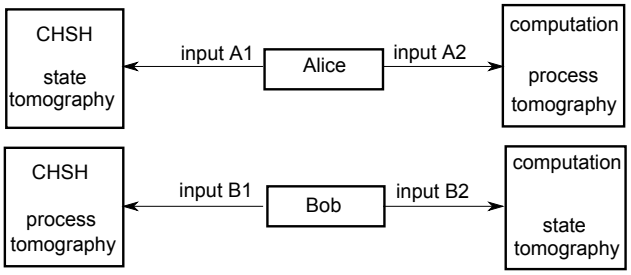


Figure 1.10: The scheme of the Reichardt, Unger, Vazirani interactive proof. By the input she gets Alice cannot distinguish the CHSH game from the state tomography, nor the computation from the process tomography. Bob's input does not allow him to distinguish the CHSH game from the process tomography nor the computation from the state tomography.

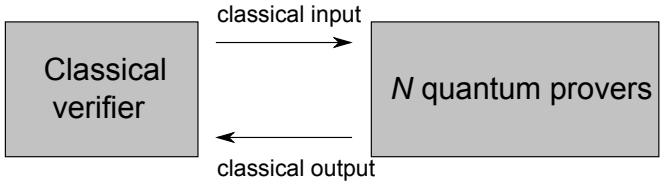


Figure 1.11: The skatch of a general interactive proof. One classical verifier is interacting with N quantum provers

Instead of using two provers, each one performing a big number of operations McKague's interactive proof uses a big number of provers each performing only one operation. As mentioned above an interactive proof has two parts, one being the honesty test (certifying that provers are quantum computers) and the other being some specific calculation. McKague uses the already established theory of self-testing for the honesty test, while the underlying computation model is 1WQC. This means that quantum provers share a graph state, each possessing one vertex. It is assumed that the provers do not communicate. The existence of the self-testing procedure for the resource state of 1WQC makes this interactive proof very elegant. However, it is not problem-free. The provers shouldn't be able to distinguish the situation in which they are being tested and the one in which they are asked to perform a computation. If they can

discriminate these two processes they can be honest when performing an honesty-test, but cheat when they are computing. As described in the previous section during the procedure of the graph state self-testing the following observables are measured on the constituent qubits: X, Z and $D = \frac{X+Z}{\sqrt{2}}$. All these measurements belong to the XZ-plane of the Bloch sphere. On the other hand universal quantum computation in MBQC involves the measurements of Pauli Y (when performing the CNOT gate for example) as well as the measurements in the XY-plane (the simulation of a general rotation). Thus, any time some of the provers get to measure Y they would know that it is not being self-tested but asked to perform some computation. Luckily, the solution for this problem is provided. Mhalla and Perdrix [MP12] proved the following theorem:

- **Theorem 1.1.** Triangular cluster states[5] are universal resources for MBQC based on measurements X, Z, $\frac{X\pm Z}{\sqrt{2}}$ and the number of vertices is polynomial in the size of the original circuit we wish to perform.

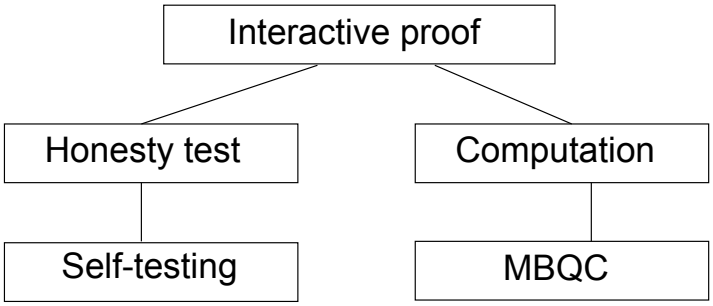


Figure 1.12: The scheme of McKague's Interactive proof

A triangular cluster state is a graph state whose underlying graph is the triangular lattice (Fig. 1.13). This completes McKague's interactive proof. A verifier can just toss a coin to decide whether to perform the honesty-test or the computation with provers being unable to distinguish the two processes. Moreover the computing is blind, since every prover performs just one measurement which tells them nothing about the computation process. More details about the MBQC on a triangular lattice will be provided in one of the following chapters (2.3.1).

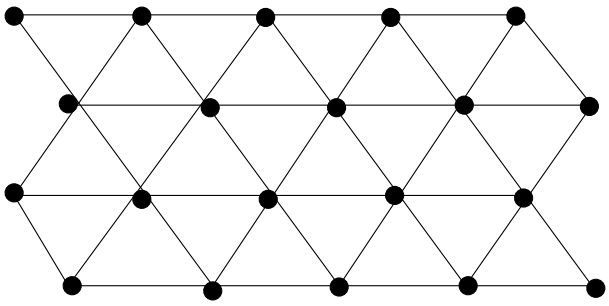


Figure 1.13: A triangular cluster state is sufficient for universal quantum computation in MBQC

[5]There is a slight abuse of the term "cluster state". It has been defined that cluster states have a square lattice as underlying graph, but here cluster state is any state having a regular lattice as the underlying graph, in this case a triangular lattice

Chapter 2

Alternative interactive proofs

The interactive proof introduced by McKague has some very convenient properties. In the first place it uses a physically tractable computation model. When dealing with graph states the test for honesty is straightforward and very elegant. As mentioned in the previous chapter the fact that the computation is blind gives more power to this interactive proof. Despite all the good points a big number of required provers makes this interactive proof very difficult for practical implementation. It doesn't seem realistic that one can get, for example, one hundred quantum provers in order to perform the interactive proof. In the last chapter of his paper presenting this interactive proof [McK13] McKague discusses the possibilities for simplification. One idea is to try to share a graph state between two parties and generalize the interactive proof for this scenario. The bottleneck of this construction is a self-test for a graph state when the underlying graph contains an odd cycle. In such a case one prover would possess an entangled pair of qubits and as the author argues "likely it is not possible to prove the self-testing theorem" [McK13] for this case. In this chapter we will present the results obtained while trying to find a way to construct some interactive proof which would be easier to implement than the one with poly(n) number of provers (n being the size of implemented quantum circuit). To get to the simple two-provers case one would have to use bipartite graphs (yet to be proved that it can work), but triangular lattice doesn't belong to this class of graphs. There are two possible solutions for this problem:

(a) looking for a way to self-test an edge (entanglement) of a graph state held by a single prover or

(b) looking for a different computation model implementable on a bipartite graph state

In the following sections some attempts to solve the aforementioned problem using the strategy (a) will be presented. Beforehand one try and difficulties connected with the strategy (b) will be reported. Shepherd and Bremner proposed a restricted class of quantum computation, *instantaneous quantum computation* (IQC in the future text) which is temporally unstructured, meaning that all operations that guide the process can be done at once [SB09]. Since the temporal structure is the only constraint there is more than one implementation of this process. The circuit that implements IQC is called IQP circuit (instantaneous quantum polytime) and it consists of the gate which is diagonal with respect to the eigenbasis of Pauli-X operator [HWA$^+$14]. There is one gate because the computation is instantaneous, but the gate can be decomposed into a product of commuting gates. There is also a MBQC implementation of IQC [SB09, HWA$^+$14] and it uses a bipartite graph as a resource. Interestingly there is strong evidence that IQC cannot be efficiently simulated by a classical computer [BJS11]. The fact that classical device cannot perform IQC implies that it can be used for the construction of an interactive proof. Taking into account this, together with convenient MBQC implementation, makes IQC a perfect model for our purpose. MBQC implementation of IQC (graph program in the further text) is given on Figure 2.1. In a graph program every qubit from an IQP circuit

is simulated using a primal qubit which is measured in the X-basis. A computation is led by mutually non-adjacent ancilla qubits that are measured in the basis $U_x(-\theta_i)ZU_x(\theta_i)$. These measurement bases are from the YZ plane of the Bloch sphere and that is actually the bottleneck for the construction of the interactive proof which would include the usual graph state self-test. As pointed out in section 1.6 one cannot self-test measurements from the YZ plane because of impossibility to distinguish a measurement operator from its complex conjugate, and clearly YZ-plane measurement bases can have imaginary relative phases. This obstacle incapacitates self-testing and dealing with it should be one of the priorities in the future research.

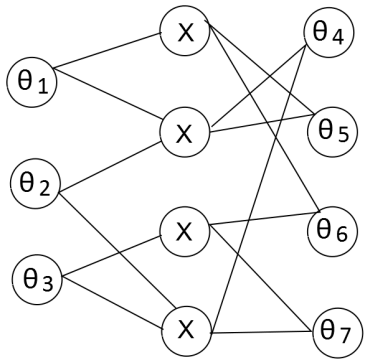


Figure 2.1: MBQC implementation of IQP circuit. Qubits are denoted by the measurement bases. X is the usual Pauli-X, while θ_i denotes the basis $U_x(-\theta_i)ZU_x(\theta_i)$, where $U_x(\theta_i)$ is rotation for θ_i around x axis [HWA$^+$14]

.

2.1 The nonlocality tests

Before even starting to think about the structure of the two-prover self-test for an arbitrary graph state one has to be convinced that the graph state is nonlocal across the introduced bipartition. The concept of nonlocality is nonexistent in classical theory and presence of it in some system is the clear sign of quantumness. On the other side, if a system does not exhibit nonlocality it can be described by a local hidden variable model, i.e. it is classically simulatable. Every self-test implies nonlocality and cannot be implemented on a local state. Graph states are clearly nonlocal, but question is if nonlocality is still exhibited if specific bipartition is introduced. In that case all qubits held by the same party are allowed to communicate and possibilities for a classical simulation increase.

The triangular state (Fig. 2.4) and the patch of two trangular states (Fig. 2.5) were the first states for which we have tested nonlocality across the specific bipartiton. However, since behind all the nonlocality tests we have used lies the same idea here we will present the nonlocality test of the triangular cluster state consisting of 12 qubits. The choice of this state is dictated by the fact that it is convenient for the implementation of specific single-qubit unitaries. Any single qubit unitary can be performed on the state of the same structure, by just increasing the length (the number of qubits).

The exact task is to test nonlocality in the scenario where two parties, A and B, share the 12-qubit triangular cluster state. The input for each party represents the measurement choice and the output is the measurement outcome. A local deterministic strategy of each party prede-

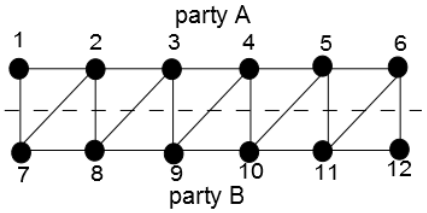


Figure 2.2: The triangular cluster state shared between two parties

fines the output for each given input. A complete local model is formed by taking into account local detreministic strategies of both parties. For example, let us assume that each party has two qubits in its possession. In one deterministic strategy party A decides that for the input XX (both qubits measured in the X-basis) it reports the output 0, while party B in one of its deterministic strategies decides that for the input ZZ the reported output is 1. In the local model corresponding to these two deterministic strategies the input $XXZZ$ will yield the output $1 \oplus 0 = 1$. The correlations obtained by measurements performed on some state are local if they can be written as a convex sum of the correlations obtained by the same measurements in all possible local deterministic strategies the parties can use. Consequently, in the opposite case the measurements are nonlocal.

For the nonlocality test construction linear programming was used (for the details about the technique check Box 2.1). If there are l possible inputs let $\mathbf{P}$ be an l-dimensional vector of measurement probabilities.[1] On the other hand let $\mathbf{P_i}$ be the vector of measurement probabilities for the same set of measurement choices in the i-th local model. Note that the elements of $\mathbf{P_i}$ can be 0 or 1. The convex sum of local models can be written as $\sum_i \mu_i \mathbf{P_i}$ where $\forall i \quad \mu_i \geq 0$ and $\sum_i \mu_i = 1$. This convex sum defines a local polytope. The scaling parameter λ is defined so that the following constraint should be satisfied:

$$\lambda \mathbf{P} + (1 - \lambda)\mathbb{1} \leq \sum_i \mu_i \mathbf{P_i} \tag{2.1.1}$$

$\frac{1}{2}\mathbb{1}$ is the l-dimensional vector representing the totally random outcome. If the minimal λ for which the constraint above is satisfied is smaller than one then the correlations obtained by the corresponding measurements on the state are nonlocal, otherwise they are local. For finding the minimal λ the following linear program was created:

$$\begin{aligned} & & \text{minimize} \quad & \lambda \\ \text{s.t.} \quad & \lambda \mathbf{P} + (1 - \lambda)\mathbb{1} \leq \sum_i \mu_i \mathbf{P_i} \\ & & \sum_i \mu_i = 1 \\ & & \forall i \mu_i \geq 0 \end{aligned} \tag{2.1.2}$$

For now we have said nothing about the measurement choice. The measurements on both sides are chosen on a such way that the expected values of the joint measurements are known and for our case these are exactly the measurements used in the McKague self-test of a specific edge: stabilizers + three measurements defined on the edge (RE2 in section 1.6). In the situation where the number of qubits is 12, together with 3 non-stabilizing measurements the final number of performed measurements per party is 15 which gives $2^{15} * 2^{15} = 2^{30}$ local models. A linear

[1] $\mathbf{P_x} = (P(m_1, M_1), ..., P(m_1, M_1))$, where $P(m_i, M_i)$ is the probability for getting the output m_i if the input is M_i

program cannot perform an optimization for such a big number of local models, so certain simplifications have to be made. In order to successfully test the nonlocality the size of a graph state has to be smaller. This can be achieved by performing Z-measurements on some qubits. As pointed out in section 1.3, the Z-measurement removes measured qubits and disentangles the graph state making it a tensor product state of two graph states (Fig. 2.3a). If one term of the tensor product is nonlocal the whole state is nonlocal. Using this procedure the idea is to divide a 12 qubit graph into subgraphs consisting of 4 qubits (produced by Z-measurements) and test nonlocality of each subgraph. There are 5 such subgraphs: 1-2-7-8, 2-3-8-9, 3-4-9-10, 4-5-10-11, 5-6-11-12 (Fig. 2.3). Before explaining the procedure in more detail the joint measurements on the 4-qubit graph state (qubits denoted with numbers from 1 to 4) with known expectation values will be presented. Four stabilizers are included:

$$S_u = X_u Z^{N(u)} \tag{2.1.3}$$

where u is specific qubit(vertex) and $N(u)$ is the set of neighbors of u. Furthermore, for the specific edge u-v three measurements are added to the stabilizers:

$$\begin{aligned} &Z_u Z^{N(u)} \\ &D_u Z^{N(u)} \\ &D_u X_v Z^{N(v)/u} \end{aligned} \tag{2.1.4}$$

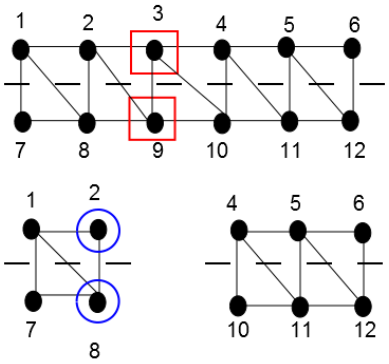
where $D_u = (X_u + Z_u)/\sqrt{2}$. The expectation values when the joined measurements are stabilizers is 1, while for the three remaining measurements expectation values are 0, $\frac{1}{\sqrt{2}}$ and $\frac{1}{\sqrt{2}}$ respectively.

Now the procedure for the nonlocality testing of subgraphs will be presented. As already shown, the graph can be divided on five 4-qubit subgraphs. For all of these subgraphs the measurements are chosen according to (2.1.3), (2.1.4) while the separate test can be performed for each edge across the bipartition. For example, to test nonlocality of the first subgraph (1-2-7-8) Z-measurements have to be applied on qubits 3 and 9 (Fig. 2.3).

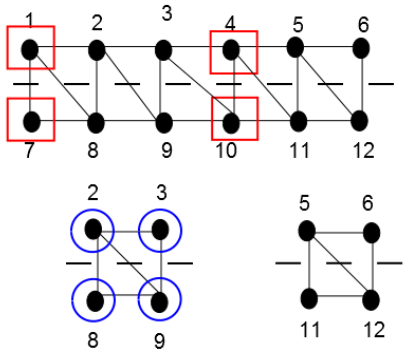
It is important to take into account that the result of Z-measurements on qubits 3 and 9 (we will denote these results as z_3 and z_9, respectively) affect the remaining graph state. On all the neighbors of qubit 3(9) operator $Z^{z_3}(Z^{z_9})$ will be applied. On Figure 2.3a this action is displayed with blue circles encircling the "affected" qubits. These transformations are taken into account as unitary transformations of operators constituting joint measurements. If, for example, the result of the Z-measurement performed on qubit 9 is $z_9 = 1$, one stabilizer will be changed, because qubits 8 and 9 are neighbors:

$$Z_1 Z_2 Z_7 (Z_8 X_8 Z_8^\dagger) = -Z_1 Z_2 Z_7 X_8 \tag{2.1.5}$$

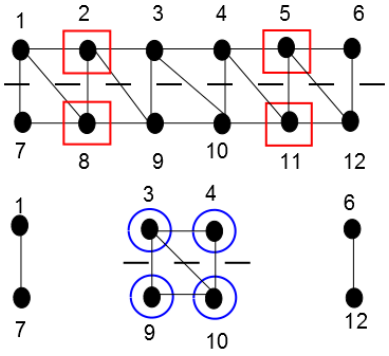
However this unitary transformation of the stabilizer will not affect its stabilizer role (meaning that the remaining graph state will still be its eigenvector corresponding to the eigenvalue 1). This means that the expected values of the measurements will not be changed, and thus nonlocality testing procedure will be the same for all results of Z-measurements performed on qubits 3 and 9. In this scenario there are 5 measurement choices for one party (XZ, ZX, ZZ, ZI, DZ) and 4 measurement choices for the other one (XZ, ZX, ZZ, ZI) up to the global phase $(-1)^{z_3(9)}$. The number of local models is $2^5 * 2^5 = 1024$. Linear program optimization showed that the minimum value of the scaling parameter for subgraph 1-2-7-8 with respect to the bipartition and for the previously defined measurements is 0.8284, meaning that the state with respect to the introduced measurements is nonlocal.



