Towards a fault-tolerant one-way quantum repeater

Maria Flors Mor Ruiz

TU Delft

Towards a fault-tolerant one-way quantum repeater

by

Maria Flors Mor Ruiz

to obtain the degree of Master of Science
at the Delft University of Technology
to be defended publicly on
30th April 2021 at 13:30

Student number:	5153506	
Project duration:	September 2, 2020 - April 9, 2021	
Thesis committee:	Prof. Dr. S. D. C. Wehner	TU Delft, supervisor
	Prof. J. Borregaard	TU Delft
	Prof. S. Gröblacher	TU Delft
Daily supervisor:	PhD candidate G. Avis	TU Delft

Abstract

Quantum communication can enable new features that are provably impossible with classical communication alone. However, the optical fibers used to send the quantum information are inherently lossy. To overcome the exponential losses over distance so-called quantum repeaters are needed to amplify the signal. As opposed to memory-based approaches, the third generation of quantum repeaters, also called one-way quantum repeaters do not require two-way communication thus enabling very high communication rates. In particular, the one-way quantum repeater based on photonic tree states proposed by Borregaard et al. (2019) realizes this task with a very modest amount of resources. Nevertheless, the method considered is susceptible to operational errors. In this work, we propose the use of code concatenation of a stabilizer code and the tree code, so that by measuring stabilizers of the stabilizer code and applying the syndrome corrections we can achieve a fault-tolerant one-way quantum repeater. In order to do so, we present a detailed protocol that uses the 5-qubit code. Moreover, we develop the first fully general simulation framework for studying the performance of tree-code based one-way quantum-repeater chains, which in this thesis is used to perform an analysis of the proposed protocol. We find that the code-concatenation protocol under consideration has a similar tolerance against operational errors to the protocol proposed by Borregaard et al. (2019). Unfortunately, we are not able to draw a distinction between the tolerance of the two approaches. We do however suggest modified protocols that may provide fault tolerance. Studying these is beyond the scope of this thesis.

Acknowledgements

There are many people who have been instrumental to this thesis. During this intense period, I have been surrounded by wonderful people that have made it possible.

First and foremost, I would like to thank my supervisor, Prof. Stephanie Wehner, and my daily supervisor, PhD candidate Guus Avis, for providing me an opportunity to work alongside them and all the brilliant individuals in their group. All the discussions we had, and their constant support and excellent feedback have shaped and helped achieve the goals of this thesis. Guus, thank you for all the countless hours that you have invested in sharing your expertise and helping me whenever I needed it. Moreover, thanks for leading my first steps into the field of quantum communication with such passion and patience. I would also like to thank everyone else in the group, who warmly welcomed me and made my work much easier and enjoyable. In particular, thanks to the people in the Blueprint Team for always having an open ear for my questions and for being my main connection to group through the countless meetings. Next, I would like to thank Helena for taking care of all the administrative aspects of the project in the most straightforward and comfortable way possible. Further, I would like to express my deepest appreciation to Prof. Johannes Borregaard for his insightful help throughout the project and his collaboration along with the graduate student Nishad Maskara on the future that this project may present. Finally, special thanks to Prof. Borregaard and Prof. Gröblacher for kindly accepting the invitation to join the thesis committee and taking the time to read this report.

I cannot begin to express my thanks to all my friends, who have taken care of me and have distracted me from work when I need it the most. Specially Elsie, thanks for letting me into your life. Albert, thanks for unconditional help. To my housemates and friends, thanks for all the lunches, dinners, walks, movie nights and countless games that have brought me the joy that the sun in the Netherlands cannot bring me. Guillem, thanks for always bringing a smile to my face and for believing in me and giving me the strength to carry on. And thank you Mama, Papa, for your endless love and support, thanks for always being there and trusting me no matter how things may go.

Thank you all from the bottom of my heart.

Contents

Abstract iii

Acknowledgements v

1 Introduction 1

2 Background 3
- 2.1 Quantum communication and quantum repeaters . . . 3
  - 2.1.1 Overview on quantum repeaters . . . 4
  - 2.1.2 One-way quantum repeater . . . 5
- 2.2 Quantum noise and secret-key rate . . . 6
  - 2.2.1 Independent noise models . . . 7
  - 2.2.2 Determine the impact of noisy quantum channels . . . 8
- 2.3 Quantum Error Correction . . . 10
  - 2.3.1 Classical linear codes . . . 10
  - 2.3.2 Stabilizer codes . . . 11
  - 2.3.3 Code concatenation . . . 17
- 2.4 Graph states, tree state and tree code . . . 18
  - 2.4.1 Graph states . . . 19
  - 2.4.2 Tree states . . . 23
  - 2.4.3 Tree code . . . 24
- 2.5 One-way quantum repeater protocol with tree code . . . 31
  - 2.5.1 Encoding . . . 32
  - 2.5.2 Re-encoding . . . 32
  - 2.5.3 Repetition and end node . . . 34
  - 2.5.4 Errors . . . 34

3 Code concatenation in the one-way quantum repeater 37
- 3.1 Choice of concatenated codes . . . 38
- 3.2 Preliminary protocol for the code concatenated one-way quantum repeater 38
- 3.3 Stabilizer measurements in trees . . . 39
  - 3.3.1 Z...Z stabilizer measurement . . . 40
  - 3.3.2 X...X stabilizer measurement . . . 41
  - 3.3.3 General X and Z stabilizer measurement . . . 42
- 3.4 Possible outer codes . . . 44
- 3.5 Protocol 1 for the code concatenated one-way quantum repeater . . . 46
  - 3.5.1 Protocol 1: Structure of the repeater chain . . . 46
  - 3.5.2 Protocol 1: Losses . . . 47
  - 3.5.3 Protocol 1: Errors . . . 48

3.6 Protocol 2 for the code concatenated one-way quantum repeater 50
3.6.1 Protocol 2: Losses 51

4 Numerical investigation of the 5-qubit code 57
4.1 Motivation and constraints for the simulation 58
4.2 The NetSquid Simulator 59
4.3 Choice of branching vector and loss probability 60
4.4 Verification of the simulation 63
4.5 Protocol 2 with four links 64
4.6 Improving the strategy 67
4.6.1 Error build-up for the free evolution links 67
4.6.2 Perfect stabilizer measurements 69
4.6.3 Behaviour of a long chain 69
4.6.4 Imperfect stabilizer measurements 71

5 Conclusions and outlook 75
5.1 Summary and conclusions 75
5.2 Future outlook 76

References 79

CHAPTER 1

Introduction

Over the last several years, the word *quantum* has been used increasingly due to advances in the field of Quantum Information Technology, such as Google's quantum advantage experiment [1]. Moreover, the European Union has announced that "*the future is Quantum*" along with the investment of one billion euro embodied in the Quantum Flagship programme [2] as part of Horizon 2020 [3]. One of the many projects that form the Quantum Flagship is the Quantum Internet Alliance (QIA) [4], whose goal is *"to develop a Blueprint for a pan-European entanglement-based Quantum Internet, by developing, integrating and demonstrating all the functional hardware and software subsystems"* [5]. The proposed Quantum Internet [6] would make use of properties inherent to quantum mechanics such as entanglement and superposition in order to bring extra features to the already known and used classical Internet, which range from Quantum Key Distribution [7] as a provably secure communication, clock synchronization [8] or blind quantum computing [9] to a global 4-dimensional quantum positioning system [10].

Common to all the aforementioned applications is distributing an entangled state or sending a single-qubit state over long distances. Long distance communication is also a common feature of the classical Internet, which needs signal amplification at some midpoints. However, in the quantum realm amplification is not allowed due to the fact that quantum information can not be copied [11]. Nevertheless, quantum repeaters [12] were introduced as a more sophisticated and adequate method. The study of quantum repeaters has evolved considerably since its first appearance in 1998. Nowadays, they can be classified in three generations differentiated by the tools that they use to share entanglement or send information and overcome possible losses and operational errors. In particular, the third generation [13, 14], uses quantum error correcting codes as the main mechanism to protect the information being sent.

Under the guidance of Prof. Wehner, a team of theoretical physicists, mathematicians, and software engineers, the NLBlueprint Team, is working on simulating a Quantum Internet for the Netherlands. The simulation allows us to investigate the requirements for a reliable quantum communication and thus enable us to define further design and research goals with the final goal to develop a realistic blueprint for a large-scale Quantum Internet. This thesis has been realized in the framework of the NLBlueprint Team contributing to the simulations of the all-photonic one-way quantum repeaters based on

the tree code, a promising candidate of the third generation of quantum repeaters. In particular, we highlight the approach taken by Borregaard et al. in [15] which requires a very modest amount of resources in order to successfully perform quantum communication. Despite the efficiency in terms of losses, the method considered does not perform fault tolerantly against operational errors.

In this thesis we direct our efforts towards a fault-tolerant one-way quantum repeater. The main contributions of this thesis are:

> We propose a protocol that uses code concatenation in order to achieve fault-tolerance in the one-way quantum repeater with minimal resources. The code concatenation is composed of the tree code and a stabilizer code, which is used to detect and correct the errors that the previous encoding [15] is unable to.

> We develop the first fully general simulation framework for studying the performance of tree-code based one-way quantum-repeater chains.

> We compare the behaviour of this protocol to the previous non-fault-tolerant protocol in [15] in order to assess a potential improvement of the error tolerance using the framework of the developed simulation.

OUTLINE AND CONTRIBUTIONS. The structure of this thesis is the following:

- In Chapter 2 we introduce the concepts and definitions that are used throughout this thesis. We start by a short introduction on quantum repeaters, channels, and codes in order to build the intuition for the subsequent chapters. Special attention is given to the tree code and an application of it in the one-way quantum repeater as they are the main ingredients of this thesis.
- In Chapter 3 we present the core of our contributions. We start with a discussion on the choice of the codes to be used in a code concatenated one-way quantum repeater. Next, we present a preliminary protocol and a general method to measure the stabilizers in the proposed code concatenation. This leads to an examination of the required characteristics for a specific outer code. Finally, two protocols for the code concatenated one-way quantum repeater using the 5-qubit code as outer code are defined, where the first one is used as a stepping stone for the second one.
- Chapter 4 contains the results of the analysis of the behaviour of the protocol presented in the previous chapter and its comparison to the one-way quantum repeater with the tree code, introduced in Section 2.5. First, we motivate the simulation of the code concatenated one-way quantum repeater. Next, the tools used to develop it are introduced. Later, a detailed explanation on the choice of the branching vector and loss probability for the simulation is presented, together with a verification of the simulation. Finally, the results of a first approach are presented and lead us to propose a better strategy, from which we give an analytical study and some preliminary results.

CHAPTER 2

Background

The goal of this chapter is to introduce a solid theoretical background information needed to understand the work presented in this thesis. As a first step in 2.1, quantum communication and its main problems are introduced in order to motivate quantum repeaters, which are explained with a focus on the one-way quantum repeater. Next, in 2.2 an explanation to understand how the noise of quantum channels works and how to quantify it is presented. Following, in 2.3 an overview of Quantum Error Correction is given targeting a more detailed explanation of stabilizer codes. Then in 2.4, graph states, the tree state and tree code, which belongs to the family of stabilizer codes, are carefully detailed. Finally, in 2.5 an application of the one-way quantum repeater using the tree code is introduced.

2.1 Quantum communication and quantum repeaters

Quantum communication [16] relies on the principles of quantum mechanics to transmit quantum signals over distances. This communication can be used as means for large-scale distributed quantum computers [17] or a secure communication using cryptography [18], by means of the widely known Quantum Key Distribution (QKD) protocol, amongst many others. One realization of this field is the Quantum Internet [6, 19], which has the goal to enable quantum communication between any two points on Earth, thus requiring transmission and distribution of quantum signals over long distances. In order to do so, the signals need to be sent through a quantum channel such as an optical fiber or free space [20]. Currently photons are used as means to carry quantum information or as the so-called "flying qubits" [20]. Moreover, the sent photons through fiber or free space are subject to losses, which increase with the channel distance.

Unfortunately, the average distances involved in telecommunication networks are typically of the order of hundreds or thousands of kilometers. This implies a severe constraint when it comes to channel losses, thus, limiting the transmission of such signals. In classical communication this issue is also present, and it is overcome by signal amplification at

intermediate points of the channel. However, in the quantum world this is not possible due to the no-cloning theorem [11], which deprives a quantum signal of such operation. Therefore, quantum repeaters were proposed as an alternative approach. Those were first described in 1998 by Briegel et al. [12] as intermediate connection points, which are a tool used in each node of a segmented long link. A few years later, the first quantum repeater architecture was proposed by Duan-Lukin-Cirac-Zoller (DLCZ) using atomic ensembles and linear optics [21].