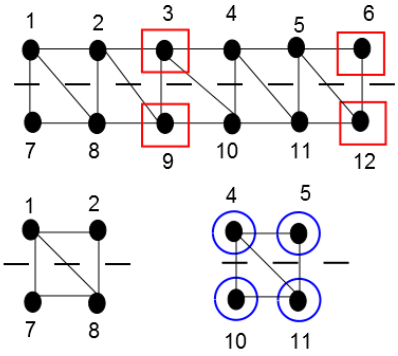
(a) The nonlocality test for the subgraph 1-2-7-8



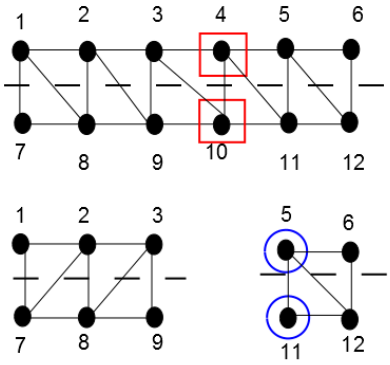
(b) The nonlocality test for the subgraph 2-3-8-9



(c) The nonlocality test for the subgraph 3-4-9-10



(d) The nonlocality test for the subgraph 4-5-10-11



(e) The nonlocality test for the subgraph 5-6-11-12

Figure 2.3: Nonlocality tests for all the subgraphs have the same structure. Z-measured qubits are denoted with red squares around them, while their neighbors are denoted with surrounding blue circles.

The analysis of the other subgraphs adds one difference, but we will show that this difference doesn't change anything in the already explained and performed test of nonlocality for subgraph 1-2-7-8 (Fig. 2.3). The only difference between subgraphs is an eventual Z unitary, originating from Z measurements on the neighboring qubit(s), but as already established this does not

introduce any difference in the stabilizer roles and expectation values, so all subgraphs will exhibit nonlocality. However, even though all expected values are the same, measurements might be changed. For example we will take a look at the additional measurement $D_u Z^{N(u)}$. In the first subgraph (Fig. 2.3a) this operator is not altered because qubit 1 plays the role of qubit u, which does not have Z-measured neighbors. For all the other subgraphs this is not the case, and the alteration of D_u cannot be avoided since for some blocks all qubits have Z-measured neighbors. As proved in [McK11] this measurement is required, since all measurements using X and Z alone can be simulated using a classical hidden variable model. D_u is introduced in the measurement which is the linear combination of one stabilizer and the operator obtained from the same stabilizer by replacing X_u with Z_u. This gives the form $D_u = \frac{X_u+Z_u}{\sqrt{2}}$, but since in our case the stabilizers have added phase factor -1, the same procedure for defining D_u will give the new form $D'_u = (-X_u + Z_u)/\sqrt{2}$. The only motive for introducing this operator in the reference experiment was to have a measurement that will not be tensor product of Xs and Zs. An equally good choice would be any normalized linear combination $\alpha X + \beta Z$, which includes $D'_u = \frac{-X_u+Z_u}{\sqrt{2}}$. Since we have proved that nonlocality tests for all subgraphs are essentially the same as the nonlocality test for subgraph 1-2-7-8 we can conclude that the whole 12-qubit graph state will also be nonlocal.

2.2 Single prover holding an entangled pair of qubits

With the current resources a self-test of a general non-bipartite graph state shared between two provers seems like a very difficult problem. If one thinks not about a general non-bipartite graph state but about specific triangular lattice graph state the situation doesn't get any better. Our first try is to isolate one cell of a triangular lattice graph state, namely one triangle and check the possibilities to self-test the graph state having this triangle as the underlying graph. Two papers are published recently [PVN14,WCY$^+$14] with the detailed descriptions of the self-test of three-qubit states: GHZ state and W-state (which is not a graph state) [CKW00]. Both groups of authors considered GHZ-state and W-state shared between three parties. This approach corresponds to the McKague's in the self-test of a multipartite graph states where each qubit is held by a separate party. At this point no one presented a successful self-test of a three-qubit non-local state shared between two parties. If such a self-test can be constructed one has to be sure that the state with two entangled qubits held by a single party can be discriminated from the state in which these two qubits are not entangled. Checking the possibility for this discrimination sets the scene for this chapter.

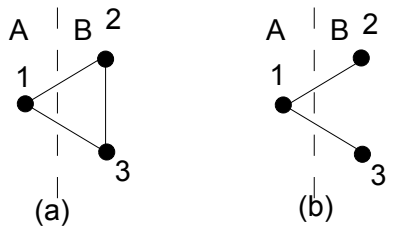


Figure 2.4: On the (a) is given the triangular state with the edge between the qubits 2 and 3, while on the (b) this edge is not present. Is it possible to distinguish these two states when they are put into a black-boxes? (Black-box containing qubit 1 is denoted as A, while the other one is denoted as B

The two states we would like to discriminate in a self-testing scenario are given on Figure 2.4. For the state on Figure 2.4a we will use the term *triangular state* and for the one on Figureb 2.4b the term *linear state* will be used. Having this notation in mind the question on

which we would like to find the answer is: Can we distinguish the triangular and the linear state if they are shared by two black-boxes and the only thing we know could be some measurement results? These two states are locally equivalent, meaning that there is a local isometry which when acts on the linear state transforms it into the triangular state or vice versa. Moreover, one state is the local complement of the other.

Definiton 2.1 [HDE+05] The *local complement* of a graph G at a vertex v is a graph $\tau_v(G)$ in which the neighborhood of v is reversed, meaning that an edge is added between all mutually non-adjacent neighbors of v, while an edge connecting two neighbors of v is erased. The local complement is obtained by performing the following operation:

$$|\tau_v(G)\rangle = U_v(G)\,|G\rangle\,, \quad U_v(G) = e^{-i\frac{\pi}{4}(\sigma_x)^v} e^{i\frac{\pi}{4}(\sigma_z)^{N_v}} \tag{2.2.1}$$

It is easy to see that triangular state is the local complement of linear state at vertex 1. The explicit form of the two states is given in equation (2.2.2). Triangular state is symmetric with respect to the arbirary permutations of qubits, while linear state is symmetric with respect to the permutation of qubits $(2,3) \longmapsto (3,2)$.

$$\begin{aligned} |\psi_{\text{triangle}}\rangle &= \frac{1}{2}(|00+\rangle + |01-\rangle + |10-\rangle - |11+\rangle) \\ |\psi_{\text{linear}}\rangle &= \frac{1}{\sqrt{2}}(|0++\rangle + |1--\rangle) \end{aligned} \tag{2.2.2}$$

The fact that one prover holding a non-entangled pair of qubits can always simulate the correlations obtained by an arbitrary measurement on a corresponding pair of entangled qubits is easy to prove and well-known [BCP+14]. Our situation is a little bit different since entangled qubits 2 and 3 are furthermore entangled with qubit 1 that is held by a separate prover.

In order to check the possibilities of black-box B, having in possession linear state, to simulate statistics for triangular state we have to define measurements that are supposed to be tested and to make some assumptions. The measurements we are primarily interested in are those that are supposed to appear in the self-test. The anticipation would lead us to the measurements used in the Mayers-Yao self-test or those used in the graph state self-test. The measurements used in both self-tests are the Pauli's X and Z and their linear combination D (introduced in the previous chapter). The following assumptions will be made:

(a) black-box B contains qubits 2 and 3 of linear state;

(b) black-box A can be opened and manipulated, its state and operations are trusted;

(c) black-box B does not have memory;

(d) there is no communication from A to B.

At first sight some of these assumptions might seem quite unreasonable. Assumption (a) is just the re-statement of the thesis we want to check. The content of assumption (b) is not usual when dealing with self-testing. Nevertheless we can make this assumption because this problem emphasizes the simulation possibilities of black box B. If black-box A is to be trusted problem becomes simpler and easier to define. In the later stages we can make it more realistic. This assumption puts the problem in a well-known steering scenario in contrast to the fully device-independent [WJD07]. Assumptions (c) and (d) are usual for these kinds of problems and while (d) seems reasonable (c) is problematic, but ruling it out basically makes the self-testing problem infeasible.

As already mentioned we have used semi-definite programming to solve this problem. More details about this very useful tool can be found in Box 2.1. For this particular problem sedumi solver was used. Code *sdpST3* written in order to perform the optimization can be found in Appendix C. Two density matrices are defined, ρ_{triangle} and ρ_{linear}, as well as six projectors onto eigen-spaces of two Pauli operators (X and Z) and $D = \frac{X+Z}{\sqrt{2}}$. The set of defined projectors is $P = \left\{P_X^{-1}, P_X^1, P_Z^{-1}, P_Z^1, P_D^{-1}, P_D^1, \right\}$. For example P_X^{-1} projects onto the eigenspace of Pauli X corresponding to the eigenvalue -1 and so on. Upon performing the measurements corresponding to the defined projectors on state ρ_{triangle} specific outcomes are obtained with well-defined probabilities. There are 36 combinations of the measurements that can be performed on qubits 2 and 3. Thus, 36 POVMs are also defined in the program, dimension of each one being 4. These POVMs (see Box 2.2) are defined in such a way that when applied on the state ρ_{linear} each of them reproduces the measurement statistics of one pair of projectors from the set P on state ρ_{triangle}. For each command of measurements to perform (for example P_X^{-1} on the qubit 2 and P_Z^{-1} on the qubit 3) the black-box applies one specific POVM. Another constraint is that the sum of four POVMs corresponding to the measurement of the product of two specific observables should be equal to the four-dimensional identity operator. For example:

$$P_X^{-1}P_Z^{-1} + P_X^{-1}P_Z^1 + P_X^1P_Z^{-1} + P_X^1P_Z^1 = \mathbb{1}. \tag{2.2.3}$$

One POVM corresponds to the one product of two projectors independently of the measurement choice of party A, since according to assumption (e) we are working in the non-signaling scenario.

The optimization is performed over all positive semi-definite matrices corresponding to the defined 36 POVMs. It showed that such a set of POVMs exists, so black-box B can successfully simulate triangular state while holding linear state. This result is not encouraging for the construction of the two-prover self-test of the non-bipartite graph state. In order to improve the self-test and get deeper knowledge of the simulation process we can exploit more assumption (c). If black-box (party) A is trusted we can manipulate its state and try to force black-box B to make an error while trying to simulate the measurement statistics. The triangular-linear state scenario does not offer too many possibilities for changing the environment of black-box B. All that party A can do is the change of the measurement projector, but as we have seen it doesn't affect simulation capabilities of black-box B. Obtained result does not give too much insight in the nature of this simulation. Maybe there is something special about the triangular state - linear state relation that admits the simulation of the measurement results. One suggestive thing is that the order of measuring of qubits 2 and 3 is irrelevant, since, as it is pointed out above, linear state is symmetric with respect to the interchange of qubits 2 and 3. Using the four-qubit state instead of the three-qubit state can give answer on the raised question 2.5. On the one hand the symmetry between qubits 2 and 3 is broken so the untrusted party has less freedom in the simulation process, while on the other hand the trusted party has more freedom when it comes to changing the whole state since it possesses two qubits.

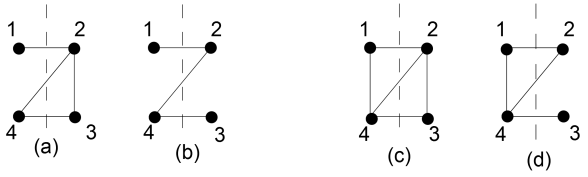


Figure 2.5: The figures (b) and (d) represent four-qubit states that are supposed to simulates the measurement statsistics obtained on the states (a) and (c) respectively

Box 2.1 Linear and semidefinite programming

Semidefinite programming is one type of a convex optimization [VB94, Fre04]. It is more general than the linear programming and every linear program can be written as a semidefinite one.

A linear programming problem can be defined as a problem of minimazing/maximizing linear function subject to some linear constraints [BV04]:

$$\begin{aligned} & \text{minimize} && c \cdot x \\ \text{s.t.} \quad & a_i \cdot x = b_i, \quad i = 1, ..., m \\ & && x \in \mathbb{R}^n_+ \end{aligned} \tag{2.2.4}$$

where x is a vector of n-variables ($x = (x_1, ..., x_n)$), $c \cdot x$ is the inner product $\sum_{j=1}^{n} c_j x_j$ and $\mathbb{R}^n_+$ is the nonnegative orthant: $x \in \mathbb{R}^n_+$ iff $\forall i \quad x_i \geq 0$. The nonnegative orthant represents a convex cone. The set C is a convex cone if it is closed under linear combinations with positive coefficients. If one is searching for a maximum, rather than a minimum, the above given form can be adapted by multiplying with -1. The function to be minimized/maximized is called the objective function. Every x satisfying all the constraints from (2.2.4) is said to be feasible, otherwise it is said to be infeasible. If an objective function for a minimum (maximum) problem can take arbitrarily large negative (positive) values the problem is said to be unbounded, otherwise it is said to be bounded.

A semidefinite programming problem has basically the same structure as a linear programming problem, but the variable is an $n \times n$ matrix, rather than an n-dimensional vector. To translate (2.2.4) to the vector space of matrices one needs to define a linear function of a matrix and positivity of matrices. Let X be a symmetric $n \times n$ matrix. The general form of a linear function of the matrix X is $\mathrm{Tr}(CX)$ where C is an arbitrary symmetric matrix compatible to X . Positivity of the vector elements from the linear programming problem here translates as positive-semidefiniteness (thus name semidefinite programming). The matrix X is semidefinite (denoted as $X \succeq 0$) if $v^T X v \geq 0$ for every $v \in \mathbb{R}^n$. All eigen-values of a positive-semidefinite matrix are nonnegative. The set of positive-semidefinite matrices is a convex cone. The general form of a semidefinite programming problem is:

$$\begin{aligned} & \text{minimize} && \mathrm{Tr}(CX) \\ \text{s.t.} \quad & \mathrm{Tr}(A_i X) = b_i, \quad i = 1, ..., m \\ & && X \succeq 0 \end{aligned} \tag{2.2.5}$$

where A_i are symmetric $n \times n$ matrices. A semidefinite program, like the linear one, can also be infeasible, feasible bounded and feasible unbounded. Defining of the maximization problem is straightforward. The formulation (2.2.5) is the primal problem. One can also define a dual problem, but for our purpose we will not need to solve dual problems. For more details see [VB94].

The semidefinite programming has found wide use in quantum information theory. For the self-testing theory in particular is important its use in Navascues-Pironio-Acin hierarchy [NPA08]. This hierarchy can be used to characterize the set of quantum correlations and it is used in the SWAP-test, numerical self-tetsing procedure [BNS$^+$14].

The idea behind this slightly more complicated test is that a party besides changing the measurement settings can flip a coin to decide whether to entangle or disentangle the pair of qubits 1-4. In this way the party B does not know what is the whole state and thus cannot perform any state-specific simulation. Four different states on Figure 2.5 parts (a), (b), (c) and (d) are denoted as $|\psi_{2-3}\rangle, |\psi_{\text{lin4}}\rangle, |\psi_{1-4,2-3}\rangle, |\psi_{1-4}\rangle$, respectively, and their explicit forms in Dirac notation are:

$$|\psi_{2-3}\rangle = \frac{1}{2}(|+0+0\rangle + |-1-0\rangle + |+0-1\rangle - |-1+1\rangle) \tag{2.2.6}$$

$$|\psi_{\text{lin4}}\rangle = \frac{1}{2}(|+0+0\rangle + |-1+0\rangle + |+0-1\rangle - |-1-1\rangle) \tag{2.2.7}$$

$$|\psi_{1-4,2-3}\rangle = \frac{1}{2}(|+0+0\rangle + |-1-0\rangle + |-0-1\rangle - |+1+1\rangle) \tag{2.2.8}$$

$$|\psi_{1-4}\rangle = \frac{1}{2}(|+0+0\rangle + |-1+0\rangle + |-0-1\rangle - |+1-1\rangle) \tag{2.2.9}$$

For this situation an SDP optimization code has to include the definition of four density matrices ($\rho_{2-3}, \rho_{\text{lin4}}, \rho_{1-4,2-3}, \rho_{1-4}$), and like in the previous case six measurement projectors and 36 POVMs. Each POVM again simulates the action of the product of two projectors on qubits 2 and 3. The simulation is now more complex since one POVM on ρ_{lin4} simulates the measurement statistics of specific projectors on the state ρ_{2-3}, but at the same time if party A entangles qubits 1 and 2 the same POVM on the state ρ_{1-4} has to simulate the measurement statistics of the same projectors on the state $\rho_{1-4,2-3}$.