2.1.1 Overview on quantum repeaters

Before going any further, it is important to mention that quantum states have a property known as *entanglement*, which is strictly inherent to the quantum world and its quantum non-locality [22]. It is at the very core of quantum communication protocols to send or share quantum information. The latter is done by generating an entangled quantum state and distributing it. This can be achieved in a variety of approaches depending on the physical implementation of choice. In particular, the probability of success for generating a single entangled link through a fiber using photons scales exponentially with the distance of the link L, such that e^{-L/L_0} [23], where L_0 is the attenuation length, which is a parameter that characterizes the loss of the fiber. Overall, the idea of using quantum repeaters is to break one long segment or link into several smaller ones, called *elementary links* in order to increase the success probability of entanglement distribution.

A memory-based quantum repeater extends this entanglement between all the elementary links so that entanglement between the initial and end node is achieved. Therefore, a quantum repeater relies on:

- **Entanglement distribution:** a process that allows the generation of entanglement over elementary links. This is typically performed by means of heralded entanglement, which requires a two-way classical message between the sender and the receiver to be able to know if the entanglement was successful or not.
- **Entanglement purification:** a process that create a more highly entangled state from lower quality entangled states. The original scheme was presented by Bennett et al. [24] and many more have been proposed ever since.
- **Entanglement swapping:** a process where by means of a Bell state measurement a longer entangled link between adjacent repeater nodes is achieved [25, 26].

So far only conventional quantum repeaters have been considered, this is because there are different families of quantum repeaters that depend on the requirements or tools that they use to deal with losses, errors and waiting times. The conventionally presented ones in introductory literature correspond to the first generation of quantum repeaters. There is a second generation of quantum repeaters, where instead of using entanglement purification, quantum error correcting codes are used to perform the same task. Finally, there is a last third generation that uses quantum error correcting codes to both create and distribute entanglement and to purify it, which avoid communication delays. The

focus of this thesis is on this last generation, which is explained in more detail in the following subsection.

In this thesis the other generations will not be explored in further depth, for the interested reader the following sources are given and are the ones mainly used [23] and [27].

2.1.2 One-way quantum repeater

The first and second generation quantum repeaters use heralded entanglement to perform entanglement distribution. This procedure requires a classical message to herald the successful entanglement distribution. Before this message is received at the end node, the qubits in the nodes where the distribution is being performed are not available or in other words, they need to "wait". In order to avoid the need of sending classical messages when sending a quantum signal between repeater nodes, the signal needs to be encoded in a loss tolerant manner [23].

Therefore, the goal of the third generation quantum repeaters is to use loss tolerant codes in order to avoid waiting times. In particular, photonic loss tolerant codes are widely used, these aim to initially send a matter qubit with encoded information, a message qubit, by encoding it in a multi-photonic loss-tolerant code and send those photons to the next repeater node. This first node is called the sending node or station. Then, at the next repeater node the information is transferred back to matter qubits so that one can know if the state was lost on the transmission between nodes. Finally, these retrieved matter qubits act as the new message qubits, which are then again encoded into photons and sent to the next node. This procedure of receiving photons and retrieving the encoded information in matter qubits and then use those to encode again the information and send it is called the re-encoding procedure. This is repeated at each intermediate node of the repeater chain until the end node is reached, where the information is finally retrieved. This approach also allows generating entanglement over long distances by sending an entangled qubit along the chain of repeaters.

Thus, the intermediate nodes are used for "refreshing" the loss-tolerant code so that the matter qubits in the nodes do not require long-lived quantum memories as in the heralded entanglement approach. In this generation the system sends information along the links in one direction, from sender to receiver, making the transmission rate of the chain to be determined by the slowest component.

The use of quantum error correcting codes requires that the memories are able to reliably store the state during the process of error correction, making it completely independent of the communication time between nodes. The use of error correcting codes for both operational errors and losses requires more advanced operations than in the previous generations in turn requiring higher gate fidelities.

In short, the third generation quantum repeaters do not need long-lived quantum memories nor two-way communication between sender and receiver. These quantum repeaters are also called one-way quantum repeater since they send only information in one direction without the need of pre-established entangled links. In order to protect the quantum infor-

mation from losses and errors they use photonic encodings and quantum error-correcting codes. Consequently, the distribution rate in one-way quantum repeaters is significantly boosted [23]. Finally, note that the one-way loss tolerant codes can only tolerate losses up to 50% [28]. Intuitively one can see that re-encoding when there is a 50% of losses or more is impossible due to the fact that the no-cloning theorem [11] would be violated [29]. This restriction leads to shorter distances between nodes, which for a conventional telecom fiber, which has $L_0 \approx 22$ km, the inter-node distance is approximately 15 km.

Many approaches to the one-way all-photonic quantum repeaters have been proposed as in [13], [15], [30], and [31] amongst many more. A general challenge for this family of quantum repeaters is how to physically implement an effective generation of multi-qubit error-correcting codes and perform error correction.

2.2 Quantum noise and secret-key rate

Noise in a quantum system can be understood as the interaction of the quantum system with some other environment system. This quantum system and its interaction form an open quantum system, in which one wishes to neglect or average over the dynamics of the environment. To model this interaction the formalism of quantum operations is used as it describes generic state changes without explicit reference to the passage of time. Consider some initial state of the quantum system ρ, and a final state ρ' (in the density matrix representation) are related by a quantum operation $\mathcal{E}$, such that $\rho' = \mathcal{E}(\rho)$. This operation captures the dynamic change to a state which occurs as the result of some physical process. The behaviour of an open quantum system can be modeled making use of the operator-sum representation [32]

$$\mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger, \tag{2.1}$$

where E_k are the operation elements for $\mathcal{E}$ known as Kraus operators, satisfying $\sum_k E_k^\dagger E_k = I$ if the quantum operation is trace-preserving, which is the case that will be considered throughout the text.

From a quantum communication point of view, one must assume that the exchange of quantum information through a quantum channel is subject to noise, just like in classical communication. In the classical case, the noise can only flip or erase a bit. Whereas the noise introduced in a quantum channel can be a continuum of errors on a single qubit.

Definition 2.2.1. A **quantum channel** is a linear, completely positive, and trace-preserving (CPTP) map, that maps density matrices to density matrices preserving both their trace ($\mathrm{Tr}(\rho) = 1$) and positiveness ($\rho \geq 0$). So that in the operation-sum representation a noisy quantum channel can be represented by equation 2.1 and $\sum_k E_k^\dagger E_k = I$.

On a system of qubits, the noise of a quantum channel can be modeled and categorized into two main groups: independent or correlated noise models. On the one hand, independent noise happens for each of the qubits in a system independently in an uncorrelated fashion. Such models are useful to describe the evolution of non-interacting qubits. On

the other hand, a correlated system of qubits is more suitably described by an appropriate correlated noise model given by the underlying interactions.

On this text the main focus will be on independent noise models on qubit systems. These are a stepping stone that allows a simple single qubit analysis. At the same time independent noise models fall into two categories, erasure/loss and arbitrary errors.

2.2.1 Independent noise models

Arbitrary errors correspond to probabilistic rotations of the qubit in the Bloch sphere. As stated previously, the arbitrary errors are continuous. At the same time, the Pauli matrices form an orthogonal basis for a two-dimensional Hilbert space [32] and therefore the operators E_k of a quantum channel can be written as,

$$E_k = e_k^I I + e_k^X X + e_k^Y Y + e_k^Z Z, \tag{2.2}$$

where e_k^i belonging to complexes are constants and the identity, I, and Pauli matrices are defined as

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \text{ and } Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.3}$$

Throughout the text the eigenstates of X and Z will be denoted as $|0\rangle$, $|1\rangle$, $|+\rangle$ and $|-\rangle$. Moreover, this shows that quantum errors despite being continuous can be corrected using a discrete set of operations. Note that $Y \propto XZ$, so that one only need to apply X and Z gates to correct an arbitrary error.

In what follows, some examples of arbitrary errors and their quantum operations that are going to be used throughout the text are presented.

Definition 2.2.2. A single qubit **bit flip channel** is a quantum channel that flips the state of a qubit from $|0\rangle$ to $|1\rangle$ and vice versa with probability ϵ. Its operation elements are

$$E_0 = \sqrt{1-\epsilon} I, \quad E_1 = \sqrt{\epsilon} X. \tag{2.4}$$

Definition 2.2.3. A single qubit **phase flip channel** is a quantum channel that adds a phase of π to the state $|1\rangle$ with probability ϵ. Its operation elements are

$$E_0 = \sqrt{1-\epsilon} I, \quad E_1 = \sqrt{\epsilon} Z. \tag{2.5}$$

Definition 2.2.4. A single qubit **depolarizing channel** is a quantum channel that depolarizes a qubit with probability ϵ. This means that with probability $\epsilon/4$ a qubit will have a X (bit-flip), Z (phase-flip) or Y (bit-flip and phase-flip). Its operation elements are

$$E_0 = \sqrt{1-\frac{3\epsilon}{4}} I, \quad E_1 = \sqrt{\frac{\epsilon}{4}} X, \quad E_2 = \sqrt{\frac{\epsilon}{4}} Y, \quad E_3 = \sqrt{\frac{\epsilon}{4}} Z. \tag{2.6}$$

Previously also erasure or loss were mentioned, which behave analogously to the classic case. The qubit is lost and a classical flag is generated, indicating that the qubit has been

erased. Given that there is no way of predicting where a loss has happened, measurements are used to reveal that information and this is why a classical flag is needed. In general this can be analyzed by the erasure channel.

Definition 2.2.5. A single qubit **erasure channel** is a quantum channel that erases a qubit with a probability μ. After such quantum channel an initial state ρ, a 2×2 density matrix, is

$$\mathcal{E}_{\text{erasure}}(\rho) = \mu \left|\perp\right\rangle \left\langle\perp\right| + (1-\mu)\rho, \tag{2.7}$$

where $\left\langle\perp\right| \rho \left|\perp\right\rangle = 0$.

This channel is used as a simple model for physical setups in which the quantum information can be lost, like optical fibers for single-photon communication [20].

2.2.2 Determine the impact of noisy quantum channels

After a quantum system has undergone errors, one is interested in comparing the initial and final states in order to quantify the errors. For that purpose, the concept of secret-key rate is introduced in this text. As previously seen, when information is sent through a quantum channel it is subjected to both losses and errors. Consider now two parties that will take place in the exchange of information, Alice and Bob. The secret-key rate is a parameter that takes into account both the quantity and the quality of the information that Alice sends to Bob through a noisy channel.

The origin of this parameter is in the realm of quantum cryptography, since it is derived from the Quantum Key Distribution (QKD) protocol, which allows for secure communication. This protocol enables two parties, Alice and Bob, to produce a shared random secret key that only they know, which allows the secure communication, since it is used by the two parties to encrypt and decrypt the information they want to share. In this text the focus is on the first quantum cryptography protocol, presented in 1984 by C. H. Bennett and G. Brassard, hence the name BB84 protocol [7].

The highlight of this protocol is that only requires preparation and measurement of single qubit quantum states. The BB84 can be described in five steps [33, 34, 35]:

1. **Preparation:** Alice prepares a quantum state based on the choices at random of two parameters, $\theta_A = \{0, 1\}$ and $x_A = \{0, 1\}$. θ_A determines the basis of the state, where 0 corresponds to the Z and 1 to the X basis and x_A determines the bit that will be prepared. Following this, the possible states are determined by $H^{\theta_A} \left|x_A\right\rangle$, where H denotes the unitary Hadamard gate[a], such that the possible states are $\left|0\right\rangle$, $\left|1\right\rangle$, $\left|+\right\rangle$ and $\left|-\right\rangle$, which in this context are known as the *BB84 states*.

2. **Distribution and measurement:** Alice sends N qubits , all chosen at random from the BB84 states to Bob. Then Bob measures the N states he receives using a basis chosen uniformly at

random $\theta_B = \{0, 1\}$ for each qubit. The outcomes of the measurements are labeled $x_B = \{0, 1\}$ for each qubit.

3. **Sifting:** Alice and Bob share the basis they have used θ_A and θ_B for each of the N qubits and discard all the rounds where the basis do not agree.

4. **Parameter estimation:** Alice chooses n rounds from the remaining ones to test and communicates to Bob which ones she has chosen. For those rounds, they share the bits x_A and x_B. From those values they compute an error rate, which if it exceeds a threshold the protocol is aborted. If the protocol is not aborted the m remaining bits constitute the raw key.

5. **Information reconciliation and privacy amplification:** In this step, Alice and Bob implement an information reconciliation protocol such that Bob can correct his x_Bs from errors that may have occurred during the communication. Lastly, a privacy amplification protocol, is applied to transform the m bits of the raw key, which are partially secure, to a secure key of $l < m$ bits.