Like in the case of the triangular/linear state simulation the SDP optimization again finds the POVMs able to satisfy all the constraints. Furthermore, by inspecting the explicit forms of POVMs it can be seen that the POVM which simulates, say $P_X^{-1} \otimes P_Z^1$ in the case with three qubits does the same in the case with four qubits. In both cases the optimization gives the same set of 36 POVMs. This suggests that the simulation process is basically state-independent. The optimization for some more complicated state has not been done, but there is no reason to believe that the behavior would be different in the state of even higher dimension. There is one trivial thing the party B can do when asked to perform some measurement and it is to entangle qubits 2 and 3 before measuring them, on this way doing one of the operations:

$$U_{\text{ent}} |\psi_{\text{linear}}\rangle = |\psi_{\text{triangle}}\rangle \tag{2.2.10}$$

$$U_{\text{ent}} |\psi_{1-4}\rangle = |\psi_{1-4,2-3}\rangle \tag{2.2.11}$$

$$U_{\text{ent}} |\psi_{\text{lin4}}\rangle = |\psi_{2-3}\rangle \tag{2.2.12}$$

where U_{ent} is the entangling operation. This operation is state-independent and followed by required projective measurements it satisfies all the constraints from the optimization. However, this operation is not jeopardizing the eventual interactive proof. If the only way to simulate measurements statistics is to entangle originally non-entangled qubits 2 and 3, that means that prior to every measurement non-entangled qubits will become entangled and operationally we will have access to the claimed state. Thus, the optimization should be altered in order to rule out this trivial solution. The idea is to put a constraint that applied POVMs do not contain some entangling operation, or in other words that POVMs belong to the class of Local Operations and Classical Communication (LOCC) (see Box 2.3). Unfortunately, there is no straightforward way to impose LOCC constraint on our POVMs. The situation can be improved if we impose that POVMs have positive partial transpose (they belong to the PPT class). However in this way we are just getting closer to the desired constraint, since having PPT is necessary, but not sufficient condition for belonging to the LOCC class. Following this argument the codes were adapted, but the given POVMs still had the same structure.

Box 2.2 POVMs

A commonly found way for describing the measurement process in quantum mechanics is by using the Von Neumann's definition of projective measurements. In many quantum mechanics textbooks the projective measurements are given as one of the main postulates of quantum mechanics. For many purposes this is satisfactory, but not in quantum information theory. The most general kind of measurements are described by Positive Operator-Valued Measures (POVMs). To get a better understanding of the nature and the origin of the POVM description of the measurement process assume that the Hilbert space $\mathcal{H}$ of some system can be written as a direct sum of its two subspaces $\mathcal{H}_1$ and $\mathcal{H}_2$:

$$\mathcal{H} = \mathcal{H}_1 \oplus \mathcal{H}_2 \tag{2.2.13}$$

Now suppose that an observer has access only to the Hilbert space $\mathcal{H}_1$. An orthogonal projective measurement

$$P_m = |p_m\rangle \langle p_m| \tag{2.2.14}$$

performed on the whole Hilbert space does not have to be orthogonal measurement in the subspace $\mathcal{H}_1$. A projector onto the subspace $\mathcal{H}_1$ can be written as

$$P_1 = \sum_{m_1} |p_{m_1}\rangle \langle p_{m_1}| = \mathbb{1}_1 \tag{2.2.15}$$

where $|p_{m_1}\rangle$ are the basis vectors of Hilbert space $\mathcal{H}_1$. A projection that projects P_m onto $\mathcal{H}_1$ actually describes the measurement process observed "from" the Hilbert space $\mathcal{H}_1$. This operator is:

$$\begin{aligned} F_m = P_1 P_m P_1 = \sum_{m_1} |p_{m_1}\rangle \langle p_{m_1}| \, |p_m\rangle \langle p_m| \sum_{m_1'} \left|p_{m_1'}\right\rangle \left\langle p_{m_1'}\right| = \\ = \sum_{m_1} |p_{m_1}\rangle \langle p_{m_1}| \, p_m\rangle \sum_{m_1'} \left\langle p_m\right| \left. p_{m_1'}\right\rangle \left\langle p_{m_1'}\right| = \\ = |\pi_m\rangle \langle \pi_m| \end{aligned} \tag{2.2.16}$$

where $|\pi_m\rangle$ is the projection of $|p_m\rangle$ onto the subspace $\mathcal{H}_1$. The vector $|\pi_m\rangle$ is normalized only if $\mathcal{H}_1 = \mathcal{H}$. This means that operators F_m describing the measurement process in the subspace $\mathcal{H}_1$ are not projectors (because $F_m^2 \neq F_m$). However these operators sum to the identity operator of the subspace $\mathcal{H}_1$:

$$\sum_m F_m = P_1 \left(\sum_m P_m \right) P_1 = P_1 \mathbb{1} P_1 = P_1^2 = P_1 = \mathbb{1}_1 \tag{2.2.17}$$

The probability to get the result m can be written as

$$\text{Prob}\,(m) = \text{Tr}(\rho F_m) \tag{2.2.18}$$

where ρ is the density matrix of the system being measured. If every operator from the set $F = \{F_m\}$ is Hermitian, positive-definite and if the set is complete (meaning that the operators sum up to the identity) then F is a valid set of POVMs. In general case there is no simple way to describe the state of the system after performing a POVM measurement.

Box 2.3 LOCC

The local operations and classical communication (LOCC) belong to the class of non-entangling operations [HHHH09]. The name in a fairly accurate way describes what these operations do. For the case of bipartite state assume that Alice holds one part and Bob the other. Alice and Bob are in the spatially distant laboratories. When performing LOCC on the whole state Alice can perform measurement only on her qubit, and Bob on his. After obtaining the result of the measurement they can tell each other what was the outcome. LOCC cannot create any entanglement, nor increase amount of the entanglement in an already partially entangled state. If LOCC are performed on separable states (those that can be written in the form of a convex sum of product states) entangled states cannot be created.

The exact form of the optimized POVMs can be found in Appendix A. After obtaining this no-go result it is instructive to put a closer look at this operators since they have some interesting properties. The first interesting fact is that matrices representing all the POVMs in the computational basis have rank 3. For all of them solving the eigen-problem gives two non-zero eigenvalues: 1/4 and 1/2. The eigenvalue 1/2 is degenerate, since the eigenspace corresponding to it is two-dimensional. All of these POVMs share this property. The second important observation, but this time not satisfied for all POVMs, concerns the eigenvectors. Namely, all POVMs simulating the projectors corresponding to the following operators $X_2 \otimes X_3, X_2 \otimes D_3, Z_2 \otimes Z_3, Z_2 \otimes D_3, D_2 \otimes X_3, D_2 \otimes D_3$ for all eigenvectors have the maximally entangled states. This property is puzzling, but there is not a simple interpretation of the POVM's eigenvectors significance.

Before concluding this chapter another possibility will be presented. Namely, instead of commanding to black-box B to perform one measurement on the four-dimensional Hilbert space of qubits 2 and 3 one can command it to perform a certain measurement M_2 on the Hilbert space corresponding to qubit 2 and after getting the result of the measurement possibly command performing the measurement M_3 on the Hilbert space corresponding to qubit 3. There are some nice properties of this type of the test.

- Party B does not know if after performing the first measurement there will be the next one. There is possibility that the verifier is checking the expectation value of the operator $M_1 \otimes M_2 \otimes \mathbb{1} \otimes M_4$ where M_1 and M_4 are measurements performed by the party A. Moreover, in this scenario verifier should always check this expectation value, regardless of the possible measurement choice for the qubit 3.

- Suppose that in order to reproduce the outcomes of the measurement M_2 party B actually applies the POVM P_2. After that it is asked to perform the measurement M_3. In order to reproduce the outcome of the $M_2 \otimes M_3$ it has to apply some POVM P_3 on the state in which are the qubits 2 and 3 after the action of the POVM P_2. Different choices for M_3 impose different POVMs P_3. In this way the party B would not just have to reproduce the statistics, but also to perform sequential measurements. The question is if all constraints (including PPT) can be satisfied in this scenario. Unfortunately SDP optimization is not appropriate for the problem formulated in this way. To describe the state of the qubits 2 and 3 and sequential measurements one would need to define Kraus operators (see Box 2.4). This would make the optimization non-linear and thus infeasible.

Box 2.4 Kraus representation

In Box 2.2 POVMs are described as projectors "seen" from some subsystem of the system on which projectors act. In that way one can consider the unitary evolution of a tensor product state of a composite system of two qubits. When the state of one of the qubits is traced out the unitary evolution of the remaining qubit is known as a superoperator. Explaining the details of this topic goes beyond the scope of this thesis, so just a very short qualitative description with the final expression will be given. For the details see [Pre13, NC10].

If ρ is an initial and ρ' a final density matrix, a superoperator is defined as a mapping $\mathcal{S} : \rho \to \rho'$, that satisfies the following conditions:

- $\mathcal{S}$ preserves Hermicity of a density matrix it acts on;
- $\mathcal{S}$ is trace preserving;
- $\mathcal{S}$ is completely positive, meaning that every extension $\mathcal{S} \otimes \mathbb{I}$ is positive;
- $\mathcal{S}$ is linear.

Every superoperator satisfying these properties have Kraus (or operator sum) representation:

$$\mathcal{S}(\rho) = \sum_q M_q \rho M_q^\dagger \tag{2.2.19}$$

where $\{M_q\}$ is the set of Kraus operators.

If a superoperator has a Kraus representation it can be extended to a unitary evolution of the state of a bigger dimension, say on a state that is a tensor product of the pure state the superoperator acts on and some other pure state. The unitary evolution introduces entanglement between these two states, so evolution of the subsystem is not unitary anymore, and furthermore, not invertible. In this way a pure state evolves to be a mixed and this formalism is appropriate when describing the decoherence process. When the set of POVMs $\{F_p\}$ acting on a state ρ can be seen as a set of one-dimensional projectors acting on a state of a bigger dimension it is possible to define the state of the system after performing a POVM measurement:

$$\rho' = \frac{\sqrt{F_q}\rho\sqrt{F_q}}{\mathrm{Tr}(F_q\rho)} \tag{2.2.20}$$

It appears that in this case the POVM set can be constructed from the set of Kraus operators. Assume that ρ_1 (the state of system 1) is in a tensor product with some pure state $|0\rangle_2 \langle 0|_2$ (the state of system 2). A POVM measurement of ρ_1 can be described as the unitary evolution that entangles systems 1 and 2 followed by the orthogonal projective measurement performed on system 2. One can show that Kraus operators defined for this process can be viewed as a POVM set. However, the expression (2.2.20) introduces a non-linear relation, so as explained in the main text, the linear optimization process cannot be applied for the case involving Kraus operators.

In this section we have presented one way to check the possibilities for the simulation of entangled part of the state on non-entangled qubits, with the rest of the state being unchanged and furthermore unknown to the party that is simulating. The strategy to force this party to

make a mistake while trying to reproduce all the statistics is not an easy problem and the one used in the conventional self-test does not give a desired result. The one possible set of POVMs that can be used to reproduce the measurement statistics is given and some of its properties are described. However, it is not assumed that this is the only set of POVMs that satisfies all the constraints, it is just the one that is obtained through the optimization process. One can improve the optimization by finding a way to impose the condition that POVMs belong to the LOCC class. Other possible way of thinking is the above described sequential measurement scenario. If the numerical problem is infeasible there might be a way to prove analytically the possibility or impossibility to perform this sequential-measurements simulation.

2.3 Three-prover interactive proof for single qubit unitaries

As explained above the problem with the triangular cluster state is that it is not bipartite, since it consists of the net of odd cycles (Fig. 1.13). In the previous chapters we could see the difficulties with self-testing a graph state containing an odd cycle and shared between two parties. For now, we will assume that such a self-test cannot be constructed and look for a different solution. As noted above the problem with McKague's interactive proof is the big number of quantum provers involved. If the idea with reducing the number of provers to two does not seem to be implementable the natural question is:

What is the lowest number of provers which would still allow the implementation of the described interactive proof without substantially changing it?

The situation we want to avoid is the one in which one party holds a pair of entangled qubits (because of the difficulties with the self-test). Having this in mind the question can be rephrased:

What is the lowest number of parties which can share qubits of a certain graph state in a way that no party holds two or more entangled qubits?

This is easily translated to the graph theory language:

What is the lowest number of parties which can share a set of vertices of some specific graph so that no party holds two or more adjacent vertices?

This question finds the answer in the well-described theory of graph coloring [Wes00].

- **Definition 2.1** A graph is n-colorable if each vertex can be painted in one of n colours in such a way that two adjacent vertices are never painted in the same color.

Having the theory of graph colorability at our disposal we want to know in how many colors the triangular lattice graph can be painted? It does not take too much knowledge about graph theory to guess that triangular lattice graph is 3-colorable. Indeed, the 3-coloring of the triangular lattice graph is given on Figure 2.6.

The task now is to adapt the existing interactive proof to the situation with one classical verifier and three quantum provers. The interactive proof will again have the same structure, consisting of the honesty test and the computation process. In this work we will present the interactive proof for a quantum computer performing only single qubit unitary operations.

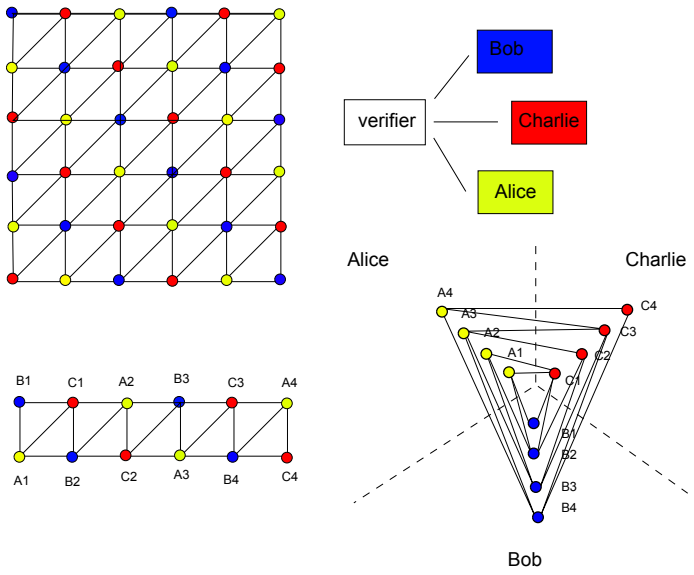


Figure 2.6: The three colored triangular lattice graph; Alice has yellow qubits, Bob blue and Charlie red ones. On the lower part of the figure is given the resource cluster state for performing a single qubit unitaries and the distribution of the state on three parties.

2.3.1 The computation process

As pointed out in section 1.7 Mhalla and Perdrix have proved that the triangular cluster state is universal resource for 1WQC based on the measurements from the XZC-plane of the Bloch sphere [MP12]. In this paragraph we will give the recapitulation of the implementation of some single qubit unitaries on the triangular cluster state, originally presented in [MP12]. For the universal set of quantum gates the authors use $S_{QU} = \{H, CZ, P(\alpha), \alpha \in [0, 2\pi]\}$, where $P(\alpha) = \cos(\alpha/2)X + \sin(\alpha/2)Z$, H is the Hadamard gate and CZ is controlled-Z. This set is proven to be universal for quantum computing in [Shi03]. If one is interested just in a single qubit unitary operations CZ is not needed. The resource state for the implementation of H and $P(\alpha)$ is 6×2 triangular grid. The implementation of the identity, Hadamard gate and $P(\alpha)$ is given on Figure 2.7. The qubits colored in black are measured in the X-basis, while the white ones are measured in the Z-basis. The one qubit marked with a checkerboard pattern is measured in the $(\cos(-\alpha)X + \sin(-\alpha)Z)$-basis. The square around a qubit means that the qubit is the input, while the encircled qubits represent outputs.

One thing that is left unclear is the choice of α. For the simplicity of the self-test we will choose $\alpha = \pi/4$, meaning that $P(\alpha) = \cos(\pi/8)X + \sin(\pi/8)Z$. In the further text the following notation will be used $F = \frac{X-Z}{\sqrt{2}}$. It is important here to point out that both the Hadamard gate and identity gate can be implemented on the triangular grid of a smaller size. In section 1.4 it has been explained that measuring two adjacent qubits in the X-basis with the qubits above and below this pair measured in the Z-basis represents the identity gate. Knowing this we can basically delete the part of the graph in which the only action is the implementation of the identity. This means that both the identity and Hadamard can be performed on the triangular cluster state consisting of just four qubits. $P(\alpha)$ also can be implemented on the reduced triangular cluster state consisting of eight qubits.

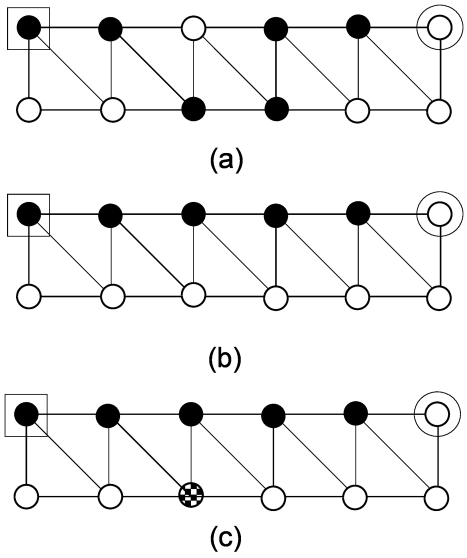


Figure 2.7: (a) shows the implementation of the Identity gate, on (b) the implementation of the Hadamard gate is given and on (c) the implementation of $P(\alpha)$. Black qubits are measured in the X-basis, the white ones in the Z-basis, while the checkerboard patterned qubit is measured in the $(\cos(-\alpha)X + \sin(-\alpha)Z)$-basis. The qubits having the square around them are the inputs, while encircled ones are the outputs.