[a] The single qubit Hadamard gate is

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

From step 4, Alice and Bob need to estimate the values of the quantum bit error rates (QBERs) in the X and Z basis, Q_X and Q_Z. The QBER in a particular basis corresponds to the probability that the outcomes that Alice and Bob get when they both measure their systems in the corresponding basis are different. To quantify the quality and quantity of this secret secure key, the secret-key rate is defined as the number of key bits generated per channel use [35]. For the presented BB84 protocol, the secret-key rate is computed as follows [33],

$$R_{\text{BB84}} = 1 - h(Q_X) - h(Q_Z), \tag{2.8}$$

where $h(x) = -x \log x - (1-x) \log(1-x)$ is the binary entropy. Since the secret-key rate is related to the QBERs one can see from equation 2.8 that it will decrease with increasing the error in the channel, and it will become zero for a certain bound, called *maximum tolerated channel noise*. For example, if the depolarizing noise channel is considered, the BB84 protocol it is agreed that it can tolerate up to a 11% of QBER, so that for higher QBERs the secret-key rate is zero [33]. The main quantum channel we want to focus on is the the one-way quantum repeater chain for which we need the insights of Quantum Error Correction.

2.3 Quantum Error Correction

In the previous section, the main independent quantum channels have been presented. As a continuation, in this section the focus will be on understanding *Quantum Error-Correcting Codes* (QECC).

Quantum Error Correction (QEC) is strongly inspired by classical error-correcting codes, however, there are three main problems that the quantum realm adds to the classical error-correcting perspective. As stated before, quantum channels introduce a continuum of different errors in a qubit. Moreover, one has to overcome two additional problems, the no-cloning theorem [11], which states that the creation of back-up copies of quantum states is not possible; and the loss of information due to the destructive measurement or collapse of the wave function. Notwithstanding these difficulties, QEC is still possible [32].

In what follows the focus will be on *stabilizer codes*, which have an analogous classical version, the *classical linear codes*.

2.3.1 Classical linear codes

Before understanding QEC, it is instructive to take a look at the classical realm. By doing this one can see that some of the techniques used in classical error correction can be useful in the quantum case, specifically, the theory of classical linear codes has been used to develop many widely used QECC.

Definition 2.3.1. A **classical error-correcting code** C over a finite alphabet Σ is a subset of Σ^n. Where Σ^n is the set of all strings of length n generated by combinations of elements from Σ. [36]

The elements of a code are called the *codewords*. Also, a code has an encoding map associated, which is used to map the possible messages to a different codewords. Therefore, the code must be of the same size as the set of all possible messages. In general the alphabet is $\Sigma = \mathbb{Z}_q$, where $\mathbb{Z}_q$ is the set of integers modulo q. Throughout this text we will only consider classical codes over $\mathbb{Z}_2 = \{0, 1\}$. Then, the codewords will be composed of n-strings of zeros and ones. Moreover, to determine and characterize the error-correcting power of a code one shall talk about the *code distance*.

Definition 2.3.2. The **distance** d of a code is the minimum Hamming distance between any two codewords, i.e.,

$$d = \min_{\substack{x,y \in C \\ x \neq y}} d_H(x, y) \tag{2.9}$$

where the Hamming distance $d_H(x, y)$ between two distinct codewords, x and y, is the number of symbols in which they differ.

Definition 2.3.3. A **classical linear code** C is a classical error-correcting code that encodes k bits into a n bit code. This can be specified by an $n \times k$ *generator matrix* G, with entries from $\mathbb{Z}_2$.

The matrix G maps messages into their encoded equivalent. So that, a k bit message x is encoded as Gx, where the message is treated as a column vector. It is important to note that throughout this section all the operations are done modulo 2. To characterize and "name" any classical linear code the following notation is used. A code C is a $[n, k, d]$ code, where n is the number of bits used to encode a k bit string and d refers to the distance of the code.

Another equivalent formulation for these codes can be introduced in terms of the *parity check matrix* H. All the n-element vectors x of a $[n, k, d]$ code are such that, $Hx = 0$, where H is an $(n-k) \times n$ matrix, with entries from $\mathbb{Z}_2$. Importantly, the distance d of a linear code equals the minimum number of columns of H that are linearly dependent. So a classical linear code can be compactly represented by either its generator matrix or parity check matrix. If one is interested in reading further and how one can move from one description to the other can check Chapter 10 Section 4 in [32].

2.3.2 Stabilizer codes

Stabilizer codes, also known as *additive* quantum codes, are a class of quantum codes whose construction is analogous to classical linear codes. To understand stabilizer codes, the stabilizer formalism will be now introduced as a powerful and compact tool with which to describe an important class of entangled states, the stabilizer states.

Stabilizer Formalism

The stabilizer formalism is a more compact way of describing a quantum state.

Definition 2.3.4. The **n-qubit Pauli group** $\mathcal{P}_n$ [37], is defined as

$$\mathcal{P}_n = \{\pm 1, \pm i\} \times \{I, X, Y, Z\}^{\otimes n}. \tag{2.10}$$

An element of the Pauli group is called a Pauli product.

Definition 2.3.5. An **n-qubit stabilizer group** $\mathcal{S}$ [37] can be defined as an Abelian (commutative) subgroup of the n-qubit Pauli group,

$$\mathcal{S} = \{S_i\} \text{ s.t.} - I \notin \mathcal{S} \text{ and } \forall S_i, S_j \in S, [S_i, S_j] = 0. \tag{2.11}$$

An element from $\mathcal{S}$ is called a stabilizer operator, and the elements in the maximally independent subset $\mathcal{S}_g$ of the stabilizer group are called the stabilizer generators. Independence in this framework means that any stabilizer generator cannot be expressed as a product of other generators. Then, any of the elements in $\mathcal{S}$ can be generated by the product of the stabilizer generators. Thus, a stabilizer group $\mathcal{S}$ can be expressed on terms of its stabilizer generators $\mathcal{S}_g$ and it is denoted by $\mathcal{S} = \langle \mathcal{S}_g \rangle$.

Definition 2.3.6. The **stabilizer state** $|\psi\rangle$, for a given stabilizer group $\mathcal{S}$, can be defined as a simultaneous eigenstate with eigenvalue $+1$ of all the stabilizer operators in $\mathcal{S}$,

$$\forall S_i \in \mathcal{S}, \quad S_i |\psi\rangle = |\psi\rangle . \tag{2.12}$$

It is sufficient if the state is an eigenstate with eigenvalue +1 of the stabilizer generators,

$$\forall g_i \in \mathcal{S}_g, \quad g_i \left|\psi\right\rangle = \left|\psi\right\rangle . \tag{2.13}$$

In general, when talking about a certain stabilizer state $|\psi\rangle$ one can say that it is *stabilized*, or invariant under the action of the operators in $\mathcal{S}$. Then, all the possible states that are stabilized by the subgroup $\mathcal{S}$ form V_S, the *vector space stabilized by* $\mathcal{S}$, and $\mathcal{S}$ is said to be the stabilizer of the space V_S. Let l be the number of elements in the stabilizer generator group $\mathcal{S}_g$. If the number of qubits of the system, n, is equal to l one can uniquely define a quantum state, meaning that V_S is spanned by a single state. If $l < n$, the degrees of freedom can be addressed by using *logical operators*, which commute with all stabilizer generators and are independent from them.

To clearly see how the stabilizer formalism works consider the following two-qubit stabilizer group

$$\mathcal{S}_{\text{Bell}} = \{I_1 I_2, X_1 X_2, Z_1 Z_2, -Y_1 Y_2\}. \tag{2.14}$$

Since all Pauli matrices anti-commute with each other and with the identity matrix it is easy to see that all elements of $\mathcal{S}_{\text{Bell}}$ commute with each other. The stabilizer group is generated by $\{X_1X_2, Z_1Z_2\}$, since $-Y_1Y_2$ can be expressed as a product of those two and $(X_1X_2)(X_1X_2) = (Z_1Z_2)(Z_1Z_2) = I_1I_2$. Therefore, $\mathcal{S}_{\text{Bell}} = \langle\{X_1X_2, Z_1Z_2\}\rangle$. In this case the number of stabilizer generators and qubits of the system is the same, so the stabilizer state of $\mathcal{S}_{\text{Bell}}$ can be uniquely defined, and it is

$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}. \tag{2.15}$$

At the same time, this state is also an eigenvector with eigenvalue +1 of the other stabilizer operators like $-Y_1Y_2$. Now, consider that X_1X_2 is removed from the stabilizer generators group, then the states that are stabilized by $\mathcal{S}_{\text{Bell}}$ are $|00\rangle$ and $|11\rangle$. By choosing logical operators $L_X = X_1X_2$ and $L_Z = Z_1I_2$ one can specify the state in the subspace. For example, if L_X is chosen, then the eigenstate with eigenvalue +1 is $|\Phi^+\rangle$.

To relate this formalism to the previous classical linear codes, the stabilizer generators can be expressed using the *parity check matrix*. This is a $l \times 2n$ matrix, H, whose rows correspond to the stabilizer generators. The matrix is divided into two parts, the left-hand side, H_X is a $l \times n$ binary matrix, which contains 1s to indicate which generators contain Xs; and the right-hand side, H_Z is a $l \times n$ binary matrix, which contains 1s to indicate which generators contain Zs. If there are 1s in both side, this indicates a Y generator. To make this more explicit, consider again $\mathcal{S}_{\text{Bell}}$, its parity check matrix is as follows,

$$H = (H_X|H_Z) = \left(\begin{array}{cc|cc} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{array}\right). \tag{2.16}$$

More explicitly, consider the ith row of the matrix that corresponds to a stabilizer generator g_i,

- If g_i contains I on the jth qubit $\rightarrow$ jth and $n+j$th column elements are 0.
- If g_i contains X on the jth qubit $\rightarrow$ jth column element is 1 and $n+j$th is 0.

- If g_i contains Y on the jth qubit $\rightarrow$ jth and $n+j$th column elements are 1.
- If g_i contains Z on the jth qubit $\rightarrow$ jth column element is 0 and $n+j$th is 1.

The matrix H is analogous to the parity check matrix of the classical linear codes, so that one can also define a generator matrix for the stabilizer codes $G = (G_X|G_Z)$. Then all the properties that are present in the classical linear codes also apply to the stabilizer codes, a detailed view on this analogy is presented in [38].

Clifford Operations

The stabilizer formalism can also be used to describe the dynamics of the stabilizer states under a variety of quantum operations. This will be useful to later on describe quantum error-correcting codes using the stabilizer formalism, and understand the effects of noise and other dynamical processes on those codes.

The subset of all unitary quantum operations that map stabilizer states to stabilizer states are the so-called (local) Clifford operations [39]. The Clifford operation can be defined as an operation that transforms a Pauli product into another Pauli product under its conjugation [37]. Consider a Clifford operation U on a stabilizer state $|\psi\rangle$, defined by the stabilizer group $\mathcal{S} = \langle\{g_i\}\rangle$,

$$U|\psi\rangle = Ug_i|\psi\rangle = Ug_iU^\dagger U|\psi\rangle = g_i'U|\psi\rangle\,, \tag{2.17}$$

where $g_i' = Ug_iU^\dagger$. This equality indicates that the state $U|\psi\rangle$ is stabilized by all g_i'. Due to the fact that U is a unitary Clifford operation, the group $\{g_i'\}$ is also an Abelian subgroup of the Pauli group. Therefore, the state $U|\psi\rangle$ is stabilized by the stabilizer group $\{g_i'\}$. So one can understand the action of U on the stabilizer state as a transformation of the stabilizer group under the conjugation of U. In plain words, to understand how U will affect the state one can only compute how it affects the generators of the stabilizer group.

It can also be the case of an unitary U that takes elements from $\{g_i\}$ to elements of $\{g_i\}$, then this U is said to be a *normalizer* of $\{g_i\}$.

This property can be taken advantage of for certain special unitary operations U. Suppose, for example, that $U = H$, where H denotes the unitary Hadamard gate that acts on a single qubit,

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}. \tag{2.18}$$

Note that,

$$HXH^\dagger = Z; \quad HYH^\dagger = -Y; \quad HZH^\dagger = X. \tag{2.19}$$

This leads to the fact that after a Hadamard gate is applied to a quantum state stabilized by Z, $|0\rangle$, the resulting state will be stabilized by X, $|+\rangle = H|0\rangle$. Similarly, consider a n-qubit state whose stabilizer is $\langle Z_1, Z_2, \ldots, Z_n\rangle$, that is the state $|0\rangle^{\otimes n}$. Applying a Hadamard gate to each of the n qubits, leads to a final state which is stabilized by $\langle X_1, X_2, \ldots, X_n\rangle$, which is easy to see that is $|+\rangle^{\otimes n}$. In the usual description of the

dynamics of a quantum system (state vector) the final state requires to specify 2^n amplitudes, meanwhile in the stabilizer formalism the dynamic transformation can be described by $\langle X_1, X_2, \ldots, X_n \rangle$, which is linear in n. Many other unitary transformations or gates can be studied like this example with the Hadamard gate, and it is precisely the possibility of studying dynamic state transformations by an exponentially smaller amount of memory is what constitutes the main advantage of the stabilizer formalism.

Errors in stabilizer formalism

So far, the basics of the stabilizer codes have been presented. The next step is to see how this formalism can detect and correct errors in a quantum state.

Definition 2.3.7. A $[[n, k]]$ **stabilizer code**, denoted by $C(\mathcal{S})$, is the vector space V_S stabilized by a subgroup $\mathcal{S}$ of $\mathcal{P}_n$ and $\mathcal{S}$ has $n - k$ independent commuting generators, $\mathcal{S} = \langle g_1, \ldots, g_{n-k} \rangle$.

Suppose a stabilizer code $C(\mathcal{S})$ is corrupted by an error $E \in \mathcal{P}_n$,

- If E anti-commutes with an element from $\mathcal{S}$, say that element is g, then $\{g, E\} = 0$ and if $|\psi\rangle \in V_S$, then
$$gE|\psi\rangle = -Eg|\psi\rangle = -E|\psi\rangle . \tag{2.20}$$
Thus $E|\psi\rangle$ is an eigenvector of g with eigenvalue -1 instead of +1, then the error can be detected by measuring g.
- In the case that $E \in \mathcal{S}$, the error does not corrupt the space.
- Finally, if $E \notin \mathcal{S}$ but commutes with all the elements of $\mathcal{S}$, $Eg = gE$ for all $g \in \mathcal{S}$, the error is neither detectable nor correctable. In this case, E is known as a *centralizer* of $\mathcal{S}$, and we will restrict to the case where the centralized are the same as the normalizers, $N(\mathcal{S})$.

Therefore, consider that $\{E_j\}$ is a set of operators in $\mathcal{P}_n$, such that $E_j^\dagger E_k \notin N(\mathcal{S}) - \mathcal{S}$ for all j and k. Then $\{E_j\}$ is a correctable set of errors for the code $C(\mathcal{S})$.

The detection of the presented errors is performed by measuring the generators, $g_1, \ldots, g_{n-k}$ of the stabilizer group to obtain the error syndrome. The syndromes are given by the outcomes of the measurement, $m_1, \ldots m_{n-k}$. Suppose the error E_j occurred, then the error syndrome is given by m_q, such that $E_j g_q E_j^\dagger = m_q g_q$. This syndrome can either correspond to a unique error operator E_j or to many different error operators. In the first case the error is corrected by applying $E_j^\dagger$. However, if two distinct errors, E_j and $E_{j'}$, have the same syndrome, applying $E_j^\dagger$ returns the state into the code space but not necessarily corrects the error [32].

An error E its said to have a *weight* determined by the number of terms in the tensor product which are not equal to the identity. For example, the weight of $X_1 Y_4 Z_9$ is three. Similarly to classical codes, the distance d for stabilizer codes can be presented as the

minimum weight of an element of $N(\mathcal{S}) - \mathcal{S}$. Then a $[[n, k]]$ code with distance d, is an $[[n, k, d]]$ stabilizer code. An important property is that a code is able to detect arbitrary errors on any t qubits if its distance is at least $2t + 1$ [32].

Examples

Here the main stabilizer codes that will be used in the text are presented. In all the examples an initial general state $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$, with $|\alpha|^2 + |\beta|^2 = 1$, is considered, so that $k = 1$. Then this state is encoded in what is called a *logical* qubit, such that $|\Psi\rangle_L = \alpha|0\rangle_L + \beta|1\rangle_L$, where $|0\rangle_L$ is the logical zero state and $|1\rangle_L$ is the logical one state. The logical states are different for each code and they are defined by the stabilizers of the code.

- The **3-qubit repetition code**, or also known as 3-qubit bit flip code which was introduced by Asher Peres in 1985 [40], has generators $g_1 = Z_1Z_2$ and $g_2 = Z_2Z_3$, such that the logical state is,

$$|\Psi\rangle_L = \alpha|000\rangle_{123} + \beta|111\rangle_{123}\,, \tag{2.21}$$

and

$$|0\rangle_L = |000\rangle_{123}\,, \quad |1\rangle_L = |111\rangle_{123}\,. \tag{2.22}$$

This code only has stabilizers composed by Pauli-Z, therefore, it will be only able to detect Pauli-X errors, which are commonly known as bit flips, and here the reason of the code's name. To understand how the syndromes work, suppose an error X_1, then the outcomes of the stabilizer should be $m_1 = -1$ and $m_2 = +1$, because the error anti-commutes with g_1 but commutes with g_2. To prove it, first take the corrupted state,

$$X_1|\Psi\rangle_L = |\Psi'\rangle_L = \alpha|100\rangle_{123} + \beta|011\rangle_{123} \tag{2.23}$$

which, by applying the generators and following the commutation relations of the Pauli matrices[1],

$$\begin{aligned} g_1|\Psi'\rangle_L &= Z_1Z_2|\Psi'\rangle_L \\ &= Z_1Z_2X_1|\Psi\rangle_L \\ &= -X_1Z_1Z_2|\Psi\rangle_L \\ &= -X_1|\Psi\rangle_L \\ &= -|\Psi'\rangle_L \end{aligned} \tag{2.24}$$

so that $m_1 = -1$ as predicted. Similarly for g_2, $m_2 = +1$. Now, all the possible error syndromes and its corrections are presented in the following table:
Also, the logical operators of the logical qubit are,

$$\overline{X} = X_1X_2X_3, \quad \overline{Z} = Z_1. \tag{2.25}$$

[1] $[a, b] = 2i\epsilon_{abc}c$, where a, b and c can be any of the Pauli matrices, and ϵ_{abc} is the Levi-Civita symbol.

Z_1Z_2	Z_2Z_3	Syndrome	Correction
+1	+1	No error	None
+1	-1	X error on qubit 3	X_3
-1	+1	X error on qubit 1	X_1
-1	-1	X error on qubit 2	X_2

Table 2.1: Error syndrome and correction table for the three qubit bit flip code.

- The **3-qubit "rotated" repetition code** or 3-qubit phase flip code is analogous to the one above but in the X basis. Therefore, the generators are $g_1 = X_1X_2$ and $g_2 = X_2X_3$, such that the logical state is,

$$\alpha \left|+++\right\rangle_{123} + \beta \left|---\right\rangle_{123}. \tag{2.26}$$

  Now all stabilizers are composed by Pauli-Xs, so only Pauli-Z errors or phase flips will be detected. The logical operators are also analogous to the bit flip code,

$$\overline{Z} = Z_1Z_2Z_3, \quad \overline{X} = X_1. \tag{2.27}$$

X_1X_2	X_2X_3	Syndrome	Correction
+1	+1	No error	None
+1	-1	Z error on qubit 3	Z_3
-1	+1	Z error on qubit 1	Z_1
-1	-1	Z error on qubit 2	Z_2

Table 2.2: Error syndrome and correction table for the three qubit phase flip code.

- The **9-qubit Shor's code** was introduced by Peter Shor, [41], in 1995. It is a $[[9, 1, 3]]$ code that has distance 3 and therefore, is able to correct an arbitrary single qubit error. This code has the following eight generators presented in Table 2.3.

g_1	Z_1Z_2
g_2	Z_2Z_3
g_3	Z_4Z_5
g_4	Z_5Z_6
g_5	Z_7Z_8
g_6	Z_8Z_9
g_7	$X_1X_2X_3X_4X_5X_6$
g_8	$X_4X_5X_6X_7X_8X_8$
$\overline{X}$	$Z_1Z_2Z_3Z_4Z_5Z_6Z_7Z_8Z_9$
$\overline{Z}$	$X_1X_2X_3X_4X_5X_6X_7X_8X_9$

Table 2.3: Generators and non-unique logical Pauli X and Z operators of the 9-qubit Shor's code.

- The **5-qubit code** or *perfect* code introduced by Laflamme et al. in 1996 [42] is the smallest stabilizer code able to correct an arbitrary single qubit error. It can be referred to as the $[[5, 1, 3]]$ code and has the following generators and logical operators presented in 2.4.

g_1	$X_1Z_2Z_4X_5$
g_2	$X_1X_2Z_3Z_5$
g_3	$Z_1X_2X_3Z_4$
g_4	$Z_2X_3X_4Z_5$
$\overline{X}$	$X_1X_2X_3X_4X_5$
$\overline{Z}$	$Z_1Z_2Z_3Z_4Z_5$

Table 2.4: Generators and logical Pauli X and Z operators of the 5-qubit code. One can note that these generators have a cyclic property and therefore, is rather easy to find other subsets that create four independent generators. This is why in the literature different sets of generators are presented for this stabilizer code. In this text the presented set will be used.

For the last two examples neither syndrome table nor logical state is presented, this is due to the fact that the stabilizer formalism already includes that information by construction.

Quantum circuits for correction

In order to detect an error it is necessary that the generators of the stabilizer are measured. As mentioned at the beginning of this section, one of the main limitations that QEC presents is the destructiveness of quantum measurements. To overcome this problem and be able to measure stabilizers, non-destructive measurements circuits were introduced [32]. For a general single qubit operator M, with eigenvalues ± 1, this circuit looks as in Figure 2.1. Moreover, the structure in Figure 2.1 is used X and Z in the stabilizer measurements.

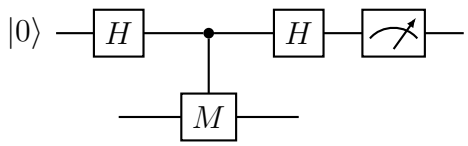


Figure 2.1: Quantum circuit for measuring a single qubit operator M with eigenvalues ± 1. The top qubit is an ancilla qubit used for the measurement, and the bottom qubit is the one being measured.

2.3.3 Code concatenation

We have seen how it is crucial that for quantum error correcting codes to be able to encode several physical qubits into a logical one. Code concatenation, as briefly presented in [43] and [38], exploits this idea by recursively encoding the qubits. So that, starting with one

qubit, this is encoded in several more and all them together for a logical qubit. This first code used is called *inner code or encoding*. Next, this logical qubit is treated as an initial physical qubit and is encoded in another code using more qubits, this code is called the *outer code or encoding*.

From another perspective, the qubits of the codewords of a code C are replaced by encoded qubits of a new code C'. In this case the code C would correspond to the outer code, and C' to the inner code.

Example: 9-qubit Shor's code

A clear example of this procedure is the previously introduced 9-qubit code Shor's code [41], as it concatenates the bit-flip code and the phase-flip code [43].

First, consider an initial physical qubit in state $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$ encoded in an outer code C, the phase-flip code, $|\Psi\rangle_C = \alpha|+++\rangle + \beta|---\rangle$. Next, assume the bit-flip code as the inner code C', so that each of the physical qubits in the phase flip are now encoded in a bit-flip code. Therefore, each $|\pm\rangle$ becomes $\frac{1}{\sqrt{2}}(|000\rangle \pm |111\rangle)$, so the state finally is,

$$|\Psi\rangle_{CC'} = \alpha|0\rangle_{CC'} + \beta|1\rangle_{CC'} \tag{2.28}$$

where,

$$\begin{aligned} |0\rangle_{CC'} &= \frac{1}{2\sqrt{2}}(|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \\ |1\rangle_{CC'} &= \frac{1}{2\sqrt{2}}(|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle). \end{aligned} \tag{2.29}$$

Note that now three blocks of bit-flip encoding are present, which have the following stabilizers, (Z_1Z_2, Z_2Z_3), (Z_4Z_5, Z_5Z_6) and (Z_7Z_8, Z_8Z_9). At the same time this three block form a single phase-flip block that has stabilizers $X_1X_2X_3X_4X_5X_6$ and $X_4X_5X_6X_7X_8X_9$, these correspond to $\overline{X}_1\overline{X}_2$ and $\overline{X}_2\overline{X}_3$ from the phase-flip code, and where $\overline{X}$ is the logical operator from the bit-flip code. The non-unique logical operators in Shor's code are $\overline{Z}_{CC'} = Z_1Z_4Z_7$ and $\overline{X}_{CC'} = Z_1Z_2Z_3$.

2.4 Graph states, tree state and tree code

In this section a detailed explanation of the tree code is presented. The tree code is a quantum error correcting code based on a graph-like structure that enjoys many properties of a set of states a known as the cluster states. The introduction on graph states is rather superficial and focused on the ingredients needed to understand the concepts of the tree code. For a more detailed and extensive approach one can look at [44].

2.4.1 Graph states

A graph is a collection of vertices and a description of which vertices are connected by an edge. Each graph can be represented by a diagram where a vertex is represented by a point and the edges by arcs joining two vertices, which do not need to be distinct. In the context of this text, vertices will denote qubits and the edges or bonds between them will denote a certain entangling operation between them.