2.3.2 The Honesty-test

The *honesty test*, like in the McKague's IP will be the self-test. All self-tests constructed up to now, except for the RUV one [RUV13], were treating one qubit per party, while the one that is going to be presented here deals with the situation when one prover/party holds more than one qubit. While the RUV self-test was considering two parties that supposedly hold tensor product of many independent states in our scenario parties hold a part of a highly entangled state. To make the proof shorter and less abstract we will present the self-test of the 12-qubit resource state, sufficient for the implementation of the simple one qubit unitary operations. Three parties sharing this physical resource state $|\Psi'\rangle$ are denoted as Alice, Bob and Charlie. Alice's qubits are denoted as A_1, A_2, A_3 and A_4, Bob's B_1, B_2, B_3 and B_4 and Charlie's C_1, C_2, C_3 and C_4 (Fig 2.6). The general structure of the self-test will remain unchanged. The aim is to construct the local isometry Φ which performs the transformation given in the following equation:

$$\Phi(|\Psi'\rangle) = |junk\rangle \otimes |\Psi\rangle \tag{2.3.1}$$

$$\Phi(A_i' |\Psi'\rangle) = |junk\rangle \otimes A_i |\Psi\rangle \tag{2.3.2}$$

where $|\Psi\rangle$ is the reference 12-qubit triangular cluster state (Fig 2.6), $|\Psi'\rangle$ is the physical state, while A_i (A_i') is the relevant reference (physical) operator. The construction of an isometry should rely on the anti-commuting relation of the certain observables corresponding to the Pauli X and Pauli Z. Furthermore, the anti-commuting relation should be obtained from the measurement results. The set of observables to be measured in the self-test should be the same as the set of the observables to be measured in the computation process. For the single qubit unitaries in 1WQC only four observables are being measured: X, Z, F and D. In order to be indistinguishable from the computation process the self-test has to include at least these four measurements. Since the triangular cluster state contains an odd cycle, in principle, it should represent the reference experiment described in section 1.6 under the name RE3. This reference experiment involves measurements of the observables X and Z. Adding the measurement of the

observables F and D to the self-test would make it more cumbersome but would not jeopardize it.

The isometry we want to construct should simulate the tensor product of the swap-gates acting on some part of the reference experiment and arbitrary number of ancillas. To construct such a swap gate one needs gates resembling Pauli's X and Z acting on a single qubit. The following table gives the list of the stabilizer measurements that the verifier asks the provers to perform. In this first sequence at least one prover is performing the measurement of X on a single qubit.

	Operator to be measured	Expectation value
S_{A_1}	$X_{A_1}Z_{B_1}Z_{B_2}Z_{C_1}$	1
S_{A_2}	$X_{A_2}Z_{B_2}Z_{B_3}Z_{C_1}Z_{C_2}$	1
S_{A_3}	$X_{A_3}Z_{B_3}Z_{B_4}Z_{C_2}Z_{C_3}$	1
S_{A_4}	$X_{A_4}Z_{B_4}Z_{C_3}Z_{C_4}$	1
S_{B_1}	$Z_{A_1}X_{B_1}Z_{C_1}$	1
S_{B_2}	$Z_{A_1}Z_{A_2}X_{B_2}Z_{C_1}Z_{C_2}$	1
S_{B_3}	$Z_{A_2}Z_{A_3}X_{B_3}Z_{C_2}Z_{C_3}$	1
S_{B_4}	$Z_{A_3}Z_{A_4}X_{B_4}Z_{C_3}Z_{C_4}$	1
S_{C_1}	$Z_{A_1}Z_{A_2}Z_{B_1}Z_{B_2}X_{C_1}$	1
S_{C_2}	$Z_{A_2}Z_{A_3}Z_{B_2}Z_{B_3}X_{C_2}$	1
S_{C_3}	$Z_{A_3}Z_{A_4}Z_{B_3}Z_{B_4}X_{C_3}$	1
S_{C_4}	$Z_{A_4}Z_{B_4}X_{C_4}$	1

Table 2.1: X Reference measurements

In the second sequence, compared to the ones defined in Table 2.1, the measurements are changed in a way that instead of measuring X_v on qubit v the prover is asked to measure Z_v. The expectation values of the operator products formed in this way are equal to 0 since X and Z anti-commute [McK11]:

$$\begin{aligned}\langle\Psi|\, Z_v Z^{N(v)}\,|\Psi\rangle = \langle\Psi|\, Z_v Z^{N(v)} X_v Z^{N(v)}\,|\Psi\rangle = \langle\Psi|\, Z_v X_v\,|\Psi\rangle = \\ = -\langle\Psi|\, X_v Z_v\,|\Psi\rangle = \\ = \langle\Psi|\, X_v Z_v\,|\Psi\rangle = 0\end{aligned} \tag{2.3.3}$$

where the fact that expectation values are real numbers is used, so $\langle\ \ \rangle^* = \langle\ \ \rangle$.

In the third sequence of measurements, the provers that were performing X-measurement on the single qubit in the first round will measure D on the same qubit. The other two provers do not change their measurement settings. Since $D_v = \frac{X_v+Z_v}{\sqrt{2}}$ it is trivial result that the expectation value of these measurements will be $\frac{1}{\sqrt{2}}$:

$$\langle\Psi|\, D_v Z^{N(v)}\,|\Psi\rangle = \frac{1}{\sqrt{2}} \tag{2.3.4}$$

Apart from the stabilizers from Table 2.1 there is a second list of stabilizers whose expectation values the provers should be asked to reproduce and they are given in Table 2.2. These stabilizers are formed in a way that at least one party is performing Z-measurement on a single qubit. Note that some measurements from Table 2.1 are reappearing here. The reason for introducing this second table will become apparent later.

	Operator to be measured	Expectation value
S_{B_1}	$Z_{A_1}X_{B_1}Z_{C_1}$	1
$S_{B_1}S_{B_2}$	$Z_{A_2}X_{B_1}X_{B_2}Z_{C_2}$	1
$S_{B_1}S_{B_2}S_{B_3}$	$Z_{A_3}X_{B_1}X_{B_2}X_{B_3}Z_{C_3}$	1
$S_{B_1}S_{B_2}S_{B_3}S_{B_4}$	$Z_{A_4}X_{B_1}X_{B_2}X_{B_3}X_{B_4}Z_{C_4}$	1
$S_{A_1}S_{A_2}S_{A_3}S_{A_4}$	$X_{A_1}X_{A_2}X_{A_3}X_{A_4}Z_{B_1}Z_{C_4}$	1
$S_{A_2}S_{A_3}S_{A_4}$	$X_{A_2}X_{A_3}X_{A_4}Z_{B_2}Z_{C_1}Z_{C_4}$	1
$S_{A_3}S_{A_4}$	$X_{A_3}X_{A_4}Z_{B_3}Z_{C_2}Z_{C_4}$	1
S_{C_4}	$Z_{A_4}Z_{B_4}X_{C_4}$	1

Table 2.2: Z Reference measurements

The measurements from Table 2.2 are subsequently changed in a way that the prover performing Z-measurement on a single qubit is asked to measure the same qubit in D-basis, while the other provers do not change their measurement settings. Since Table 2.2 contains stabilizers in which Alice and Charlie simultaneously perform Z-measurements on a single qubit it is important that they do not both measure D instead of Z, but only one of them at a time. The expectation values of these measurements are given in the following equation:

$$\begin{aligned} \langle\Psi| D_v M_A M_B |\Psi\rangle = \\ = \frac{1}{\sqrt{2}} \langle\Psi| (X+Z) M_A M_B |\Psi\rangle = \\ = \frac{1}{\sqrt{2}} \langle\Psi| X M_A M_B |\Psi\rangle + \frac{1}{\sqrt{2}} \langle\Psi| Z M_A M_B |\Psi\rangle = \frac{1}{\sqrt{2}} \end{aligned} \tag{2.3.5}$$

where M_A and M_B are the products of operators measured by Alice and Bob (without loss of generality it is assumed that the single qubit measurement is performed by Charlie). The first term in the third line is equal to zero due to the same argument used in equation (2.3.3).

The noticeable difference to the self test of one qubit per prover in this scenario is the whole new set of measurements to be performed, the one given in Table 2.3. Two last reference measurements in Table 2.3 are the so-called "triangle-measurements", formed by multiplying the stabilizers corresponding to the vertices of one triangle. It is easy to check that all triangle-measurements have the expectation value -1.

After presenting the list of the measurements that should be performed by three provers we will show that correct reproduction of all expectation values can lead to a certain commuting and anti-commuting relations. Note that up to now we were talking about reference measurements. Physical measurements will be denoted with prime ()'. For example, if the provers are asked to perform the measurement of the stabilizer S_{A_3} the measured observable actually will be $S'_{A_3} = X'_{A_3}(Z_{B_3}Z_{B_4})'(Z_{C_2}Z_{C_3})'$.

- **Lemma 1** If the physical measurements corresponding to the reference measurements defined in the tables 2.1, 2.2 and equations (2.3.3),(2.3.4) and (2.3.5) on the physical state $|\Psi'\rangle$ reproduce the expectation values of the respective reference measurements on the reference state $|\Psi\rangle$ the anti-commutation relation $\{X'_v, Z'_v\}|\Psi'\rangle = 0$ is satisfied for every qubit v of the state $|\Psi'\rangle$.

Proof. Let v be any qubit from the graph state $|\Psi'\rangle$. From the expectation values given in Table 2.1 we know that

	Operator to be measured	Expectation value
$S_{A_1}S_{B_3}$	$X_{A_1}Z_{A_2}Z_{A_3}Z_{B_1}Z_{B_2}X_{B_3}Z_{C_1}Z_{C_2}Z_{C_3}$	1
$S_{A_1}S_{C_4}$	$X_{A_1}Z_{A_4}Z_{B_1}Z_{B_2}Z_{B_4}Z_{C_1}X_{C_4}$	1
$S_{A_2}S_{B_1}$	$X_{A_2}Z_{A_1}X_{B_1}Z_{B_2}Z_{B_3}Z_{C_2}$	1
$S_{A_2}S_{B_4}$	$X_{A_2}Z_{A_3}Z_{A_4}Z_{B_2}Z_{B_3}X_{B_4}Z_{C_1}Z_{C_2}Z_{C_3}Z_{C_4}$	1
$S_{A_3}S_{B_2}$	$Z_{A_1}Z_{A_2}X_{A_3}X_{B_2}Z_{B_3}Z_{B_4}Z_{C_1}Z_{C_3}$	1
$S_{A_3}S_{C_4}$	$X_{A_3}Z_{A_4}Z_{B_3}Z_{C_2}Z_{C_3}X_{C_4}$	1
$S_{A_4}S_{B_1}$	$X_{A_4}Z_{A_1}X_{B_1}Z_{B_4}Z_{C_1}Z_{C_3}Z_{C_4}$	1
$S_{A_4}S_{C_2}$	$X_{A_4}Z_{A_2}Z_{A_3}Z_{B_2}Z_{B_3}Z_{B_4}X_{C_2}Z_{C_3}Z_{C_4}$	1
$S_{B_2}S_{C_3}$	$Z_{A_1}Z_{A_2}Z_{A_3}Z_{A_4}X_{B_2}Z_{B_3}Z_{B_4}Z_{C_1}Z_{C_2}X_{C_3}$	1
$S_{B_4}S_{C_1}$	$Z_{A_1}Z_{A_2}Z_{A_3}Z_{A_4}X_{B_4}Z_{B_1}Z_{B_2}X_{C_1}Z_{C_3}Z_{C_4}$	1
$S_{C_2}S_{A_1}$	$X_{A_1}Z_{A_2}Z_{A_3}Z_{B_1}Z_{B_3}X_{C_2}Z_{C_4}$	1
$S_{B_3}S_{C_4}$	$Z_{A_2}Z_{A_3}Z_{A_4}X_{B_3}Z_{B_4}Z_{C_2}Z_{C_3}Z_{C_4}$	1
$S_{C_1}S_{A_3}$	$Z_{A_1}Z_{A_2}X_{A_3}Z_{B_1}Z_{B_2}Z_{B_3}Z_{B_4}X_{C_1}Z_{C_2}Z_{C_3}$	1
$S_{A_2}S_{B_2}S_{C_1}$	$X_{A_2}X_{B_2}X_{B_1}Z_{B_3}X_{C_1}$	-1
$S_{A_3}S_{B_4}S_{C_3}$	$X_{A_3}X_{B_4}X_{C_3}Z_{C_2}Z_{C_4}$	-1

Table 2.3: XZ Reference measurements

$$X'_v \left|\Psi'\right\rangle = (Z^{N(v)})' \left|\Psi'\right\rangle \tag{2.3.6}$$

Then, following equations (2.3.3) and (2.3.4) we can see that:

$$\left\langle\Psi'\right| X'_v Z'_v \left|\Psi'\right\rangle = 0 \tag{2.3.7}$$

$$\left\langle\Psi'\right| X'_v D'_v \left|\Psi'\right\rangle = \frac{1}{\sqrt{2}} \tag{2.3.8}$$

Now, given the satisfied expectation values from Table 2.2 we obtain

$$Z'_v \left|\Psi'\right\rangle = (M_A)'(M_B)' \left|\Psi'\right\rangle \tag{2.3.9}$$

and following equation (2.3.5)

$$\left\langle\Psi'\right| Z'_v D'_v \left|\Psi'\right\rangle = \frac{1}{\sqrt{2}} \tag{2.3.10}$$

Having established these relations we can see that the vectors $X'_v \left|\Psi'\right\rangle$ and $Z'_v \left|\Psi'\right\rangle$ are orthogonal and the vector $D'_v \left|\Psi'\right\rangle$ on them have projections $1/\sqrt{2}$. From this it can be concluded that

$$D'_v = \frac{X'_v + Z'_v}{\sqrt{2}} \tag{2.3.11}$$

From this equation and unitarity and Hermicity of D'_v, X'_v and Z'_v the anti-commuting relation can be found:

$$\begin{aligned} \mathbb{1} = (D'_v)^2 = \frac{1}{2}(X'_v + Z'_v)(X'_v + Z'_v) = \\ \frac{1}{2}(\mathbb{1} + X'_v Z'_v + Z'_v X'_v + \mathbb{1}) \\ \Rightarrow X'_v Z'_v + Z'_v X'_v = 0 \end{aligned} \tag{2.3.12}$$

□

Because of the reasons given in previous discussion, even if not necessary for the self-test, the F-measurements have to be included. Using the same procedure like for the measurements including D, from the correct expectation values it can be concluded that $F' |\Psi'\rangle = \frac{X'-Z'}{\sqrt{2}} |\Psi'\rangle$.

Before proceeding to the isometry construction we have to deal with some problems related to the fact that a single prover holds more than one qubit. All that is said up to now is not essentially different from the self-test corresponding to the situation in which each qubit is held by a separate prover. These problems arising in this new scenario are:

(a) $(Z_{V_u} Z_{V_v})' |\Psi'\rangle \neq Z'_{V_u} Z'_{V_v} |\Psi'\rangle$ in a general case, where $V \in \{A, B, C\}$ and $u, v \in \{1, 2, 3, 4\}$.

(b) In a general case $[X'_{V_u}, X'_{V_v}] |\Psi'\rangle \neq 0$ where $V \in \{A, B, C\}$ and $u, v \in \{1, 2, 3, 4\}$ unless $u = v$.

(c) In a general case $[Z'_{V_u}, Z'_{V_v}] |\Psi'\rangle \neq 0$ where $V \in \{A, B, C\}$ and $u, v \in \{1, 2, 3, 4\}$ unless $u = v$.

(d) In a general case $[X'_{V_u}, Z'_{V_v}] |\Psi'\rangle \neq 0$ where $V \in \{A, B, C\}$ and $u, v \in \{1, 2, 3, 4\}$ and $u \neq v$.

The enumerated problems seem to be very troublesome, but we will try to give satisfactory solutions for all of them. The first thing we would like to convince ourselves is that, for example, $(Z_{B_3} Z_{B_4})' |\Psi'\rangle = Z'_{B_3} Z'_{B_4} |\Psi'\rangle$. This can easily be done by making the measurement process sequential. The prover will be first asked to measure the observable Z_{B_3} and after reporting the measurement result it will be commanded to measure the observable Z_{B_4}. However, one should be careful here because the prover have measured Z'_{B_4} after being asked to measure Z_{B_3}. It doesn't have to be the same measurement it performs when it is asked to measure just Z_{B_4}. To distinguish these two cases the following notation for sequential measurements will be used: $(Z_{B_3} Z_{B_4})' |\Psi'\rangle = Z'_{B_3} (Z'_{B_4})^{Z_{B_3}} |\Psi'\rangle$ and in a general case $(Z'_{B_4})^{Z_{B_3}} |\Psi'\rangle \neq Z'_{B_4} |\Psi'\rangle$. Even though this problem might seem cumbersome the solution for it can be obtained with one nice trick. Assume that the verifier wants to check the expectation value of S_{A_2}. In this situation the prover named Alice has to measure just X_{A_2}. However, to hide the real intentions the verifier can first command the measurement of X_{A_1} and afterwards X_{A_2}. After getting this command the prover would first measure the observable X'_{A_1} and then $(X'_{A_2})^{X_{A_1}}$ Other provers measure what is needed to reproduce the expectation value of S_{A_2} and upon collecting all the results verifier just discards the measurement results obtained on the qubit A_1 and calculates the expectation value $\left\langle S'_{A_2} \right\rangle$. If $X'_{A_2} \neq (X'_{A_2})^{X_{A_1}}$ the expectation value would be wrong and the provers would fail the self test. This strategy imposes a little bit different reference measurements than those given in Tables 1 and 2 and, of course, adds a little bit of post-processing of the collecting the results. Having all this in mind the verifier constructs the following strategy:

- the provers are commanded to perform the measurements given in Tables 2.1, 2.2, 2.3 and equations (2.3.3), (2.3.4), (2.3.5).
- the stabilizers from Tables 2.1 and 2.2 are such that at least one prover (say Alice) performs a measurement on the only one of his four qubits. When checking these stabilizers the verifier tosses a coin.
  - If he gets a head Alice is commanded to perform a single qubit measurement and the process is done. This is just the measurement equivalent to the one verifier would command in case when all the qubits are held by a separate party:

$$\left\langle S'_{A_i} \right\rangle = \left\langle X'_{A_i} Z'^{N(A_i)} \right\rangle \tag{2.3.13}$$

– If he gets a tail he looks for a suitable stabilizer in which Bob measures the same qubit in the same basis, but apart from that measures some more of his qubits. The verifier commands the measurement of other qubits like he would do when checking that other stabilizer, but discards the reported outcomes, except for the one outcome necessary for the calculation of the expectation value of the original stabilizer.

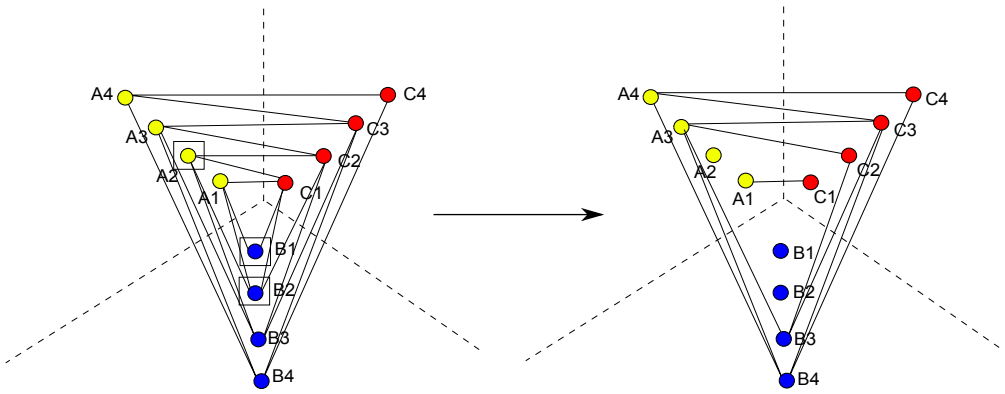
$$\left\langle S'_{A_i} \right\rangle = \left\langle (\sum P_x^{X'_{A_j}}) X'_{A_i} Z'^{N(A_i)} \right\rangle \tag{2.3.14}$$

$$\left\langle (S_{A_j} S_{A_i})' \right\rangle = \left\langle X'_{A_j} X'_{A_i} Z'^{N(A_j)} Z'^{N(A_i)} \right\rangle \tag{2.3.15}$$

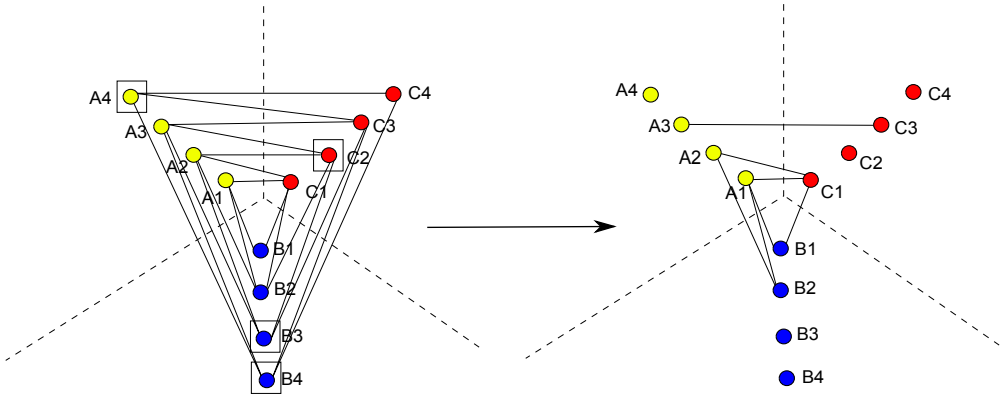
where $P_x^{X'_{A_j}}$ is the projector to the eigenspace of X'_{A_j} corresponding to the eigenvalue x. In the equation (2.3.14) it is implied that discarding of the measurement outcomes upon measuring the observable X'_{A_j} in the expectation value can be written as the action of the sum of projectors $\sum P_x^{X'_{A_j}}$, while the expectation value of the usual measurement can be written as the action of the weighted sum of projectors $\sum_x P_x^{X'_{A_j}}$. From elementary linear algebra we know that $\sum P_x^{X'_{A_j}} = \mathbb{1}$ while $\sum_x P_x^{X'_{A_j}} = X'_{A_j}$. The measurements from Table 2.3 are helpful in this part since they represent stabilizers (or triangle-measurements) in which one party measures both Z and X, and not just X or just Z as in Tables 2.1 and 2.2.

Introducing sequentiality in the measurement process, however, is not without consequences. If one measures the observable M on the state $|\phi\rangle$ and upon getting the outcome measures the observable N, from the measurement outcomes it is, in principle, not possible to retrieve the expectation value $\langle\phi| NM |\phi\rangle$. This is due to the fact that the second measurement is performed on the state into which $|\phi\rangle$ collapsed upon the first measurement. The special case in which sequantiality does not jeopardize the calculation of the expectation value $\langle\phi| NM |\phi\rangle$ by simple collection of the results of the individual measurements is when N and M commute. For our case this means that in order to perform a self-test consisted of sequential measurements we have to be sure that all the operators that supposedly act on different qubits held by the same prover commute, what is actually the content of the enumerated problems (b), (c) and (d). In the construction of the isometry we will see that even without sequential measurements, the commutation relation $[M_u, N_v] |\Phi'\rangle = 0$, with $M, N \in \{X, Z, D\}$, $u, v \in \{1, 2, 3, 4\}$ and $u \neq v$, must be satisfied. The most efficient way to prove the commuting relations is by proving that operators M_u and N_v act on the separate Hilbert spaces corresponding to qubits u and v, respectively. For this purpose we can use the theorem about rigidity of the CHSH game from [RUV12] (See section 1.7). The authors have proven that two provers can win n sequential CHSH games with probability $n\omega$, where ω is the probability to win the game by using optimal quantum strategy, only if they hold tensor product of n EPR pairs tensored with some state of arbitrary dimension and on each of them apply the measurements from the optimal strategy for winning the single CHSH game. One EPR pair can be extracted from our 12-qubit triangular cluster state by performing Z measurements on certain qubits. In section 1.4 we said that measuring some qubit of a graph state in the Z-basis effectively removes it from the other qubits, leaving the rest of the graph state unchanged. This means that by performing certain Z-measurements one can erase some edges from the graph and, in general, it is possible to extract one entangled pair of qubits (Fig. 2.8a and 2.8b). Moreover it is possible to extract a tensor product of two EPR pairs, tensored with the state of the other qubits (Fig. 2.8c)

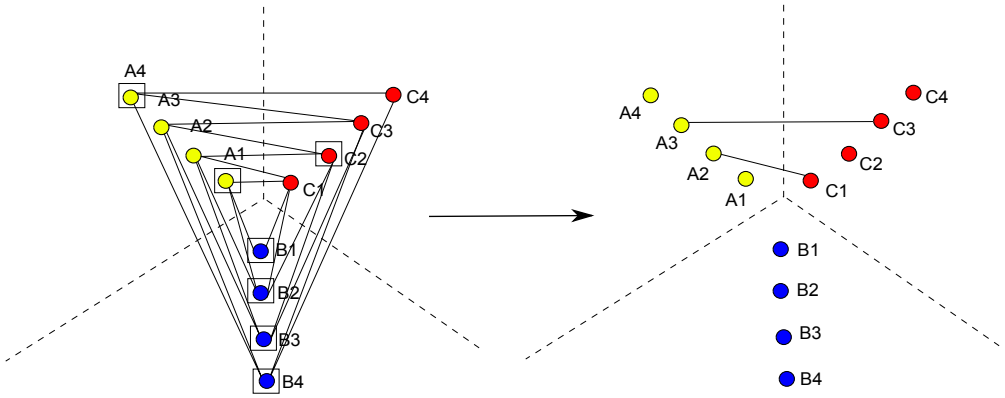
With the established procedure to extract one or more EPR pairs from our triangular cluster state we can introduce the addition to the self-test based on the CHSH game and its properties.



(a) Extracting the EPR pair consisting of the qubits A1 and C1. Every qubit with a square around it is measured in the Z-basis and thus disentangled from the other qubits. All the edges connecting a Z-measured qubit with any other qubit are simply erased.



(b) Extracting the EPR pair consisting of the qubits A3 and C3. The qubits A4, C2, B3 and B4 are measured in the Z-basis.



(c) Extracting the tensor product of two EPR pairs.

Figure 2.8: The process of extracting EPR pairs by performing corresponding Z-measurements.

The provers (Alice, Bob and Charlie) will not choose the strategy, they will not even be aware that they are playing the CHSH game. They will be commanded to perform measurements from the optimal strategy to win the CHSH game, and if they "win" the game the verifier can be sure that they perform measurements as commanded and that they possess the advertised state. Because the verifier and the provers have in interest to win the CHSH game and strategy is chosen by the verifier we will call this unconventional game the reciprocal CHSH game. The protocol for playing the game is as follows:

- The verifier commands to the provers to perform Z measurements on some of their qubits. If they are honest after the measurement two provers will share one or two EPR pairs. Without loss of generality we will assume that Alice and Charlie end up sharing the maximally entangled state(s).

- Alice is commanded to perform Z (corresponding to the input 0) or X (corresponding to the input 1) measurement on her part of the EPR pair, while Charlie is commanded to perform D (input 0) or F (input 1) measurement on his qubit. If they share two EPR pairs both get commands to measure two qubits in corresponding bases. By the inputs what to measure Alice and Charlie cannot know that the usual self-testing procedure (or the computation) is being conducted.

- The verifier collects the outputs and checks if the condition to win the reciprocal CHSH game is satisfied. If the percentage of won reciprocal CHSH games on a single EPR pairis smaller than ω times, and if the percentage of won reciprocal CHSH games on two tensored EPR pairs is smaller than 2ω, the provers have failed to pass the self-test.

Let us now analyze the above described process in more details taking into account the results of [RUV12]. The rigidity of CHSH game tells that Alice and Charlie when having two tensored EPR pairs have the highest probability to win two sequential CHSH games if they apply optimal measurements on the subspaces corresponding to each qubit. This means that the double reciprocal CHSH game will be won if each prover when commanded to measure some observable on the qubit v measures in the Hilbert space corresponding to the qubit v and possibly some ancillas but not in the Hilbert space corresponding to the other qubits of the triangular cluster state. Since provers, by the input they get, do not know when they are playing the reciprocal CHSH game they always have to follow the instructions about the Hilbert space in which the measurements should be performed. Also, as Z-measurements are valid inputs of the reciprocal CHSH game the provers cannot discriminate when they are asked to disentangle the qubit and when the game is actually being played. The reciprocal CHSH game has some even stronger consequences. One trivial reason not to win the game is if the provers do not possess the maximally entangled pair of qubits. The verifier commands the provers to perform Z-measurements in order to disentangle the qubits that are supposed to play the reciprocal CHSH game from all other qubits. If the physical state contains more edges than the reference state in some of the reciprocal CHSH games redundant edge will stay unerased and the qubits involved in the game will not be maximally entangled, which will lead to the failure to pass the self-test. This is due to the monogamy of entanglement, property that forbids two maximally entangled qubits to be entangled to anything else [CKW00]. If one of two entangled qubits, A and B, is entangled with the third qubit C then A and B cannot be maximally entangled. Every edge from the reference state should be involved in at least one reciprocal CHSH game, so if the physical state has less edges than the reference state it will not be possible to win every game with demanded probability. Thus, introducing the reciprocal CHSH games allows the verifier to check if the provers share the claimed state.

Before passing to the isometry construction there is one important point to be made about the CHSH game. The described measurements represent optimal strategy when the provers

share the EPR pair $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. Z-measurements can induce local Z operations on all adjacent qubits if the measurement outcome is 1. This means that upon appropriate Z-measurements the provers can share one of the four Bell states (see equations (1.1.4) to (1.1.6)), so according to the extracted state the verifier has to adapt strategy for playing the reciprocal CHSH game.

- **Theorem 2.1** If the physical experiment simulates the reference experiments then they are also equivalent.

Proof. With having established all the necessary commuting and anti-commuting relations we can start the construction of the isometry. The scheme of the isometry is given on the Figure 2.9. It is just the generalized swap gate, where each party performs swap on four qubits it holds.

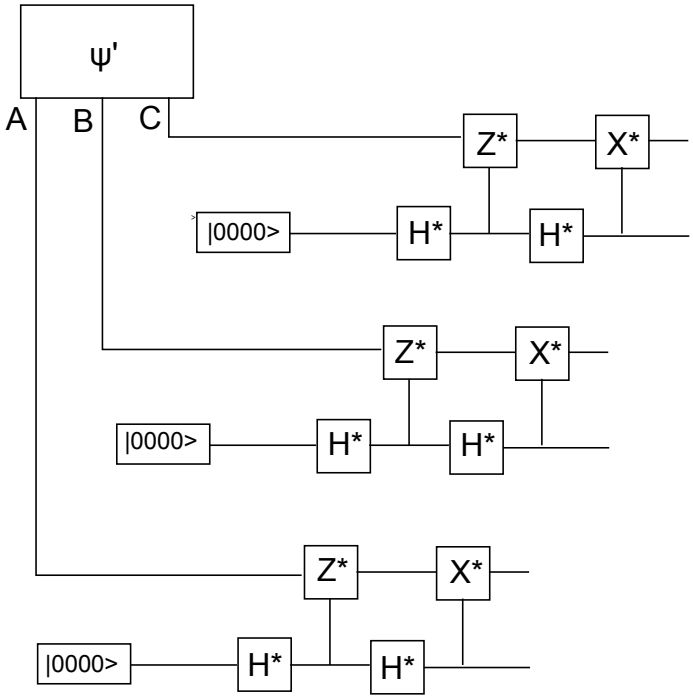


Figure 2.9: The local isometry

H^* is the tensor product of four single-qubit Hadamards:

$$H^* = H \otimes H \otimes H \otimes H \tag{2.3.16}$$

This generalized Hadamard acts as follows:

$$\begin{aligned} H^* |0000\rangle &= \frac{1}{2^2} \sum_{i,j,k,l \in \{0,1\}} |ijkl\rangle \\ H^* |i'j'k'l'\rangle &= \frac{1}{2^2} \sum_{i,j,k,l \in \{0,1\}} (-1)^{ii'+jj'+kk'+ll'} |ijkl\rangle \end{aligned} \tag{2.3.17}$$

Controlled gates denoted as CZ^* and CX^* are generalized CZ and $CNOT$:

$$|\mathbf{p}\rangle = |p_1 p_2 p_3 p_4\rangle \tag{2.3.18}$$

$$CZ^* |\mathbf{p}\rangle |\psi\rangle = |\mathbf{p}\rangle \prod_{i=1}^{4} \otimes Z'^{p_i}_i |\psi\rangle \tag{2.3.19}$$

$$CX^* |\mathbf{p}\rangle |\psi\rangle = |\mathbf{p}\rangle \prod_{i=1}^{4} \otimes X'^{p_i}_i |\psi\rangle \tag{2.3.20}$$

With these definitions the part of the swap-gate performed by a single party is:

$$\begin{aligned}
\Phi_A |0000\rangle |\Psi'\rangle = CX^* H^* CZ^* H^* |0000\rangle |\Psi'\rangle = \\
= CX^* H^* CZ^* \frac{1}{2^2} \sum_{i=0}^{1}\sum_{j=0}^{1}\sum_{k=0}^{1}\sum_{l=0}^{1} |ijkl\rangle |\Psi'\rangle = \\
= CX^* H^* \frac{1}{2^2} \sum_{i=0}^{1}\sum_{j=0}^{1}\sum_{k=0}^{1}\sum_{l=0}^{1} |ijkl\rangle Z'^{\,i}_1 Z'^{\,k}_2 Z'^{\,k}_3 Z'^{\,l}_4 |\Psi'\rangle = \\
= CX^* \frac{1}{2^4} \sum_{i'=0}^{1}\sum_{j'=0}^{1}\sum_{k'=0}^{1}\sum_{l'=0}^{1} |i'j'k'l'\rangle \sum_{i=0}^{1}\sum_{j=0}^{1}\sum_{k=0}^{1}\sum_{l=0}^{1} (-1)^{ii'+jj'+kk'+ll'} Z'^{\,i}_1 Z'^{\,k}_2 Z'^{\,k}_3 Z'^{\,l}_4 |\Psi'\rangle = \\
= \frac{1}{2^4} \sum_{i'=0}^{1}\sum_{j'=0}^{1}\sum_{k'=0}^{1}\sum_{l'=0}^{1} |i'j'k'l'\rangle X'^{\,i'}_1 X'^{\,j'}_2 X'^{\,k'}_3 X'^{\,l'}_4 \sum_{i=0}^{1}\sum_{j=0}^{1}\sum_{k=0}^{1}\sum_{l=0}^{1} (-1)^{ii'+jj'+kk'+ll'} Z'^{\,i}_1 Z'^{\,k}_2 Z'^{\,k}_3 Z'^{\,l}_4 |\Psi'\rangle
\end{aligned} \tag{2.3.21}$$

The full swap-gate is given as a tensor product of three swap-gates performed by a single party[2]:

$$\Phi |000000000000\rangle |\Psi'\rangle = \frac{1}{2^{12}} \sum_{i_1,\ldots,i_{12}\in\{0,1\}} |i_1 i_2 \ldots i_{12}\rangle X'^{i_1}_{A_1} \ldots X'^{i_{12}}_{C_4} \sum_{i'_1,\ldots,i'_{12}\in\{0,1\}} (-1)^{i_1 i'_1 + \ldots + i_{12} i'_{12}} Z'^{i'_1}_{A_1} \ldots Z'^{i'_{12}}_{C_4} |\Psi'\rangle \tag{2.3.22}$$