Formally, a graph is a pair

$$G = (V, E) \tag{2.30}$$

of a finite set $V = \{1, \ldots, n\}$ and a set $E \subset [V]^2$, whose elements are subsets of V with two elements each [44]. The elements of V are called vertices and the elements of E edges. In the case of simple graphs, which is a graph that contains neither loops (edges connecting vertices with itself) nor multiple edges, the definition of graph states can be presented in terms of their stabilizers.

Definition 2.4.1. Let $G = (V, E)$ be a graph [44]. A **graph state vector** $|G\rangle$ is the unique common eigenvector to the set of independent commuting observables:

$$K_\nu = X_\nu \bigotimes_{w \in \mathcal{N}_\nu} Z_w$$

where $\mathcal{N}_\nu$ denotes the neighbours of qubit ν, which are the qubits directly linked to qubit ν. The stabilizer states correspond to the eigenstates with eigenvalue $+1$ of all the K_ν for all $\nu \in V$. The stabilizer subgroup $\mathcal{S}$ of the graph state is generated by the set $\mathcal{S} = \langle \{K_\nu \mid \nu \in V\} \rangle$.

In general to generate a simple graph state $|G\rangle$ with stabilizers $\mathcal{S} = \langle \{K_\nu \mid \nu \in V\} \rangle$, one should prepare each of the qubits in V in the $|+\rangle$ state, by applying a Hadamard gate to an initial state $|0\rangle$. Then the edges E correspond to an entangling operation between the pairs of connected qubits, which operation corresponds to a CPHASE gate. The CPHASE or controlled phase-flip gate is a unitary gate that acts on a 2-qubit system, and is able to create entanglement when applied to a product state,

$$\text{CPHASE} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \tag{2.31}$$

An important property of graph states comes from the definition of their generators. Consider measuring a graph state vector $|G\rangle$ following the pattern given by K_ν, i.e. measuring the qubit at vertex ν in the X-basis and the vertices w in $\mathcal{N}_\nu$ in the Z-basis. The outcomes of such measurements, $m_\nu^X = \pm 1$ and $m_w^Z = \pm 1$, are constrained by K_ν, namely,

$$m_\nu^X \prod_{w \in \mathcal{N}_\nu} m_w^Z = 1. \tag{2.32}$$

These constraints can be used as a tool to deal with losses in a graph state. To illustrate this, consider a simple graph of only two qubits. The two qubits in the graph state will

have labels 0 and 1. First, to generate the graph state the circuit in Figure 2.2 will take place.

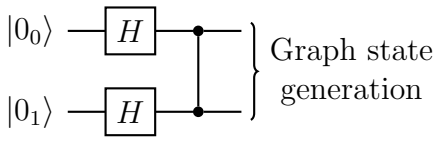


Figure 2.2: Quantum circuit for the two-qubit simple graph generation.

So that the final state is,

$$|\psi_{01}\rangle = \frac{1}{\sqrt{2}}(|0_0\rangle\,|+_1\rangle + |1_0\rangle\,|-_1\rangle) \tag{2.33}$$
$$= \frac{1}{\sqrt{2}}(|+_0\rangle\,|0_1\rangle + |-_0\rangle\,|1_1\rangle).$$

Now is easy to identify the stabilizers of this simple example. Those are X_0Z_1 and Z_0X_1, such that,

$$X_0Z_1\,|\psi_{01}\rangle = |\psi_{01}\rangle\,, \tag{2.34}$$
$$Z_0X_1\,|\psi_{01}\rangle = |\psi_{01}\rangle\,. \tag{2.35}$$

Take the first one, if the observable X_0 is measured then the outcome of Z_1 is known with certainty. Therefore, from now on we can say that X_0 is an indirect measurement of the observable Z_1. Thus, if qubit 1 undergoes a loss error and so it is no longer available, Z_1 can still be measured indirectly. Performing indirect measurements is a helpful tool that will be used recurrently in this text.

Each graph state vector $|G\rangle$ corresponds uniquely to a graph G, meaning that two different graphs, $G = (V, E)$ and $G' = (V, E')$, cannot describe the same graph state[2], $|G\rangle \neq |G'\rangle$. However, two different graphs might have graph states that are equal up to some local unitary (LU) operation. Two graphs $G = (V, E)$ and $G' = (V, E')$ are *LU-equivalent*, if there exists a local[3] unitary U such that,

$$|G'\rangle = U\,|G\rangle\,. \tag{2.36}$$

Now,

$$\Sigma' := U\mathcal{S}U^\dagger = \{UsU^\dagger | s \in \mathcal{S}\} \tag{2.37}$$

where S is the stabilizer of $|G\rangle$, then $s'\,|G'\rangle = |G'\rangle$ for every $s' \in \Sigma'$. Now, one would say that Σ' is a stabilizing subgroup of $|G'\rangle$. However, in general Σ' is not equal to the stabilizer of $|G'\rangle$, since in general Σ' is not a subgroup of a Pauli subgroup. For Σ' to be the stabilizer group of $|G'\rangle$ one needs to consider U to be in the local Clifford group of n qubits[4] [39], as seen in 2.3.2. So, from now on two graph states $|G\rangle$ and $|G'\rangle$ are

[2]See proof in [44]

[3]Locality refers to the systems associated with vertices of G and G'

[4]The local Clifford group on n qubits is $\mathcal{C}_n^1 := \{U \in \mathbf{U}(2)_V | U\mathcal{P}_nU^\dagger = \mathcal{P}_n\}$, which is the n-fold tensor product of the one-qubit Clifford group $\mathcal{C}^1 := \{U \in \mathbf{U}(2) | U\mathcal{P}U^\dagger = \mathcal{P}\}$

said to be *LC-equivalent* if and only if they are related by some local Clifford unitary U, $|G'\rangle = U|G\rangle$. As stated in *proposition 4* in [44], any stabilizer state is LC-equivalent to some graph state, the proof of such statement can be found in [45]. Similarly to this statement, more generally for all stabilizer codes, any stabilizer code is LC-equivalent to some graph code, quantum codes where an underlying graph determines them [46].

The action of local Clifford operations in graph state can be generally described by the *local complementation*, which is a rule for graph transformation.

Definition 2.4.2. A **local complementation** τ_ν acts on a vertex ν of a graph G by inverting the edges connecting the neighbours of ν [44]. After such action, the graph is $G' = \tau_\nu(G)$.

Then, by local complementation of a graph G at some vertex $\nu \in V$, one obtains an LC-equivalent graph state $|G'\rangle = |\tau_\nu(G)\rangle = U^\tau_\nu(G)|G\rangle$, where $U^\tau_\nu(G) = e^{-i\frac{\pi}{4}X_\nu} e^{i\frac{\pi}{4}\otimes_{w\in\mathcal{N}} Z_w}$ is a local Clifford unitary, . Therefore, two graph states are LC-equivalent if the corresponding graphs are related by a sequence of local complementations, i.e. $G' = \tau_{\nu_1} \circ \cdots \circ \tau_{\nu_n}(G)$ for some $\nu_1, \ldots, \nu_n \in V$. This is usually referred to as the LC-rule and this equivalence is proofed in. [44].

Local Pauli measurements

The measurement of some Pauli operator X, Y or Z at a single vertex ν in a graph state is a useful trick used throughout the text. In general, it consists on a sequence of local complementations together with the deletion of vertex ν, together with some LC-unitaries at vertex ν.

Generically, a Pauli measurement of the graph state $|G\rangle$ at a vertex ν results in a graph state $|G'\rangle$ with the remaining unmeasured vertices.

Measuring vertex ν in the Z basis: Removes the qubit ν, up to a local Z rotation.

Measuring vertex ν in the Y basis: Applies a local complementation τ_ν and removes the qubit ν, up to a LC-unitary.

Measuring vertex ν in the X basis: Chooses any vertex $u \in \mathcal{N}_\nu$, applies a local complementation τ_u, applies the rule for measuring in Y and applying τ_u again, up to a LC-unitary.

The following notation $P^{i,\pm}_\nu$ is used to describe projective measurements where, the sub-index ν denotes on which qubit the measurement is performed on, and $i = X, Y, Z$ indicates the basis of the measurement and the super-index $\pm$ denotes the outcome of such measurement, that can be ± 1.

Now, considering LC-unitarity, a projective measurement of X, Y or Z on the qubit associated with a vertex ν in a graph G yields up to local unitaries, $U^{i,\pm}_\nu$, a new graph

state $|G'\rangle$ on the remaining vertices. The resulting graph G' is:

$$P_\nu^{Z,\pm}|G\rangle = \frac{1}{2}|Z,\pm\rangle_\nu \otimes U_\nu^{Z,\pm}|G-\nu\rangle\,, \tag{2.38}$$

$$P_\nu^{Y,\pm}|G\rangle = \frac{1}{2}|Y,\pm\rangle_\nu \otimes U_\nu^{Y,\pm}|\tau_\nu(G)-\nu\rangle\,, \tag{2.39}$$

$$P_\nu^{X,\pm}|G\rangle = \frac{1}{2}|X,\pm\rangle_\nu \otimes U_\nu^{X,\pm}|\tau_u(\tau_\nu\circ\tau_u(G)-\nu)\rangle\,, \tag{2.40}$$

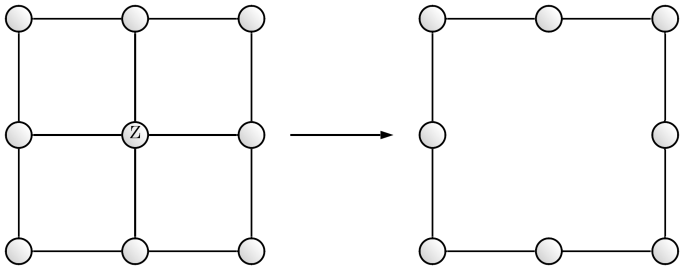
for any choice of some $u \in \mathcal{N}_\nu$. The local unitaries $U_\nu^{i,\pm}$ are:

$$U_\nu^{Z,+} = 1, \qquad U_\nu^{Z,-} = \otimes_{w\in\mathcal{N}(\nu)} Z^w \tag{2.41}$$

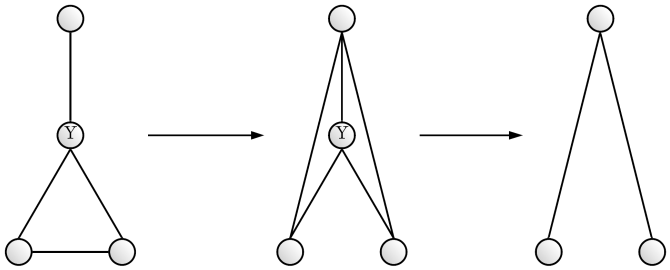
$$U_\nu^{Y,+} = \otimes_{w\in\mathcal{N}_\nu}(-iY_w)^{\frac{1}{2}}, \qquad U_\nu^{Y,-} = \otimes_{w\in\mathcal{N}_\nu}(iY_w)^{\frac{1}{2}} \tag{2.42}$$

$$U_\nu^{X,+} = (iY_u)^{\frac{1}{2}} \otimes_{w\in\mathcal{N}_\nu-\mathcal{N}_u-\{u\}} Z_w, \qquad U_\nu^{X,-} = (-iY_u)^{\frac{1}{2}} \otimes_{w\in\mathcal{N}_u-\mathcal{N}_\nu-\{\nu\}} Z_w. \tag{2.43}$$

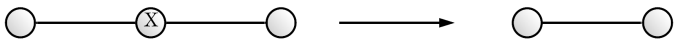
The local unitary for the measurement of X depends on the choice of u, but the different choices result in graph states that ate equivalent via an LC-unitary. In Figure 2.3 some examples with a graphical representation of the measurements in the different basis is illustrated.



(a) Central qubit measured in the Z basis.



(b) Central qubit measured in the Y basis.



(c) Central qubit measured in the X basis.

Figure 2.3: Graphical representation of measuring in the different basis a qubit in a graph. (a) in the Z basis, (b) in the Y basis and (c) in the X basis. At the left, the initial graph G is represented and at the right the final graph G' after the Pauli measurement is represented. All the final graphs are up to a LC-unitary, as described in equation 2.38. Additionally, in (b) an intermediate step has been drawn for clarity.

Cluster states

A cluster state is a particular instance of a graph state, where the underlying graph is a connected subset of a d-dimensional lattice. But in the framework of quantum computation, only cluster states of dimension two and higher are useful as stated in [47]. In the next part of this section, a more deep description of an example of a two-dimensional cluster state, the tree state, is presented.