[3]

Knowing that $[X'_m, Z'_n] = 0$ for $m \neq n$, otherwise $\{X'_m, Z'_n\} = 0$ we can perform the following simplification:

$$\begin{aligned}
X'^{i_1}_{A_1} \ldots X'^{i_{12}}_{C_4} \sum_{i'_1,\ldots,i'_{12}\in\{0,1\}} (-1)^{i_1 i'_1 + \ldots + i_{12} i'_{12}} Z'^{i'_1}_{A_1} \ldots Z'^{i'_{12}}_{C_4} |\Psi'\rangle = \\
= \sum_{i'_1,\ldots,i'_{12}\in\{0,1\}} Z'^{i'_1}_{A_1} \ldots Z'^{i'_{12}}_{C_4} X'^{i_1}_{A_1} \ldots X'^{i_{12}}_{C_4} |\Psi'\rangle \\
= \Omega X'^{i_1}_{A_1} \ldots X'^{i_{12}}_{C_4} |\Psi'\rangle
\end{aligned} \tag{2.3.23}$$

where in the last line the following notation was used:

$$\Omega = \sum_{i'_1,\ldots,i'_{12}\in\{0,1\}} Z'^{i'_1}_{A_1} \ldots Z'^{i'_{12}}_{C_4} \tag{2.3.24}$$

This allows us to rewrite equation (2.3.22):

[2]The following correspondence between qubits notation holds $i_1 \Leftrightarrow A_1, i_2 \Leftrightarrow A_2, i_3 \Leftrightarrow A_3, i_4 \Leftrightarrow A_4, i_5 \Leftrightarrow B_1, i_6 \Leftrightarrow B_2, i_7 \Leftrightarrow B_3, i_8 \Leftrightarrow B_4, i_9 \Leftrightarrow C_1, i_{10} \Leftrightarrow C_2, i_{11} \Leftrightarrow C_3, i_{12} \Leftrightarrow C_4$

[3]In the further text we will use the following notation $|000000000000\rangle = |0\ldots0\rangle$

$$\Phi\left|0...0\right\rangle\left|\Psi'\right\rangle=\frac{1}{2^{12}}\sum_{i_1,...,i_{12}\in\{0,1\}}\left|i_1i_2...i_{12}\right\rangle\Omega X'^{i_1}_{A_1}...X'^{i_{12}}_{C_4}\left|\Psi'\right\rangle \tag{2.3.25}$$

In the next step we would like to get rid of X's from equation (2.3.25). For this we can use the expectation values of the stabilizers:

$$X'_{A_i}\left|\Psi'\right\rangle=Z'^{N(A_i)}\left|\Psi'\right\rangle \tag{2.3.26}$$

When dealing with the terms containing products of X's (X-products in the following text) operating on the qubits that belong to different parties we will differentiate two cases:

(a) the product does not contain X's acting on adjacent qubits;

(b) the product contains X's acting on one or more pairs of adjacent qubits.

In the case (a) the X-products acting on $\left|\Psi'\right\rangle$ can simply be written as the Z-products acting on the same state. We can see this case like the sequential application of equation (2.3.26). In this generalization of (2.3.26) the set of qubits on which X's act and the set of qubits on which Z's act are disjoint. The case (b) can be dealt with by sequentially applying the rule from (2.3.26), but one has to be careful. To demonstrate the uniqueness of this case we will treat the product $X'_{A_1}X'_{B_2}$. As explained we will first use the stabilizer S'_{B_2} to get rid of X'_{B_2}:

$$X'_{A_1}X'_{B_2}\left|\Psi'\right\rangle=X'_{A_1}\left(Z'_{A_1}Z'_{A_2}Z'_{C_1}Z'_{C_2}\left|\Psi'\right\rangle\right) \tag{2.3.27}$$

To propagate X'_{A_1} to the right it has to exchange place with Z'_{A_1} and because these two observables anti-commute a minus sign will appear in the expression. To summarize, if the $\zeta\left(i_1,...,i_{12}\right)$ is the number of edges connecting the qubits on which X's from some X-product act,[4] the following equation holds:

$$\prod_i X'_i\left|\Psi'\right\rangle=(-1)^{\zeta(i_1,...,i_{12})}\prod_i Z'_i\left|\Psi'\right\rangle \tag{2.3.28}$$

It is easy to check that following identity is satisfied (for example multiplying term by term):

$$\begin{aligned}Z'^iZ'^jZ'^kZ'^l\Omega=\Omega Z'^iZ'^jZ'^kZ'^l=\Omega,\\ \forall i,j,k,l\in\{0,1\}\end{aligned} \tag{2.3.29}$$

Now we have all ingredients to prove that $\Phi\left|\Psi'\right\rangle=\left|\Psi\right\rangle\otimes\left|junk\right\rangle$. Taking into account equations (2.3.28) and (2.3.29), we can rewrite equation (2.3.25) in a rather simple way:

$$\Phi\left|0...0\right\rangle\left|\Psi'\right\rangle=\frac{1}{2^{12}}\sum_{i_1,...,i_{12}\in\{0,1\}}(-1)^{\zeta(i_1,...,i_{12})}\left|i_1i_2...i_{12}\right\rangle\Omega\left|\Psi'\right\rangle \tag{2.3.30}$$

$$\Phi\left|0...0\right\rangle\left|\Psi'\right\rangle=\left(\frac{1}{2^6}\sum_{i_1,...,i_{12}\in\{0,1\}}(-1)^{\zeta(i_1,...,i_{12})}\left|i_1i_2...i_{12}\right\rangle\right)\otimes(\frac{1}{2^6}\Omega\left|\Psi'\right\rangle) \tag{2.3.31}$$

The state $\frac{1}{2^6}\sum_{i_1,...,i_{12}\in\{0,1\}}(-1)^{\zeta(i_1,...,i_{12})}\left|i_1i_2...i_{12}\right\rangle$ is the reference state $\left|\Psi\right\rangle$. It is clear if we think in the way the 12-qubit triangular state is formed. At the begining all the qubits are in the state $\left|+\right\rangle$. The ensor product of these states written in the computational basis is the sum of 2^{12} terms with the correct normalization factor $\frac{1}{2^6}=\frac{1}{\sqrt{2}^{12}}$. To produce the graph state CZ gates are applied on the qubits connected by an edge. When CZ acts on two qubits it creates

[4] $\zeta\left(i_1,...,i_{12}\right)$ for the state $\left|i_1...i_{12}\right\rangle$ also corresponds to the number of edges connecting the qubits which are in the state $\left|1\right\rangle$.

a minus sign only when both qubits are in the state $|1\rangle$. Thus the sign in front of the term $|i_1...i_{12}\rangle = |i_1\rangle ... |i_{12}\rangle$ depends on the number of edges connecting all qubits that are in the state $|1\rangle$ in that term. Footnote 3 tells that this number is exactly $\zeta(i_1, ..., i_{12})$, so we can write:

$$|\Psi\rangle = \left(\frac{1}{2^6} \sum_{i_1,...,i_{12} \in \{0,1\}} (-1)^{\zeta(i_1,...,i_{12})} |i_1 i_2 ... i_{12}\rangle \right) \tag{2.3.32}$$

This completes the first part of the proof, since we have shown that $\Phi |\Psi'\rangle = |\Psi\rangle \otimes |junk\rangle$ where $|junk\rangle = \frac{1}{2^6} \Omega |\Psi'\rangle$.

Now we want to prove that $\Phi(M'_m |\Psi'\rangle) = M_m |\Psi\rangle \otimes |junk\rangle$, where $M' \in \{X', Z', D', F'\}$ and m is any vertex.

First we will consider the expression (taking into account the equation (2.3.22)):

$$\begin{aligned} \Phi\left(Z'_j |\Psi'\rangle\right) = \Phi |0...0\rangle |\Psi'\rangle = \frac{1}{2^{12}} \sum_{i_1,...,i_{12} \in \{0,1\}} |i_1 i_2 ... i_{12}\rangle X'^{i_1}_{A_1} ... X'^{i_{12}}_{C_4} ... \times \\ ... \times \sum_{i'_1,...,i'_{12} \in \{0,1\}} (-1)^{i_1 i'_1 + .. + i_{12} i'_{12}} Z'^{i'_1}_{A_1} ... Z'^{i'_{12}}_{C_4} Z'_j |\Psi'\rangle \end{aligned} \tag{2.3.33}$$

This equation can further be simplified on the same way as we did with equation (2.3.22). The only difference is that terms in the sum corresponding to the X-products containing X_j will have additional minus sign appearing because $X'_j Z'_j = -Z'_j X'_j$. The sign corresponding to the X-products not containing X'_j remains unchanged. This is equivalent to deleting all the edges connecting qubit denoted by j with other qubits. From previous discussion we know that measuring Z on a certain qubit effectively removes it with all its edges from the graph state. The explained effect of the Z'_j measurement can be written on the following way (using equation (2.3.33)):

$$\Phi\left(Z'_j |\Psi'\rangle\right) = \left(\frac{1}{2^6} \sum_{i_1,...,i_{12} \in \{0,1\}} (-1)^{\zeta(i_1,...,i_j \wedge 0,...,i_{12})} |i_1 i_2 ... i_{12}\rangle \right) \otimes \left(\frac{1}{2^6} \Omega |\Psi'\rangle\right) \tag{2.3.34}$$

where $\wedge$ is the logical AND. On the other hand, from (2.3.32) we can get:

$$Z_j |\Psi\rangle = \left(\frac{1}{2^6} \sum_{i_1,...,i_j \oplus 1,...,i_j \wedge 0,...,i_{12} \in \{0,1\}} (-1)^{\zeta(i_1,...,i_{12})} |i_1 i_2 ... i_{12}\rangle \right) \tag{2.3.35}$$

since the action of Z_j consists on the adding a minus sign in front of the terms in which a qubit j is in the state $|1\rangle$, and leaving the unchanged sign of the terms in which j is in the state $|0\rangle$. By simple comparison of equations (2.3.34) and (2.3.35) we can conclude that:

$$\Phi\left(Z'_j |\Psi'\rangle\right) = Z_j |\Psi\rangle \otimes |junk\rangle \tag{2.3.36}$$

This can easily be generalized on the case of arbitrary Z-product acting on $|\Psi'\rangle$.

To see how X'_j acts on the physical state $|\Psi'\rangle$, like in the case of Z'_j we will put it into equation (2.3.22):

$$\Phi\left(X'_j\left|\Psi'\right\rangle\right) = \Phi\left|0...0\right\rangle\left|\Psi'\right\rangle = \frac{1}{2^{12}}\sum_{i_1,...,i_{12}\in\{0,1\}}\left|i_1 i_2 ... i_{12}\right\rangle X'^{i_1}_{A_1}...X'^{i_{12}}_{C_4}...\times$$
$$... \times \sum_{i'_1,...,i'_{12}\in\{0,1\}}(-1)^{i_1 i'_1+..+i_{12}i'_{12}} Z'^{i'_1}_{A_1}...Z'^{i'_{12}}_{C_4} X'_j\left|\Psi'\right\rangle \tag{2.3.37}$$

The presence of X'_j will change the simplification of (2.3.37) compared to (2.3.22) in a way that X-products not containing X'_j will effectively gain it, while those containing X'_j will lose it. We can write it in the following way:

$$\Phi\left(X'_j\left|\Psi'\right\rangle\right) = \left(\frac{1}{2^6}\sum_{i_1,...,i_{12}\in\{0,1\}}(-1)^{\zeta(i_1,...,i_j\oplus 1,...,i_{12})}\left|i_1 i_2 ... i_{12}\right\rangle\right)\otimes\left(\frac{1}{2^6}\Omega\left|\Psi'\right\rangle\right) \tag{2.3.38}$$

where $\oplus$ is addition modulo 2. On the other hand, when X_j acts on $\left|\Psi\right\rangle$ written in the computational basis it acts just as a bit-flip for the qubit j:

$$X_j\left|\Psi\right\rangle = \left(\frac{1}{2^6}\sum_{i_1,...,i_j\oplus 1,...,i_j\oplus 0,...,i_{12}\in\{0,1\}}(-1)^{\zeta(i_1,...,i_{12})}\left|i_1 i_2 ... i_{12}\right\rangle\right) \tag{2.3.39}$$

By comparing equations (2.3.38) and (2.3.39) we can conclude that

$$\Phi\left(X'_j\left|\Psi'\right\rangle\right) = X_j\left|\Psi\right\rangle\otimes\left|junk\right\rangle \tag{2.3.40}$$

Since we proved that $D'_j = \frac{X'_j+Z'_j}{\sqrt{2}}$ and $F'_j = \frac{X'_j-Z'_j}{\sqrt{2}}$, from the linearity it follows that:

$$\Phi\left(D'_j\left|\Psi'\right\rangle\right) = D_j\left|\Psi\right\rangle\otimes\left|junk\right\rangle \tag{2.3.41}$$
$$\Phi\left(F'_j\left|\Psi'\right\rangle\right) = F_j\left|\Psi\right\rangle\otimes\left|junk\right\rangle \tag{2.3.42}$$
$$\tag{2.3.43}$$

This completes the proof of Theorem 2.1 □

Chapter 3

Summary and Outlook

This thesis presents the initial results from the efforts to construct a simple and implementable interactive proof aiming to certify the true quantum behavior of some, supposedly, quantum device. In Chapter 1 we tried to give concise theoretical introduction to the subject. This chapter did not include any new results. The first aim was to give the motivation for doing this kind of research and point out its significance. The second aim was to present the main physical and mathematical concepts necessary for solving the problem (graph states, 1WQC, nonlocality) as well as to shortly explain the most important previous results (self-testing and interactive proofs). Nonlocality is the corner-stone concept that enables the whole construction. In the first chapter it is pointed out either explicitly (1.5) or implicitly (1.3, 1.4, 1.6). This central role of nonlocality when it comes to distinguishing classical from quantum comes from the famous Bell's theorem, which marks nonlocality as a quantum imprint. The most important previous result is the interactive proof developed by McKague [McK13]. This interactive proof involves one classical user/computer (verifier) and n quantum devices (provers). The verifier can convince himself that the provers exhibit truly quantum behavior and besides that by classical communication with provers it can perform universal quantum computation.

In Chapter 2 the idea to reduce number of necessary provers is developed. The results presented in this chapter are mainly new. The first attempt was to construct a self-test (and subsequently an interactive proof) for the case when one classical verifier is interacting with two quantum provers. Section 2.1 shows the proof that nonlocality condition is satisfied and in principle a self-test can be constructed. Following this result the attempts to actually construct the self-test for the case with two provers are presented in the section 2.2. Unfortunately, a valid self-test could not be constructed, since it appears to be difficult to self-test entanglement of two qubits held by the same prover. However, this section bears certain significance, since it does not bring a definite no-go statement. The procedure is explained in detail and the points where the procedure gets stuck are clearly pointed out. The possible ways out are emphasized as well. In section 2.3 the initial idea is relaxed in two ways. On one hand the number of provers is increased on 3, and on the other the system (verifier + provers) does not perform universal quantum computing, but only single qubit unitaries. These simplifications allowed for the construction of an interactive proof. This new construction uses some properties of two previous interactive proofs [RUV13, McK13]. Compared to the interactive proof presented in [McK13] where each qubit is held by a separate prover we have paid a special attention to locate all the problems arising from the fact that one prover holds more than one qubit. To certify the honesty of provers we have made the measurement process sequential and based on the properties of the CHSH games and its usefulness in self-testing scenario, presented in [RUV13], we have introduced an addition to the self-test called the reciprocal CHSH game.

The results presented in this thesis allow for further development. The three provers inter-

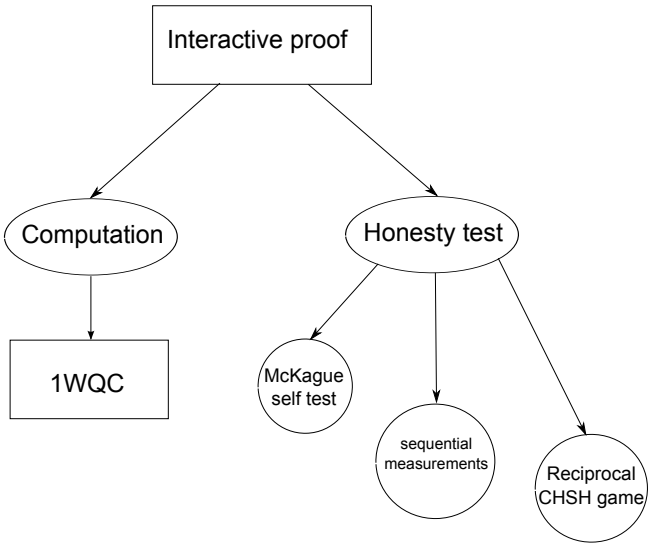


Figure 3.1: The general scheme showing all the parts included in the new interactive proof presented in this thesis

active proof should include estimations on robustness of the system. It would be very good if the interactive proof can be extended so the system 'verifier + 3 provers' can perform universal quantum computing and not just single qubit unitaries. Rigidity of CHSH game and the introduced reciprocal CHSH game can demonstrate more power on the case when one prover possesses more qubits. Some ideas from section 2.3 might be reused for the possible two-provers self-test of a graph state containing odd cycles as well as the other ideas presented in section 2.2. One possible direction for future research can be construction of a self-test that would allow the use of measurements from the YZ-plane or the XY-plane of the Bloch sphere. There are many ways to improve the existing interactive proofs and self-tests and for sure as time goes the interest of scientific community in these problems will increase.

Bibliography

[Aar13] Scott Aaronson. D-wave: Truth finally starts to emege, 2013.