2.4.2 Tree states

The specific structure of a tree state can be characterized by a branching vector $\bar{t} = [b_0, b_1, \ldots, b_d]$, which explicitly denotes the connectivity of the tree starting at the top node, which from now on it will be referred to as the root qubit, through the d levels of the tree.

Since the tree state is a simple graph state, its generation and stabilizers are analogous to the general description for a graph state. To be able to refer to each one of the qubits in the tree unequivocally, one should label the qubits as follows in Table 2.5.

Labels		
Root qubit	$\rightarrow$	0
1st level qubits	$\rightarrow$	$1, \ldots, b_0$
2nd level qubits	$\rightarrow$	$i1, \ldots, ib_1$ [a]
3rd level qubits	$\rightarrow$	$ij1, \ldots, ijb_2$ [b]
	$\vdots$	
dth level qubits	$\rightarrow$	$i \ldots k1, \ldots, i \ldots kb_d$ [c]

[a] $i = 1, \ldots, b_0$, i denotes the label of the 1st level qubit they are linked to.
[b] $j = 1, \ldots, b_1$, j denotes the label of the 2nd level qubit they are linked to.
[c] $k = 1, \ldots, b_{d-1}$, k denotes the label of the $d-1$th level qubit they are linked to.

Table 2.5: Labels of the vertices/qubits in a tree code of branching vector $\bar{t} = [b_0, b_1, b_2, \ldots, b_d]$

A visualization of an example of such tree states with the proposed labelling of the qubits can be seen in Figure 2.4.

Redundancy of the tree state

As seen for the general simple graph states, the outcomes of the measurements in a stabilizer are constrained and this is used to perform indirect measurements. Now a

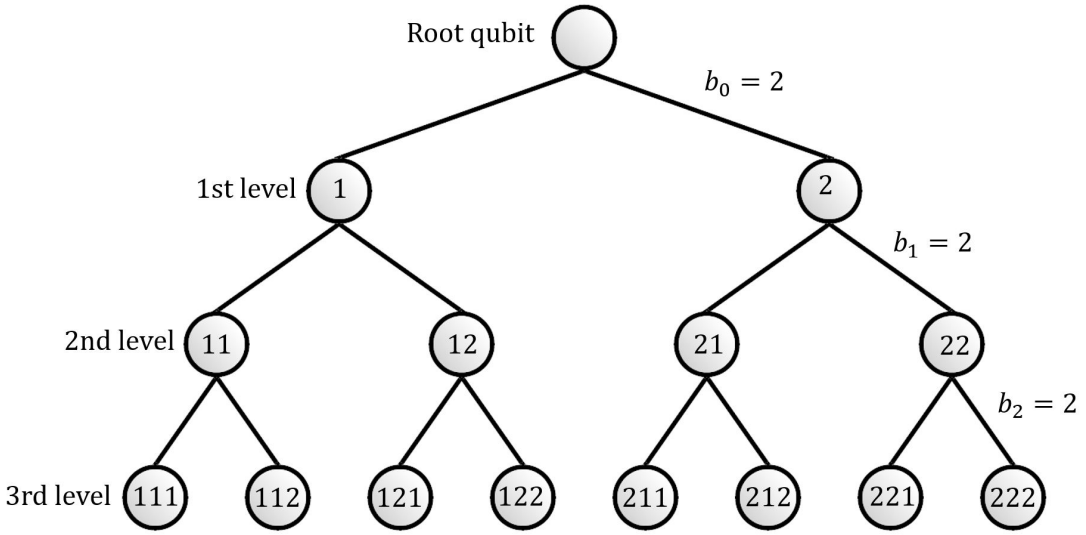


Figure 2.4: Tree of depth 3 with branching vector $[b_0, b_1, b_2] = [2, 2, 2]$.

detailed look at indirect measurements on tree states is presented, as this is one of the crucial features of these tree clusters.

In general, any given qubit within the tree state can be removed indirectly by performing measurements on a subset of qubits below it in the tree. Now, consider that a Z measurement in qubit i, a first-level qubit, is to be performed. To do so, one has to pick one of the branches under qubit i and measure the 2nd level qubit, ij, in the X basis and 3rd level qubits, ijk for $k = 1, \dots b_2$, in Z basis. Therefore, the performed measurement is

$$X_{ij} \otimes_{m \in \mathcal{N}(ij)-i} Z_m, \tag{2.44}$$

where the sub-indices in the Pauli matrices denote in which qubit they are applied to. This expression is almost exactly the stabilizer for qubit ij, only Z on one of the neighbours is missing. The latter is exactly the first-level qubit i, the one where an indirect Z measurement is to be performed on. Then, the stabilizer for qubit ij is as follows,

$$K_{ij} = X_{ij} \otimes_{m \in \mathcal{N}(ij)} Z_m. \tag{2.45}$$

Measuring K_{ij} would have outcome $+1$, since it is a stabilizer of the tree state, then the outcomes of measuring $X_{ij} \otimes_{m \in \mathcal{N}(ij)-i} Z_m$ and Z_i, will be both either ± 1, but they need to be the same. This is how an indirect Z measurement is applied to a first-level qubit in a tree state. An analogous procedure can be applied to any qubit in any level of the tree, changing the levels of all the implied qubits in the indirect measurement. From this, one can clearly see that the tree state has some redundancy to it making the tree state a good candidate to deal with losses and errors given that a same outcome can be obtained in a variety of ways.

2.4.3 Tree code

Now that the tree state has been introduced, one can use it to encode and protect information, thus generating the *tree code*, which has been used previously in [30], [48]

and in [15]. As previously stated, the tree state can be defined only with its stabilizers and its branching vector, therefore, one can easily see that the tree code belongs to the family of QECC of the stabilizer codes. Particularly, the tree code takes one qubit and encodes it in a tree with branching vector $\bar{t} = [b_0, b_1, \ldots, b_d]$, so that $n = b_0 + \cdots + b_0 b_1 \ldots b_d = \sum_{i=0}^{d} \prod_{j=0}^{i} b_j$ and $k = 1$. On the following parts of this section a detailed explanation of the tree code is presented.

Encoding procedure

Consider an initial state $\alpha |0\rangle + \beta |1\rangle$ and a tree state of branching vector $\bar{t}$. A Bell state measurement between the qubit with the initial state and root qubit needs to be performed in order to encode the initial state into the tree code. In Figure 2.5 the encoding procedure is drawn.

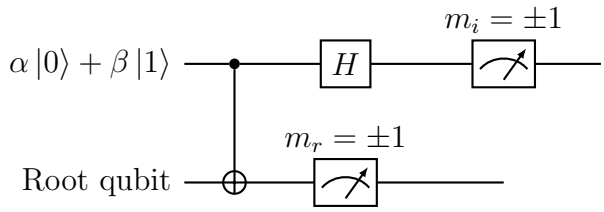


Figure 2.5: Quantum circuit for the encoding procedure of the tree code.

The Bell state measurement has four possible outcomes ($m_i = \pm 1$ and $m_r = \pm 1$), and the encoded state or tree code will depend on them. In general, any measurement in the tree with outcomes ± 1 is such that the state after such measurement will depend on them. The post-measurement state with outcome $+1$ will be considered the *correct state*. Thus, throughout the following sections the concept of *correction* will be used after any kind of measurement in the tree with outcome -1, where the correction will be some kind of logical operator on the tree.

The corrections for the encoding procedure to get the post-measurement state with outcomes $m_i = +1$ and $m_r = +1$ are: if $m_i = -1$ a logical Z needs to be applied to the tree and if $m_r = -1$ a logical X needs to be applied to the tree.

Stabilizers of the encoded tree

It is important to note that after performing the encoding, the tree "loses" the root qubit and therefore, some of the stabilizers of the tree state are no longer possible to measure, those are $K_0 \propto X_0$, $K_i \propto Z_0$ for any $i = [1, b_0]$.

In general, the tree state before encoding is,

$$|\Psi\rangle \propto |0\rangle_0 |\tau_+\rangle + |1\rangle_0 |\tau_-\rangle . \tag{2.46}$$

where the sub-index 0 denotes the root qubit and $|\tau_\pm\rangle$ refers to the states of other qubits of the tree, which are also known as the *sub-trees* of the tree. These sub-trees have as a

root qubit each of the first-level qubits. After the encoding procedure is done, that state is,

$$|\Psi\rangle \propto c_1 |\tau_+\rangle + c_2 |\tau_-\rangle , \tag{2.47}$$

where c_1 and c_2 are two complex numbers that depend on the encoded state and the outcomes of the Bell state measurement. The states $|\tau_\pm\rangle$ can also be interpreted as the logical states of the tree code. If the state to be encoded was $|0\rangle$, the logical state would be $|\tau_+\rangle$, therefore, it will be considered the logical 0 and same follows for $|\tau_-\rangle$, which will be considered the logical 1.

Now, consider T_i to be an operator such that $T_i = Z_0 K_i = X_i \prod_{j=1}^{b_1} Z_{ij}$, then $K_i = Z_0 T_i$. Since, $K_i |\Psi\rangle = |\Psi\rangle$, then $Z_0 |\Psi\rangle = T_i |\Psi\rangle$,

$$Z_0 |\Psi\rangle \propto |0\rangle_0 |\tau_+\rangle - |1\rangle_0 |\tau_-\rangle , \tag{2.48}$$

thus it follows that $T_i |\tau_\pm\rangle = \pm |\tau_\pm\rangle$. Therefore, the operator $S_{ij} = T_i T_j$ for $i, j = [1, b_0]$ is such that $T_i T_j |\Psi\rangle = |\Psi\rangle$, not that for $i = j$, $S_{ij} = I$. Moreover, since T_i does not affect the root qubit, then the operators S_{ij} are the stabilizers for the first-level qubits after the encoding procedure. It is important to note that these stabilizers enforce correlations between sub-trees in a same tree. In contrast, the already known stabilizers from the tree state are only internal to each sub-tree.

Logical operations of the encoded tree

As said before, logical operations of the encoded tree will be used often as corrections that need to be applied to the tree after any qubit measurement. The main logical operations that will be used are the logical Pauli-X and the logical Pauli-Z. Generically and by construction, operations on the tree code can be done either directly or indirectly, the intuition for an indirect operation arises from the stabilizers of the tree.

- **Logical Pauli-X:** The operation can be done directly by applying Z to each of the first-level qubits of the tree.

$$LX = \prod_{i=1}^{b_0} Z_i. \tag{2.49}$$

  As any operator acting on a tree, this logical operator is not uniquely defined and hence it can be performed indirectly on the qubits below the first level. If one recalls the stabilizer $Z_i X_{ij} \prod_{k=1}^{b_2} Z_{ijk}$, the effect of Z to the first-level qubit, Z_i is the same as $X_{ij} \prod_{k=1}^{b_2} Z_{ijk}$ on the qubits below i. From this we learn that the direct one is easier or "cheaper" since only requires a single operation per sub-tree, while the indirect one requires more than one for trees of depth three or more. Regardless of the choice, the effect of such logical operation on the logical states defined previously, needs to be as follows,

$$LX |\tau_\pm\rangle = |\tau_\mp\rangle . \tag{2.50}$$

- **Logical Pauli-Z:** This operation can be performed directly in a single sub-tree by applying X on the first-level and Z on the second-level and it is given by,

$$LZ = X_i \prod_{j=1}^{b_1} Z_{ij}, \tag{2.51}$$

where $i \in \{1, b_0\}$. In this case one can perform Z on one of the second-level qubits indirectly. This is analogous to the indirect Z measurement in logical X, but one level below, $X_{ijk} \prod_{l=1}^{b_3} Z_{ijkl}$. Note that to do this one needs to have deep trees. Any way, the effect of such logical operation on the logical states defined previously, needs to be as follows,

$$LZ \left|\tau_{\pm}\right\rangle = \pm \left|\tau_{\pm}\right\rangle . \tag{2.52}$$

Note that the direct operation is the same as the operators T_i.

Decoding procedure

To recover the state that was encoded in a tree one can make use of the properties of the graph states. The decoding procedure consists on *detaching* all the qubits from the tree but one of the first-level qubits which will be left with the same state as the qubit that was encoded.

First, one needs to pick one of the first-level qubits to be the final qubit. Then, for all the other first-level qubits and their corresponding sub-trees the recipe in Table 2.6 is followed.

Detach a sub-tree		
1st level qubit	$\rightarrow$	measured in Z basis
2nd level qubits	$\rightarrow$	measured in X basis
3rd level qubits	$\rightarrow$	measured in Z basis
	...	

Table 2.6: Measurements to perform in a sub-tree to detach it from the tree.

This choice of measurements is motivated by the properties of those measurements on the graph state. Recall that measuring a qubit in Z basis is equivalent to removing that qubit and losing any connectivity from the rest of the graph. Thus, makes sense that the remaining first-level qubits are measured in Z so that each sub-tree is detached from the graph. Moreover, one can use the redundancy of the measurements of the tree to also perform this Z on the first-level qubit in a sub-tree indirectly many times using all the sub-sub-trees from the sub-tree. Since the outcomes of all the Z measurements, both direct and indirect, in a sub-tree are correlated they can be majority voted[5]. In a scenario where no errors are considered this is not necessary, but this is almost never the case so that performing this majority voting also ensures some redundancy against errors.

[5]A majority vote consist on deciding an outcome from a set of options. The final outcome corresponds to the most repeated option in the set.

After this, one is left with one entire sub-tree where the qubits on the second level or below need to be detached. Once again, one can think about the leaves of this first-level qubit as sub-trees to be detached. Therefore, the recipe presented in Table 2.7 is followed.

Detach a second-level sub-tree		
2nd level qubits	$\rightarrow$	measured in Z basis
3rd level qubits	$\rightarrow$	measured in X basis
4th level qubits	$\rightarrow$	measured in Z basis
	...	

Table 2.7: Measurements to perform in a sub-tree in the second level to detach it from the tree.

By doing this one recovers a single qubit state, which depends on the outcomes of the measurements performed in the decoding. Those measurements are direct and indirect Z measurements and thus those leave the state up to a Pauli-Z correction as seen in part 2.4.1, in particular in equation 2.41. Therefore, if the product of all the outcomes in the decoding is -1, the single qubit state needs a correction of a Pauli-Z. When dealing with losses one needs to take into account that in the first set of measurements, at least one Z measurement per sub-tree, either direct or indirect, needs to be successful in order to be able to know the right correction. Moreover, in the last step the Z measurements on the second-level qubits need to be successful either directly or indirectly also to be able to retrieve the right correction. Finally, to recover the information in the computational basis an H gate needs to be applied. From this procedure its easy to get an intuition on the recursion property inherent to the tree code.

Example: [3, 2] tree code

To show all the properties of the tree code a simple example is presented. Consider the branching vector [3, 2], then the tree state for the 10 qubits is,

$$|\Psi\rangle = \frac{1}{2^{4/2}} \left(|0\rangle(|0\rangle|+,+\rangle + |1\rangle|-,-\rangle)^{\otimes 3} + |1\rangle(|0\rangle|+,+\rangle - |1\rangle|-,-\rangle)^{\otimes 3} \right), \tag{2.53}$$

where **this** color represents the root qubit, **this** the first-level qubits and **this** the second-level qubits. This state is generated by preparing all the qubits in the plus state and performing a CPHASE between all the connected qubits. Also, $|\tau_{\pm}\rangle$ can be identified,

$$|\tau_{\pm}\rangle = (|0\rangle|+,+\rangle \pm |1\rangle|-,-\rangle)^{\otimes 3}. \tag{2.54}$$

Now, consider a qubit with state $\alpha|0\rangle + \beta|1\rangle$ that is encoded into the [3, 2] tree state by means of a Bell state measurement, the resulting state is,

$$|\Psi\rangle = \frac{(1+m_r)\alpha + (1-m_r)\beta}{4\sqrt{2}} |\tau_+\rangle + m_i \frac{(1-m_r)\alpha + (1+m_r)\beta}{4\sqrt{2}} |\tau_-\rangle, \tag{2.55}$$

where $m_r, m_i = \pm 1$ are the outcomes of the Bell state measurement of the root qubit and the information qubit respectively.

Now that the state of the tree code is explicitly presented for the [3, 2] case, its stabilizers can be studied. In this case the T_i operators are,

$$T_1 = X_1 Z_{11} Z_{12}, \quad T_2 = X_2 Z_{21} Z_{22}, \quad T_3 = X_3 Z_{31} Z_{32}. \tag{2.56}$$

Each T_i has the same effect on a different sub-tree, but that results on the same effect in the tree, which is as follows,

$$T_i \left|\tau_\pm\right\rangle = (\left|1\right\rangle\left|-,-\right\rangle \pm \left|0\right\rangle\left|+,+\right\rangle)(\left|0\right\rangle\left|+,+\right\rangle \pm \left|1\right\rangle\left|-,-\right\rangle)^{\otimes 2}, \tag{2.57}$$

such that,

$$T_i \left|\tau_\pm\right\rangle = \pm \left|\tau_\pm\right\rangle \tag{2.58}$$

for any $i \in \{1, 2, 3\}$ and this holds for any outcome of the Bell state measurement. As stated previously, the new stabilizers are, T_1T_2, T_2T_3 and T_1T_3. Note that $(T_1T_2)(T_2T_3) = T_1T_3$, and so only T_1T_2 and T_2T_3 are necessary. To prove that those are stabilizers, one only needs to apply them to the state in equation 2.55. Now, there are eight generators for the [3, 2] code, that has 9 qubits, are given in Table 2.8.

g_1	Z_1X_{11}
g_2	Z_1X_{12}
g_3	Z_2X_{21}
g_4	Z_2X_{22}
g_5	Z_3X_{31}
g_6	Z_3X_{32}
g_7	$X_1Z_{11}Z_{12}X_2Z_{21}Z_{22}$
g_8	$X_2Z_{21}Z_{22}X_3Z_{31}Z_{32}$

Table 2.8: Generators of the [3, 2] tree code.

This code's generators are rather similar to Shor's ones, in Table 2.3. Here are the steps to make the similarities between these two codes more apparent. First, note that the generators are not unique, for example, g_2 can be redefined to $g_2' = g_1g_2 = X_{11}X_{12}$, and similarly for g_4 and g_6. Then, consider an H gate is applied to each of the second-level qubits, so that now the stabilizers are the ones presented in Table 2.9.

Generators	[3, 2] tree code	Shor's code
g_1	Z_1Z_{11}	Z_1Z_2
g_2	$Z_{11}Z_{12}$	Z_2Z_3
g_3	Z_2Z_{21}	Z_4Z_5
g_4	$Z_{21}Z_{22}$	Z_5Z_6
g_5	Z_3Z_{31}	Z_7Z_8
g_6	$Z_{31}Z_{32}$	Z_8Z_9
g_7	$X_1X_{11}X_{12}X_2X_{21}X_{22}$	$X_1X_2X_3X_4X_5X_6$
g_8	$X_2X_{21}X_{22}X_3X_{31}X_{32}$	$X_4X_5X_6X_7X_8X_9$

Table 2.9: Comparison of the generators of the [3, 2] tree code and Shor's code.

Here one can identify the relation between the indices of the two codes. Remember that the Shor's code is obtained through code concatenation of the bit flip code and the phase

flip code. This is such that qubits 1, 4 and 7 are encoded into a phase flip code and then each of those qubits is encoded into a bit flip code, such that (1, 2, 3), (4, 5, 6) and (7, 8, 9) form three blocks of bit flip code and all together form a phase flip code.

By looking at the 'modified' generators of the [3, 2] tree code and of the Shor's code, one can see that the three qubits of each sub-tree form a bit flip block and then these three form a phase flip block. Or analogously, the first-level qubits are encoded in a phase flip code and then each of those qubits is encoded in a bit flip code with their second-level qubits. Therefore, one can say that the [3, 2] code and the Shor's code are LC-equivalent.

Moreover, one can see how a tree code is a concatenated code, since the structure and the entanglement relation between the root qubit and the first-level qubits is the same as the one for a first-level qubit with its leaves or second-level qubits, and so on depending on the depth of the tree.

Next, from 2.55 consider the outcomes $m_r, m_i = +1$, that correspond to the *correct* state,

$$|\Psi\rangle_{\text{correct}} = \frac{1}{2\sqrt{2}}(\alpha\,|\tau_+\rangle + \beta\,|\tau_-\rangle). \tag{2.59}$$

If instead, the outcomes were $m_r = -1$ and $m_i = +1$, the state would look as follows,

$$|\Psi\rangle_{m_r=-1} = \frac{1}{2\sqrt{2}}(\beta\,|\tau_+\rangle + \alpha\,|\tau_-\rangle). \tag{2.60}$$

To correct this state an unitary operation such that $U\,|\tau_\pm\rangle = |\tau_\mp\rangle$ needs to be applied. This operation is the logical Pauli-X, $U = LX$ and in this case this can be performed in two ways for each sub-tree. The state of a sub-tree is,

$$|0\rangle|+,+\rangle \pm |1\rangle|-,-\rangle \tag{2.61}$$

and the operation needs to be such that the result is

$$|0\rangle|+,+\rangle \mp |1\rangle|-,-\rangle. \tag{2.62}$$

This can be done by either applying Z on the **first-level qubit**, or applying X to both of the **second-level qubits**. Next, consider the state with outcomes $m_r = +1$ and $m_i = -1$,

$$|\Psi\rangle_{m_i=-1} = \frac{1}{2\sqrt{2}}(\alpha\,|\tau_+\rangle - \beta\,|\tau_-\rangle). \tag{2.63}$$

To correct this state an unitary operation such that $U\,|\tau_\pm\rangle = \pm\,|\tau_\pm\rangle$ needs to be applied. As seen before, the logical Pauli-Z has this effect, therefore $U = LZ$, which in this case is done by applying an X to the first-level qubit and Z to the second-level qubit on one of the sub-trees, which is essentially applying the operator T_i in the ith sub-tree, as shown in equations 2.57 and 2.58. Finally, if the outcomes are $m_r = -1$ and $m_i = -1$,

$$|\Psi\rangle_{m_r,m_i=-1} = \frac{1}{2\sqrt{2}}(\beta\,|\tau_+\rangle - \alpha\,|\tau_-\rangle). \tag{2.64}$$

The operations that need to be applied are both LX and LZ.

Lastly, the decoding procedure is presented in detail for this example. First, two of the sub-trees need to be detached following the sequence in Table 2.6. Consider the state

of a single sub-tree 2.61, then the **first-level qubit** is measured in Z basis and the **second-level qubits** in the X basis. There are two possible outcomes, if the product of the outcomes is +1, the post-measurement state is $|0\rangle|+,+\rangle$ and if its -1, the post-measurement state is $\pm|1\rangle|-,-\rangle$. After doing this for both sub-trees, the state of the tree is[6],

$$\frac{1}{\sqrt{2}}(\alpha(|0\rangle|+,+\rangle + |1\rangle|-,-\rangle) \pm \beta(|0\rangle|+,+\rangle - |1\rangle|-,-\rangle)), \tag{2.65}$$

where the $\pm$ indicates the sign of the product of all outcomes. Next, the second-level qubits are to be measured in the Z basis to detach them. Then a single qubit is left in the tree whose state depends on the product of all measurements, including the ones on the second-level qubits,

$$\alpha|+\rangle \pm \beta|-\rangle, \tag{2.66}$$

where the $\pm$ indicates the sign of the product of all outcomes. Then if that value is negative Z needs to be applied to the qubit and finally H to recover the state in the computational basis, this way one recovers the initial state that was encoded in a tree $\alpha|0\rangle + \beta|1\rangle$ in one of the first-level qubits in the tree. In this example with the [3, 2] tree code, the corrections applied throughout the different procedures have been explicitly presented. These corrections are the same for any other tree code with different branching vector.

2.5 One-way quantum repeater protocol with tree code

In this section, the one-way quantum repeater protocol with the tree code is presented. Both the one-way quantum repeater and the tree code have been previously introduced in 2.1.2 and 2.4.3 correspondingly. The explained protocol is based on the one proposed by Johannes Borregaard et al. in [15], based on photonic tree states.

One-way quantum repeaters do not require long-lived quantum memories nor two-way quantum communication, so that in comparison with the other types of quantum repeater their distribution rates are only limited by the slowest component enabling very high communication rates. As the name suggest, the repeater chain only needs communication in one direction in order to send or share a quantum state. These repeater require photonic encodings and quantum error-correcting codes to protect the information against losses and errors. Additionally, the tree code is used as the photonic encoding, a choice which relies on the loss tolerance that this encoding shows due to its highly entangled and redundant state.

The structure of an one-way quantum repeater chain with the tree code is formed by a sending station, where the quantum information is encoded and sent to the first repeater node. Then, at each of the repeater nodes a re-encoding procedure is done, until the end node of the chain is reached and the information is retrieved. All of this is presented considering that the photons are sent through a fiber and thus, are subjected to losses. Finally, an analysis of the behaviour of such repeater chain under errors is explored.

[6] On the following equation, the measured qubits are discarded from the state.

2.5.1 Encoding

At the first station, which can be called the sending station, a tree state is generated. In [15] it is claimed that a photonic tree of depth d required for the repeater station can be generated with $d-1$ memory qubits (spin qubits) and one single photon emitter per repeater station using repeated photon emissions. Moreover, one must be able to perform CPHASE gates between neighbouring spins, where the spins are in a linear configuration. This generation is done such that the root qubit of the tree is in one of the memory qubits available in the repeater and the rest of the qubits in the tree are represented by photons.