[ABG+07] A. Acin, N. Brunner, N. Gisin, S. Massar, S. Pironio, and V. Scarani. Device-independent security of quantum cryptography against collective attacks. *Phys. Rev. Lett.*, 98(23):230501, Jun 2007.

[BB06] Dan E Browne and Hans J Briegel. One-way quantum computation-a tutorial introduction. *arXiv preprint quant-ph/0603226*, 2006.

[BBGL11] Jean-Daniel Bancal, Nicolas Brunner, Nicolas Gisin, and Yeong-Cherng Liang. Detecting genuine multipartite quantum nonlocality: A simple approach and generalization to arbitrary dimensions. *Physical Review Letters*, 106(2):020405, January 2011.

[BCP+14] Nicolas Brunner, Daniel Cavalcanti, Stefano Pironio, Valerio Scarani, and Stephanie Wehner. Bell nonlocality. *Reviews of Modern Physics*, 86(2):419–478, April 2014.

[Bel04] J. S. Bell. *Speakable and Unspeakable in Quantum Mechanics: Collected Papers on Quantum Philosophy.* Cambridge University Press, 2. edition, June 2004.

[BJS11] Michael J. Bremner, Richard Jozsa, and Dan J. Shepherd. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 467(2126):459–472, February 2011.

[BK05] S. D. Barret and P. Kok. Efficient high-fidelity quantum computation using matter qubits and linear optics. *Phys. Rev. A*, 71(060310), 2005.

[BMP+00] P. O. Boykin, T. Mor, M. Pulver, V. Roychowdhurry, and F. Vatan. A new universal and fault-tolerant quantum basis. *Infrormation Processing Letters*, (75(3)):101–107, August 2000.

[BNS+14] J. D. Bancal, M. Navascues, V. Scarani, T. Vertesi, and T. H. Yang. Opening the black box: physical characterization from nonlocal correlations. preprint, July 2014.

[BR01] H. J. Briegel and R. Raussendorf. A one-way quantum computer. *Physical Review Letters*, 86(22):5188–5191, May 2001.

[BV04] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization.* Cambridge University Press, 2004.

[CH74] John F. Clauser and Michael A. Horne. Experimental consequences of objective local theories. *Phys. Rev. D*, 10(2):526–535, Jul 1974.

[CHSH69] J. F. Clauser, M. A. Horne, A. Shimony, and R. A. Holt. Proposed experiment to test local hidden-variable theories. *Physical Review Letters*, 23:880, 1969.

[CKW00] Valerie Coffman, Joydip Kundu, and William K Wootters. Distributed enanglement. *Phys. Rev. A*, 61(5):052306, April 2000.

[CL07] Kai Chen and Hoi-Kwong Lo. Multi-partite quantum cryptographic pprotocol with nopisy ghz states. *Quantum Information and Computation*, 7(8):689–715, 2007.

[Deu85] David Deutsch. The church-turing principle and the universal quantum computer. *Proceedings of the Royal Society of London A*, 400(97), 1985.

[DJ92] David Deutsch and Richard Jozsa. Rapid solutions of problems by quantum computation. *Proceedings of the Royal Society of London A*, 439(553), 1992.

[EPR35] A. Einstein, B. Podolsky, and N. Rosen. Can quantum-mechanical description of physical reality be considered complete? *Phys. Rev.*, 47(10):777–780, May 1935.

[Fre04] Robert M. Freund. Introduction to semidefinite programming. Technical report, MIT, 2004.

[GC99] Daniel Gottesman and Isaac L. Chuang. Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations. *Nature*, 402:390–393, 1999.

[GHZ89] Daniel M. Greenberger, Michael A. Horne, and Anton Zeilinger. Going beyond bell's theorem. *in 'Bell's Theorem, Quantum Theory and Conceptions of the Universe', M. Kafatos (ed)*, pages 69–72, 1989.

[GLLL+11] I. Gerhardt, Q. Liu, A. Lamas-Linares, J. Skaar, V. Scarani, V. Makarov, and C. Kurtsiefer. Experimentally faking the violation of bell's inequalities. *Physical Review Letters*, 107(17):170404, Oct 2011.

[Gro96] Lov K. Grover. Acomputation mechanical algorithm for datebase search. *28th Annual ACM Symposium on the Theory of Computing*, page 212, May 1996.

[Gro01] Lov K. Grover. From schrodinger's equation to quantum search algorithm. *American J0urnal of Physics*, 69(7):769–777, 2001. Pedagogical view of the algorithm and its history.

[HDB05] M. Hein, W. Dur, and H.J. Briegel. Entanglement properties of multipartite entangled states under the influence of decoherence. *Phys. Rev. A*, 71:032350, 2005.

[HDE+05] M. Hein, W. Dur, J. Eisert, R. Raussendorf, M. Van den Nest, and H. J. Briegel. Entanglement in graph states and applications. In *Proceedings of the International School of Physics "Enrico Fermi" on "Quanum Computers, Algorithms and Chaos"*, 2005.

[HHHH09] Ryszard Horodecki, Pawel Horodecki, Michal Horodecki, and Karol Horodecki. Quantum entanglement. *Rev. Mod. Phys.*, (81):865–942, 2009.

[HWA+14] Matty J. Hoban, Joel J. Wallman, Hussain Anwar, Nairi Usher, Robert Raussendorf, and Dan E. Browne. Measurement-based classical computation. *Physical Review Letters*, 112(14):140505, April 2014.

[JAG+11] M. W. Johnson, M. H. S. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, P. Johansson, J.and Bunyk, E. M. Chapple, C. Enderud, J. P. Hilton, K. Karimi, E. Ladizinsky, N. Ladizinsky, T. Oh, I. Perminov, C. Rich, M. C. Thom, E. Tolkacheva, C. J. S. Truncik, S. Uchaikin, J. Wang, B. Wilson, and G. Rose. Quantum annealing with manufactired spins. *Nature*, 473(7346):194–198, 2011.

[Kok07] Pieter Kok. Review article: Linear optical quantum computing. *Rev. Mod. Phys.*, 79(135), 2007.

[Mac13] Lorenzo Maccone. A simple proof of bell's inequality. *American Journal of Physics*, 81:854, 2013.

[McK11] Matthew McKague. Self-testing graph states. In D. Bacon, M. Martin-Delgado, and M. Roettler, editors, *Theory of Quantum Computing, Communication and Cryptography*, pages 104–121. Springer Berlin, 2011.

[McK13] Matthew McKague. Interactive proofs for bqp via self-tested graph states. September 2013.

[MM10] Matthew McKague and Michele Mosca. Generalized self-testing and the security of the 6-state protocol. In W. van Dam, V. Kendon, and S. Severini, editors, *Theory of Quantum Computation, Communication and Cryptography, Lecture Notes in Computer Science*, volume 6519, pages 113–130. Springer Berlin, 2010.

[MMMO06] F. Magniez, D. Mayers, M. Mosca, and H. Ollivier. Self-testing of qunatum circuits. In M. Bugliesi, editor, *Proceedings of the 33rd International Colloquium on Automata, Languages and Programming*, pages 72–83, 2006.

[MP12] Mehdi Mhalla and Simon Perdrix. Graph states, pivot minor and universality of (x,z)-measurements. *arXiv:1202.6551 [quant-ph]*, 2012.

[MY04] Dominic Mayers and Mayers Yao. Self-testing quantum apparatus. *QIC*, 4(4):273–286, July 2004.

[NC10] Micheal A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2010.

[NPA08] Miguel Navascues, Stefano Pironio, and Antonio Acin. A convergent hierarchy of semidefinite programs characterizing the set of quantum correlations. *New Journal of Physics*, 10:073013, 2008.

[PAB+09] S. Pironio, A. Acin, N. Brunner, N. Gisin, S. Massar, and V. Scarani. Device-independent quantum key distribution secure against collecteve attacks. *New Journal of Physics*, 11:045021, April 2009.

[Pai79] A. Pais. Einstein and quantum theory. *Rev. Mod. Phys.*, 51(4):863–914, 1979.

[Pea70] P. M. Pearle. Hidden-variable example based upon data rejection. *Phys. Rev. D*, 2(8):1418–1425, Oct 1970.

[Pre13] John Preskill. Quantum computation : Physics 219 lecture notes, 2013.

[PVN14] Karoly F. Pal, Tamas Vertesi, and Miguel Navascues. Device independent tomography of multipartite quantum states. 2014.

[RBB03] R. Raussendorf, D. E. Browne, and Bri. Meacomputer-based quantum computation with cluster states. *Phys. Rev. A*, 68(2), 2003.

[RUV12] B. W. Reichardt, F. Unger, and U. Vazirani. A classical leash for a quantum system: Command of quantum systems via rigidity of chsh games. *arXiv:1209.0448 [quant-ph]*, 2012.

[RUV13] B. W. Reichardt, F. Unger, and U. Vazirani. Classical command of quantum systems. *Nature*, 496(7446):456–460, April 2013.

[SB09] Dan Shepherd and Michael J. Bremner. Temporally unstructured quantum computation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 465(2105):1413–1439, May 2009.

[Sch02] D. Schlingemann. Stabilizer codes can be realized as graph codes. *Quant. Inf. Comp.*, 2(4):307–323, 2002.

[Shi03] Yaoyun Shi. Both toffoli and controlled-not need little help to do universal quantum computing. *Quantum Information and Computation*, 3(1):84–92, January 2003.

[Sho94] Peter W. Shor. Algorithms for quantum computing: discrete logarithms and factoring. In *35th Annual Symposium on Foundations of Computer Science*, pages 124–134. IEEE, 1994.

[Sho97] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal of Computing*, 26:1484–1509, 1997.

[Sim94] Daniel R. Simon. On the power of quantum computation. *35th Annual Symposium on Foundations of Computer Science*, pages 116–123, 1994.

[SSSV14] Seung Woo Shin, Graeme Smith, John A. Smolin, and Umesh Vazirani. How "quantum" is the d-wave machine? *arXiv*, 2014.

[Sve87] George Svetlichny. Distinguishing three-body from two-body nonseparability by a bell-type inequality. *Phys. Rev. D*, 35(10):3066–3069, May 1987.

[SW02] D. Schlingemann and R. F. Werner. Quantum error-correcting codes associated with graphs. *Phys. Rev. A*, 65:012308, 2002.

[Tsi80] Boris Tsirelson. Quantum generalizations of bell's inequality. *Letters in Mathematical Physics*, 4(4):93–100, 1980.

[VB94] Lieven Vandenberghe and Stephen Boyd. Semidefinite programming. *SIAM Review*, 38:49–95, 1994.

[vDMMS00] W. van Dam, F. Magniez, M. Mosca, and M. Santha. Self-testing of universal and fault-tolerant sets of quantum gates. In *STOC'00: Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 688–696, 2000.

[WCY+14] X. Wu, Y. Cai, T. H. Yang, H. N. Le, JD. Bancal, and V. Scarani. Ronust self-testing of the 3-qubit w[state. July 2014.

[Wes00] Douglas B. West. *Introduction to Graph Theory.* :Prentice Hall, 2 edition, September 2000.

[WJD07] H. M. Wiseman, S. J. Jones, and A. C. Doherty. Steering, entanglement, nonlocality and the epr paradox. *Physical Review Letters*, 98:140402, 2007.

[Zai97] Christof Zaika. Grover's quantum seasearch algorithm is optimal. *Phys. Rev. A*, 60:2746–2751, 1997.

Appendix A

In this appendix we will give the matrix representation of POVMs obtained from the optimization process described in section 2.2. The POVMs will be denoted as $P^{a,b}_{MN}$, where MN represents the pair of observables whose projectors the POVM simulates and ab denotes the corresponding pair of eigenvalues.

$$P^{1,1}_{XX} = \begin{pmatrix} 0.125 & 0 & 0 & -0.125 \\ 0 & 0.375 & 0.125 & 0 \\ 0 & 0.125 & 0.375 & 0 \\ -0.125 & 0 & 0 & 0.125 \end{pmatrix} \qquad P^{1,-1}_{XX} = \begin{pmatrix} 0.375 & 0 & 0 & 0.125 \\ 0 & 0.125 & -0.125 & 0 \\ 0 & -0.125 & 0.125 & 0 \\ 0.125 & 0 & 0 & 0.375 \end{pmatrix}$$

$$P^{-1,1}_{XX} = \begin{pmatrix} 0.375 & 0 & 0 & 0.125 \\ 0 & 0.125 & -0.125 & 0 \\ 0 & -0.125 & 0.125 & 0 \\ 0.125 & 0 & 0 & 0.375 \end{pmatrix} \qquad P^{-1,-1}_{XX} = \begin{pmatrix} 0.125 & 0 & 0 & -0.125 \\ 0 & 0.375 & 0.125 & 0 \\ 0 & 0.125 & 0.375 & 0 \\ -0.125 & 0 & 0 & 0.125 \end{pmatrix} \tag{3.0.1}$$

$$P^{1,1}_{XZ} = \begin{pmatrix} 0.25 & 0.125 & 0.125 & 0 \\ 0.125 & 0.25 & 0 & 0.125 \\ 0.125 & 0 & 0.25 & 0.125 \\ 0 & 0.125 & 0.125 & 0.25 \end{pmatrix} \qquad P^{1,-1}_{XZ} = \begin{pmatrix} 0.25 & -0.125 & -0.125 & 0 \\ -0.125 & 0.25 & 0 & -0.125 \\ -0.125 & 0 & 0.25 & -0.125 \\ 0 & -0.125 & -0.125 & 0.25 \end{pmatrix}$$

$$P^{-1,1}_{XZ} = \begin{pmatrix} 0.25 & -0.125 & -0.125 & 0 \\ -0.125 & 0.25 & 0 & -0.125 \\ -0.125 & 0 & 0.25 & -0.125 \\ 0 & -0.125 & -0.125 & 0.25 \end{pmatrix} \qquad P^{-1,-1}_{XZ} = \begin{pmatrix} 0.25 & 0.125 & 0.125 & 0 \\ 0.125 & 0.25 & 0 & 0.125 \\ 0.125 & 0 & 0.25 & 0.125 \\ 0 & 0.125 & 0.125 & 0.25 \end{pmatrix} \tag{3.0.2}$$

$$P^{1,1}_{XD} = \begin{pmatrix} 0.162 & 0.088 & 0.088 & -0.088 \\ 0.088 & 0.338 & 0.088 & 0.088 \\ 0.088 & 0.088 & 0.338 & 0.088 \\ -0.088 & 0.088 & 0.088 & 0.162 \end{pmatrix} \qquad P^{1,-1}_{XD} = \begin{pmatrix} 0.338 & -0.088 & 0.088 & 0.088 \\ -0.088 & 0.162 & -0.088 & -0.088 \\ -0.088 & -0.088 & 0.162 & -0.088 \\ 0.088 & -0.088 & 0.088 & 0.338 \end{pmatrix}$$

$$P^{-1,1}_{XD} = \begin{pmatrix} 0.338 & -0.088 & 0.088 & 0.088 \\ -0.088 & 0.162 & -0.088 & -0.088 \\ -0.088 & -0.088 & 0.162 & -0.088 \\ 0.088 & -0.088 & 0.088 & 0.338 \end{pmatrix} \qquad P^{-1,-1}_{XD} = \begin{pmatrix} 0.162 & 0.088 & 0.088 & -0.088 \\ 0.088 & 0.338 & 0.088 & 0.088 \\ 0.088 & 0.088 & 0.338 & 0.088 \\ -0.088 & 0.088 & 0.088 & 0.162 \end{pmatrix} \tag{3.0.3}$$

$$P_{ZX}^{1,1} = \begin{pmatrix} 0.25 & 0.125 & 0.125 & 0 \\ 0.125 & 0.25 & 0 & 0.125 \\ 0.125 & 0 & 0.25 & 0.125 \\ 0 & 0.125 & 0.125 & 0.25 \end{pmatrix} \qquad P_{ZX}^{1,-1} = \begin{pmatrix} 0.25 & -0.125 & -0.125 & 0 \\ -0.125 & 0.25 & 0 & -0.125 \\ -0.125 & 0 & 0.25 & -0.125 \\ 0 & -0.125 & -0.125 & 0.25 \end{pmatrix}$$

$$P_{ZX}^{-1,1} = \begin{pmatrix} 0.25 & -0.125 & -0.125 & 0 \\ -0.125 & 0.25 & 0 & -0.125 \\ -0.125 & 0 & 0.25 & -0.125 \\ 0 & -0.125 & -0.125 & 0.25 \end{pmatrix} \qquad P_{ZX}^{-1,-1} = \begin{pmatrix} 0.25 & 0.125 & 0.125 & 0 \\ 0.125 & 0.25 & 0 & 0.125 \\ 0.125 & 0 & 0.25 & 0.125 \\ 0 & 0.125 & 0.125 & 0.25 \end{pmatrix} \tag{3.0.4}$$

$$P_{ZZ}^{1,1} = \begin{pmatrix} 0.375 & 0 & 0 & 0.125 \\ 0 & 0.125 & -0.125 & 0 \\ 0 & -0.125 & 0.125 & 0 \\ 0.125 & 0 & 0 & 0.375 \end{pmatrix} \qquad P_{ZZ}^{1,-1} = \begin{pmatrix} 0.125 & 0 & 0 & -0.125 \\ 0 & 0.375 & 0.125 & 0 \\ 0 & 0.125 & 0.375 & 0 \\ -0.125 & 0 & 0 & 0.125 \end{pmatrix}$$

$$P_{ZZ}^{-1,1} = \begin{pmatrix} 0.125 & 0 & 0 & -0.125 \\ 0 & 0.375 & 0.125 & 0 \\ 0 & 0.125 & 0.375 & 0 \\ -0.125 & 0 & 0 & 0.125 \end{pmatrix} \qquad P_{ZZ}^{-1,-1} = \begin{pmatrix} 0.375 & 0 & 0 & 0.125 \\ 0 & 0.125 & -0.125 & 0 \\ 0 & -0.125 & 0.125 & 0 \\ 0.125 & 0 & 0 & 0.375 \end{pmatrix} \tag{3.0.5}$$

$$P_{ZD}^{1,1} = \begin{pmatrix} 0.338 & 0.088 & 0.088 & 0.088 \\ 0.088 & 0.162 & -0.088 & 0.088 \\ 0.088 & -0.088 & 0.162 & 0.088 \\ 0.088 & 0.088 & 0.088 & 0.338 \end{pmatrix} \qquad P_{ZD}^{1,-1} = \begin{pmatrix} 0.162 & -0.088 & -0.088 & -0.088 \\ -0.088 & 0.338 & 0.088 & -0.088 \\ -0.088 & 0.088 & 0.338 & -0.088 \\ -0.088 & -0.088 & -0.088 & 0.162 \end{pmatrix}$$

$$P_{ZD}^{-1,1} = \begin{pmatrix} 0.162 & -0.088 & -0.088 & -0.088 \\ -0.088 & 0.338 & 0.088 & -0.088 \\ -0.088 & 0.088 & 0.338 & -0.088 \\ -0.088 & -0.088 & -0.088 & 0.162 \end{pmatrix} \qquad P_{ZD}^{-1,-1} = \begin{pmatrix} 0.338 & 0.088 & 0.088 & 0.088 \\ 0.088 & 0.162 & -0.088 & 0.088 \\ 0.088 & -0.088 & 0.162 & 0.088 \\ 0.088 & 0.088 & 0.088 & 0.338 \end{pmatrix} \tag{3.0.6}$$

$$P_{DX}^{1,1} = \begin{pmatrix} 0.162 & 0.088 & 0.088 & -0.088 \\ 0.088 & 0.338 & 0.088 & 0.088 \\ 0.088 & 0.088 & 0.338 & 0.088 \\ -0.088 & 0.088 & 0.088 & 0.162 \end{pmatrix} \qquad P_{DX}^{1,-1} = \begin{pmatrix} 0.338 & -0.088 & 0.088 & 0.088 \\ -0.088 & 0.162 & -0.088 & -0.088 \\ -0.088 & -0.088 & 0.162 & -0.088 \\ 0.088 & -0.088 & 0.088 & 0.338 \end{pmatrix}$$

$$P_{DX}^{-1,1} = \begin{pmatrix} 0.338 & -0.088 & 0.088 & 0.088 \\ -0.088 & 0.162 & -0.088 & -0.088 \\ -0.088 & -0.088 & 0.162 & -0.088 \\ 0.088 & -0.088 & 0.088 & 0.338 \end{pmatrix} \qquad P_{DX}^{-1,-1} = \begin{pmatrix} 0.162 & 0.088 & 0.088 & -0.088 \\ 0.088 & 0.338 & 0.088 & 0.088 \\ 0.088 & 0.088 & 0.338 & 0.088 \\ -0.088 & 0.088 & 0.088 & 0.162 \end{pmatrix} \tag{3.0.7}$$

$$P_{ZD}^{1,1} = \begin{pmatrix} 0.338 & 0.088 & 0.088 & 0.088 \\ 0.088 & 0.162 & -0.088 & 0.088 \\ 0.088 & -0.088 & 0.162 & 0.088 \\ 0.088 & 0.088 & 0.088 & 0.338 \end{pmatrix} \qquad P_{ZD}^{1,-1} = \begin{pmatrix} 0.162 & -0.088 & -0.088 & -0.088 \\ -0.088 & 0.338 & 0.088 & -0.088 \\ -0.088 & 0.088 & 0.338 & -0.088 \\ -0.088 & -0.088 & -0.088 & 0.162 \end{pmatrix}$$

$$P_{ZD}^{-1,1} = \begin{pmatrix} 0.162 & -0.088 & -0.088 & -0.088 \\ -0.088 & 0.338 & 0.088 & -0.088 \\ -0.088 & 0.088 & 0.338 & -0.088 \\ -0.088 & -0.088 & -0.088 & 0.162 \end{pmatrix} \qquad P_{ZD}^{-1,-1} = \begin{pmatrix} 0.338 & 0.088 & 0.088 & 0.088 \\ 0.088 & 0.162 & -0.088 & 0.088 \\ 0.088 & -0.088 & 0.162 & 0.088 \\ 0.088 & 0.088 & 0.088 & 0.338 \end{pmatrix} \tag{3.0.8}$$

$$P_{DD}^{1,1} = \begin{pmatrix} 0.25 & 0.125 & 0.125 & 0 \\ 0.125 & 0.25 & 0 & 0.125 \\ 0.125 & 0 & 0.25 & 0.125 \\ 0 & 0.125 & 0.125 & 0.25 \end{pmatrix} \qquad P_{DD}^{1,-1} = \begin{pmatrix} 0.25 & -0.125 & -0.125 & 0 \\ -0.125 & 0.25 & 0 & -0.125 \\ -0.125 & 0 & 0.25 & -0.125 \\ 0 & -0.125 & -0.125 & 0.25 \end{pmatrix}$$

$$P_{DD}^{-1,1} = \begin{pmatrix} 0.25 & -0.125 & -0.125 & 0 \\ -0.125 & 0.25 & 0 & -0.125 \\ -0.125 & 0 & 0.25 & -0.125 \\ 0 & -0.125 & -0.125 & 0.25 \end{pmatrix} \qquad P_{DD}^{-1,-1} = \begin{pmatrix} 0.25 & 0.125 & 0.125 & 0 \\ 0.125 & 0.25 & 0 & 0.125 \\ 0.125 & 0 & 0.25 & 0.125 \\ 0 & 0.125 & 0.125 & 0.25 \end{pmatrix} \tag{3.0.9}$$

Appendix B

In this appendix we give the Matlab code used for performing the nonlocality tests described in section 2.1.

```
n = 6
DeterministicPoints = zeros(2^(2*n),n^2);

for i = 1:2^n
    for j = 1:2^n
        for k = 1:n
            for l = 1:n
                A = matrixcombination(n);
                if A(i,k) == A(j,l)
                    DeterministicPoints((i-1)*2^n+j,(k-1)*n + l) = 1;
                else
                DeterministicPoints((i-1)*2^n+j,(k-1)*n + l) = 0;
                end
            end
        end
    end
end
DeterministicPoints = DeterministicPoints'

%gives the matrix of deterministic points for 6
%measurements per each side

%PX gives the probabilities for obtaining results o(Ai) = o(Aj) accoring
to
%already known expected results
%EV are the expectation values

EV = [1 -1 1 0 -1 1/sqrt(2) 1/sqrt(2)]';
PX = (EV+1)/2;
P0 = (PX - 1/2);
f = zeros(1,2^(2*n) + 1);
f(1,1) = -1;

%P0 gives the difference of these probabilities and random outcomes coming
%from term lambda*PX + (1-lambda)*I/2
```

```
DPoints2(1,:) = DeterministicPoints(1,:);
DPoints2(2,:) = DeterministicPoints(8,:);
DPoints2(3,:) = DeterministicPoints(16,:);
DPoints2(4,:) = DeterministicPoints(19,:);
DPoints2(5,:) = DeterministicPoints(21,:);
DPoints2(6,:) = DeterministicPoints(25,:);
DPoints2(7,:) = DeterministicPoints(34,:);

%Takes out probabilities that are not probabilities of reference
%measurements

Aeq = zeros(7,2^(2*n)+1);
Aeq(:,1) = P0;
Aeq(:,2:2^(2*n)+1) = -1*DPoints2;
Aeq(8,2:2^(2*n) + 1) = ones(1,2^(2*n));

%will solve the problem lamdba*(PX - I/2) - mu*(det.points.) = -I/2 and
%convexicity of mu

beq = ones(7,1);
beq = -1/2*beq;
beq(8,1) = 1;

%associated with -I/2

A = zeros(2^(2*n) + 1);
for i = 2:length(A)
    A(i,i) = -1;
end

b = zeros(length(A),1);
minimum = linprog(f,A,b,Aeq,beq);
lambda = minimum(1,1)
```

Appendix C

In this appendix we provide the Matlab code used for performing numerical work described in section 2.2

sdpST3.m:

```
dB = 2;
dC = 2;
[V,D] = eig((1/sqrt(2))*[1 1; 1 -1]);
V1 = [V(1,1); V(2,1)];  %eigenvector corresponding to eigenvalue -1 of our
diagonal operator (X+Z)/sqrt(2)
V2 = [V(2,1); V(2,2)];  %eigenvector corresponding to eigenvalue 1 of our
diagonal operator (X+Z)/sqrt(2)

Max(:,:,1) = (1/2)*[1 1; 1 1]; %projector on x=0
Max(:,:,2) = (1/2)*[1 -1; -1 1]; %projector on x=1
Max(:,:,3) = [1 0; 0 0]; %projector on z=0
Max(:,:,4) = [0 0; 0 1]; %projector on z=1
Max(:,:,5) = V2*V2' %projector on d=0
Max(:,:,6) = V1*V1'; %projector on d=1

Mby = Max;
Mcz = Max;

psitriangle = 1/2*((1/sqrt(2))*kron([1;0;0;0],[1;1])
+(1/sqrt(2))*kron([0;1;0;0],[1;-1])+(1/sqrt(2))*kron([0;0;1;0],[1;-1]) -
(1/sqrt(2))*kron([0;0;0;1],[1;1]));
rhotriangle = psitriangle*psitriangle';
psiGHZ = (1/(2*sqrt(2)))*(kron([1;0;0;0],[1;1]) +kron([0;1;0;0],[1;1]) +
kron([0;0;1;0],[1;-1]) - kron([0;0;0;1],[1;-1]));
rhoGHZ = psiGHZ*psiGHZ';

cvx_begin SDP
cvx_solver sedumi

    variable p
    variable Nbcyz(dB*dC,dB*dC,36) complex

    expression sumNbc(dB*dC,dB*dC,9)
```

```
    maximise( p )

    for a = 0:1
        for b = 0:1
            for c = 0:1
                for x = 0:2
                    for y = 0:2
                        for z = 0:2

trace(Tensor(Max(:,:,1+a+2*x),Mby(:,:,1+b+2*y),Mcz(:,:,1+c+2*z))*rhotriang
le) == ...

trace(Tensor(Max(:,:,1+a+2*x),Nbcyz(:,:,1+c+2*b+4*z+12*y))*rhoGHZ);
                        end
                    end
                end
            end
        end
    end


    for y = 0:2
        for z = 0:2
            sumNbc(:,:,1+y+3*z) = zeros(dB*dC)
            for b = 0:1
                for c = 0:1
                    sumNbc(:,:,1+y+3*z) = sumNbc(:,:,1+y+3*z) +
Nbcyz(:,:,1+c+2*b+4*z+12*y)
                    Nbcyz(:,:,1+c+2*b+4*z+12*y) - p*eye(dB*dC) ==
hermitian_semidefinite(dB*dC)
                    PartialTranspose(Nbcyz(:,:,1+c+2*b+4*z+12*y)) ==
hermitian_semidefinite(dB*dC)
                end
            end
            sumNbc(:,:,1+y+3*z) == eye(dB*dC)
        end
    end

 cvx_end

 A = zeros(36,4);
 for i = 1:36
      [q,w] = eig(Nbcyz(:,:,i));
      q = round(1000*q)
     for j = 1:4
                 A(i,j) = IsProductVector(q(:,j));
     end
 end
 A
```

sdpST4.m:

```
%Checking the possibility of the simulation for the 4qubit graph state
(two
%triangles with and without entanglement on both sides)


dB = 2;
dC = 2;
[V,D] = eig((1/sqrt(2))*[1 1; 1 -1]);
V1 = [V(1,1); V(2,1)]; %eigenvector corresponding to eigenvalue -1 of our
diagonal operator (X+Z)/sqrt(2)
V2 = [V(2,1); V(2,2)]; %eigenvector corresponding to eigenvalue 1 of our
diagonal operator (X+Z)/sqrt(2)

Max(:,:,1) = (1/2)*[1 1; 1 1]; %projector on x=0
Max(:,:,2) = (1/2)*[1 -1; -1 1]; %projector on x=1
Max(:,:,3) = [1 0; 0 0]; %projector on z=0
Max(:,:,4) = [0 0; 0 1]; %projector on z=1
Max(:,:,5) = V2*V2'; %projector on d=0
Max(:,:,6) = V1*V1'; %projector on d=1

Mby = Max;
Mcz = Max;
Mdf = Max;

Max(:,:,7) = eye(2);
Mcz(:,:,7) = eye(2);

plus = 1/sqrt(2)*[1;1];
psi = Tensor(plus,plus,plus,plus); %initial state before constructing the
graph state
CZ = [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 0 -1];
CZ13 = [1 0 0 0 0 0 0 0;0 1 0 0 0 0 0 0;0 0 1 0 0 0 0 0;0 0 0 1 0 0 0 0;0
0 0 0 1 0 0 0;0 0 0 0 0 -1 0 0;0 0 0 0 0 0 1 0;0 0 0 0 0 0 0 -1];
CZ14 = eye(16);
CZ14(10,10) = -1;
CZ14(12,12) = -1;
CZ14(14,14) = -1;
CZ14(16,16) = -1;
Id = [1 0; 0 1];
psi12E34E =
CZ14*(Tensor(Id,CZ13)*(Tensor(CZ13,Id)*(Tensor(Id,Id,CZ)*(Tensor(Id,CZ,Id)
*(Tensor(CZ,Id,Id)*psi))))); %state with entanglement on both sides
rho12E34E = psi12E34E*psi12E34E';
psi12E =
CZ14*(Tensor(Id,CZ13)*(Tensor(CZ13,Id)*(Tensor(Id,CZ,Id)*(Tensor(CZ,Id,Id)
*psi)))); %state with entanglement just on trusted side, supposed to
simulate psi12E34E
rho12E = psi12E*psi12E';
psi34E =
CZ14*(Tensor(Id,CZ13)*(Tensor(CZ13,Id)*(Tensor(Id,Id,CZ)*(Tensor(Id,CZ,Id)
*psi)))); %state with entanglement just on the prover side
rho34E = psi34E*psi34E';
```

```
psiNE = CZ14*(Tensor(Id,CZ13)*(Tensor(CZ13,Id)*(Tensor(Id,CZ,Id)*psi)));
%state with no entanglement, supposed to simulate psi34E
rhoNE = psiNE*psiNE';

cvx_begin SDP
cvx_solver sedumi

    variable p
    variable Nbcyz4(dB*dC,dB*dC,42) complex

    expression sumNbc(dB*dC,dB*dC,12)

    maximise( p )

    for a = 0:2
        for d = 0:1
          for b = 0:1
            for c = 0:1
                for x = 0:2
                    for f = 0:2
                    for y = 0:2
                        for z = 0:2
                            trace(Tensor(Max(:,:,1+a+2*x),Mdf(:,:,1+d +
2*f),Mby(:,:,1+b+2*y),Mcz(:,:,1+c+2*z))*rho12E34E) ==
trace(Tensor(Max(:,:,1+a+2*x),Mdf(:,:,1+d +
2*f),Nbcyz4(:,:,1+c+2*b+4*z+12*y))*rho12E);
                             trace(Tensor(Max(:,:,1+a+2*x),Mdf(:,:,1+d +
2*f),Mby(:,:,1+b+2*y),Mcz(:,:,1+c+2*z))*rho34E) ==
trace(Tensor(Max(:,:,1+a+2*x),Mdf(:,:,1+d +
2*f),Nbcyz4(:,:,1+c+2*b+4*z+12*y))*rhoNE);
                        end
                    end
                end
            end
          end
        end
        end
    end


    for i = 1:6
        for a = 0:2
            for d = 0:1
                for x = 0:2
                    for f = 0:2
       trace(Tensor(Max(:,:,1+a+2*x),Mdf(:,:,1+d +
2*f),Mby(:,:,i),Mcz(:,:,7))*rho12E34E) ==
trace(Tensor(Max(:,:,1+a+2*x),Mdf(:,:,1+d + 2*f),Nbcyz4(:,:,36 +
i))*rho12E);
       trace(Tensor(Max(:,:,1+a+2*x),Mdf(:,:,1+d +
2*f),Mby(:,:,i),Mcz(:,:,7))*rho34E) ==
trace(Tensor(Max(:,:,1+a+2*x),Mdf(:,:,1+d + 2*f),Nbcyz4(:,:,36 +
i))*rhoNE);
                    end
```

```
                end
            end
        end
    end



    for y = 0:2
        for z = 0:2
            sumNbc(:,:,1+y+3*z) = zeros(dB*dC)
            for b = 0:1
                for c = 0:1
                    sumNbc(:,:,1+y+3*z) = sumNbc(:,:,1+y+3*z) +
Nbcyz4(:,:,1+c+2*b+4*z+12*y)
                    Nbcyz4(:,:,1+c+2*b+4*z+12*y) - p*eye(dB*dC) ==
hermitian_semidefinite(dB*dC)
                    PartialTranspose(Nbcyz4(:,:,1+c+2*b+4*z+12*y)) ==
hermitian_semidefinite(dB*dC)
                end
            end
            sumNbc(:,:,1+y+3*z) == eye(dB*dC)
        end
    end

    sumNbc(:,:,10) = Nbcyz4(:,:,37) + Nbcyz4(:,:,38)
    sumNbc(:,:,11) = Nbcyz4(:,:,39) + Nbcyz4(:,:,40)
    sumNbc(:,:,12) = Nbcyz4(:,:,41) + Nbcyz4(:,:,42)

    for u = 1:3
        sumNbc(:,:,9+u) == eye(dB*dC)
    end


 cvx_end
```