After generating the tree, quantum information needs to be encoded in it by means of a Bell state measurement as seen in 2.4.3. In particular, a message qubit, $\alpha\left|0\right\rangle_s + \beta\left|1\right\rangle_s$, is prepared in another of the memory qubits available. Then a Bell state measurement between the root qubit and the message qubit, both in memory qubits, is performed such that the tree code has been encoded and only the photons are left. Note that as said in 2.4.3, corrections must be applied depending on the outcomes of the Bell state measurement. Then the photons of the encoded tree are sent or transmitted through fiber to the next repeater station, a process that may be subjected to photon losses.

In particular, the tree code or any other loss tolerant code can fundamentally never tolerate more than 50% losses per elementary link. This corresponds to a distance of approximately 15 km for a telecom fiber, which has an attenuation length $L_0 \approx 22$ km [23]. Thus, any one-way quantum-repeater chain could have at most this spacing. However, in reality the spacing needed is probably smaller, as the code is probably not near-deterministic all the way until 50% losses.

2.5.2 Re-encoding

The goal of an intermediate repeater station in the one-way quantum repeater is to re-encode the retrieved qubit after the decoding in a new tree generated in that repeater station. This procedure is somewhat similar to the procedure of encoding the message qubit at the sending station. In broad terms, the re-encoding procedure consists in generating a new tree of photons with a spin qubit as the root qubit, followed by a Bell state measurement between the message qubit and the root qubit. So one can understand that this re-encoding "refreshes" or "repairs" the encoded tree in order to be able to go to longer distances.

As seen in the decoding procedure of the tree code in 2.4.3, the message qubit is recovered in a photonic qubit of the first level of the tree. In order to do this, all the first-level photons that are not used to retrieve the information are measured directly and indirectly in Z, thus, detaching them from the tree up to a Z correction as seen in 2.4.1. This (in)direct Z measurement is also applied to the second-level photons below the targeted first-level photon

Then in order to "refresh" the tree, the re-encoding procedure requires a Bell state measurement between one of the first-level photons of the encoded tree that arrives at the repeater station and the root qubit of a new generated tree, along with the measurement

of all the remaining qubits in the encoded tree as specified in the decoding procedure.

Note that the measurements in the decoding procedure do not need to follow any specific order due to the fact that they are all on different sub-spaces. This allows for the re-encoding procedure at each repeater station to be performed without prior knowledge of which photons are lost. Specifically, one could first attempt a Bell state measurement between one of the first-level photons and the root qubit of the new tree. While always keeping in mind that the success of this measurement depends on if the selected first-level photon is lost or not.

Therefore, the first approach that comes to one mind is a trial-and-error procedure to perform a successful Bell state measurement. But this approach may damage or perturb the root qubit of the new tree. Inevitably, every time the Bell state measurement is unsuccessful, due to a lost first-level photon, one would have to generate a new tree. A better approach would be to consider a re-encoding procedure that operates in a loss tolerant manner. This is such that if the Bell state measurement is attempted on a lost first-level photon, the measurement is aborted without perturbing the root qubit of the new tree.

To avoid having to generate a new tree-cluster state every time there are losses, in [15] the following procedure was proposed. A *heralded storage* of the message qubit in an auxiliary spin qubit is performed first. This is obtained with a spin-photon controlled-phase gate with a first-level photon of the encoded tree. Afterwards the success of the storage is determined by the detection of the photon in the X basis. Depending on the success one should proceed as follows,

- If the **storage is unsuccessful** due to photon loss, the auxiliary spin is reinitialized and a new attempt is made with another first-level photon. Also, one wants to detach the sub-tree that the lost first-level photon belonged to, following the sequence in Table 2.6. This is such that the root qubit of the new tree-cluster is unaffected.

- If the **storage is successful**, the auxiliary spin qubit can be referred to as storage qubit. Then, a deterministic Bell state measurement is performed between the storage qubit and the root qubit of the new tree. To teleport the encoded quantum information into the new tree we need to leave the tree initially received in the repeater station with no photons, this is done as specified in the decoding procedure where now the selected first-level qubit has become the storage qubit. This is done by means of (in)direct Z measurements on all first-level photons and second-level photons under the targeted first-level photon. Finally, the re-encoding procedure is complete.

The last step in the repeater station is to send the new encoded tree to the next repeater station, where again the transmission may cause losses.

2.5.3 Repetition and end node

This procedure of re-encoding and transmission is repeated until one reaches the end node of the repeater chain. There the message qubit can either be transferred to an auxiliary spin qubit as before, or directly measured without being transferred. The latter is done by performing a sequence of measurements on the photons of the encoded tree which consist on (in)direct Z measurements on all first-level photons and second-level photons under the targeted first-level photon.

Note that at all repeater stations at least two memory qubits are needed in order to perform the initial encoding and the re-encodings. Moreover, at each repeater station a tree needs to be generated a processor with $d-1$ memory qubits (spin qubits) and one single photon emitter is needed. Therefore, in order to be able to perform this repeater chain one must have two memory qubits, thus one is able to generate at least trees of depth 3.

2.5.4 Errors

The simplicity of the presented protocol for the one-way quantum repeater is nothing but an advantage. Yet one can still find downsides to it, such as the fact that this specific encoding is not a fault tolerant encoding, meaning that is not able to correct arbitrary errors. In particular, if an error happens on the qubits that take part into the re-encoding Bell state measurement it will map into an error on the new logical qubit, which is impossible to detect, track or correct. Thus, one can say that this encoding suffers from a non-fault tolerant Bell state measurement. Therefore, this can be considered to be the main bottleneck in terms of error tolerance of the one-way quantum repeater with the tree code.

However, as seen in the decoding procedure of the tree code in 2.4.3, the tree code is not fault tolerant but has a certain redundancy against errors. This comes from the fact that in the decoding procedure the Z measurements in a sub-tree can be performed indirectly and directly and the outcomes of those are correlated in such a way that they can be majority voted in the scenario where errors are considered.

In this particular encoding one can think of the possible error sources, which include but are not limited to the tree generation, which includes the emission of photons and the controlled-phase gates between neighbouring qubits, the measurements performed on the photons of the tree in order to decode it, the Bell state measurement itself and the transmission of photons through fibers.

Moreover, from the tree generation procedure one can see that the errors between photons in different levels in a same sub-tree should be correlated from a realistic point of view. Specifically, the correlations in the tree code are created by the CPHASE gates between a qubit in a certain level with its leaves, which are the qubits attached to it in a level below, in such a way that if an error occurs in the higher-level qubit before this operation, correlated errors are created in the leaves. This is true for trees and sub-trees.

In the numerical optimization in [15] a single-qubit depolarizing noise, as defined in 2.2.1, is considered. Moreover, they looked into how single qubit depolarizing errors add to a total re-encoding error, ϵ_r. Additionally, they considered that all the qubits that are transmitted are subject to the same depolarizing rate ϵ. For not very high ϵ, these errors can be mostly corrected making use of the redundancy of the tree code by majority voting. However, if an error happens on the first-level qubit in the sub-tree were the heralded storage succeeds there is no way of detecting or correcting it. The lower-level qubits in that sub-tree are protected against errors if the depth is larger than 2, due to the ability of performing indirect Z measurements. In general, by making the tree deeper the redundancy of the measurements in the decoding increases, but one also has to keep in mind that if many qubits are lost, there may not be much redundancy left in the measurement results, making them more prone to errors.

CHAPTER 3

Code concatenation in the one-way quantum repeater

Code concatenation has been presented as a tool used to make larger codes out of small codes as it has been presented in 2.3.3 [38, 43]. Briefly, the idea behind code concatenation is "double encoding" mainly because the physical qubits that form the logical qubit of the outer code are in turn the logical qubits of the inner code. Then the physical qubits of the inner code correspond to the "real" physical qubits which are subjected to this double encoding.

As stated in 2.5.4, the main bottleneck in terms of error tolerance of the one-way quantum repeater based on the tree code is that the re-encoding at each repeater station relies on a single message qubit which is non-error-proof. This entails that at each repeater station we must re-encode the quantum information in such a way that we are left with a single message qubit which is encoded in a new tree-cluster state by means of a Bell state measurement. If any error would happen in that qubit there is no way to either detect nor correct that error, making the repeater chain very susceptible to errors.

Therefore, we propose to take advantage of the idea of code concatenation to avoid depending on a single qubit in the re-encoding. The goal is to keep using the tree code as inner code and use a stabilizer code as outer code.

In this chapter, first a discussion about the choice of the codes used in the encoding is presented, followed by a preliminary protocol for the code concatenated one-way quantum repeater. Afterwards, the method to measure stabilizers of the outer code on the inner code is explained and the necessary characteristics of a suitable specific outer code are discussed. Finally, two protocols for the code concatenated one-way quantum repeater using the 5-qubit code as outer code are presented in detail.

3.1 Choice of concatenated codes

In this section we want to discuss the main reasons of choosing code concatenation of a stabilizer code, as an outer code, and the tree code, as inner code.

First of all, the tree code is used as the inner code as it presents an important loss tolerance due to the redundancy inherent in its structure and inner entanglement [49]. In this code, a single qubit is encoded in a tree, the latter being a simple graph state that can be characterized by a branching vector, which denotes the connectivity of the different levels of the tree. The properties of graph states, tree states and the tree code have been studied in Section 2.4.

Additionally, the choice of using a stabilizer code as the ones presented in 2.3.2, as outer code is based on the fact that we want to perform stabilizer measurements in the repeater chain, so that the errors that affect the outer code can be corrected. In order to perform these stabilizer measurements on the outer code, the qubits in the inner code, the tree code, will be used and discarded after each measurement. Moreover, this is also motivated by the fact that the photons that form the trees have been discarded up to now in the measurements to decode the tree. The usefulness of these photons can be further expanded for stabilizer measurements. In this particular encoding the qubits in the trees will be divided between the ones used for decoding and the ones used for stabilizer measurements.

3.2 Preliminary protocol for the code concatenated one-way quantum repeater

In this section we present a *preliminary* protocol on how the one-way quantum repeater works with a code concatenated encoding. Importantly, this protocol is introduced as a starting point for the later protocols. The main difference is that the present protocol does not consider the stabilizer measurements and the protocols that will be presented next will consider them. The reason why these measurements have not yet been added into the protocol is because before making any specific choices we need to get more detail on how they work and what they entail.

The preliminary protocol starts at the initial sending station, where a "double" encoding needs to be performed. First the message qubit is encoded in a $[[n, 1, d]]$ stabilizer code and then each of the n qubits resulting of the first encoding are encoded into photonic trees of a certain branching vector, which does not necessarily need to be the same for all n trees. Then those n trees are sent to the first repeater station through a fiber. In the repeater station each tree is re-encoded by means of a Bell state measurement between one first-level photon and the root qubit of a new tree. Therefore, at each repeater station n Bell state measurements are performed. Then, the re-encoded n trees are sent to the next repeater station and so on until the end node of the repeater chain is reached. At this last step, the n trees are decoded returning n qubits which are next decoded from the stabilizer code, so that a single qubit can be retrieved.

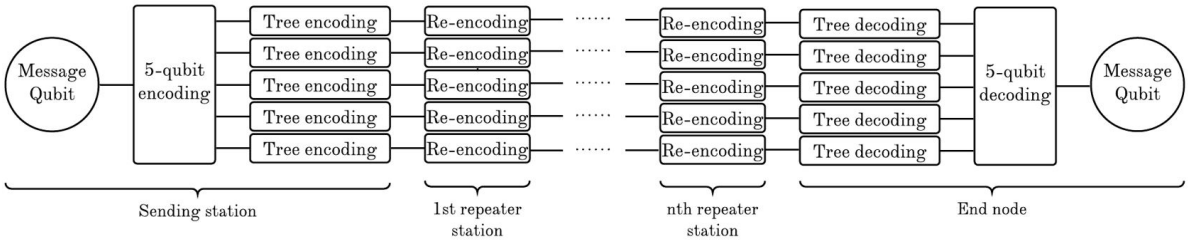


Figure 3.1: Code concatenated one-way quantum repeater with n repeater stations, using the 5-qubit code, $[[5, 1, 3]]$, as outer code and the tree code as inner code. The five edges connecting the different repeater stations denote the five trees being transmitted.

In Figure 3.1 this protocol is sketched for the 5-qubit code as outer code, so that one can grasp the main scheme of the one-way quantum repeater with code concatenation. Previously, the protocol for the one-way quantum repeater based on an all-photonic tree encoding has been presented in Section 2.5, where we have seen that this protocol requires a processor with $d-1$ spin systems and one quantum emitter per repeater station for a tree code of branching vector $\bar{t} = [b_0, \ldots, b_d]$. Following this structure, at each repeater station of the code concatenated one-way quantum repeater n processors with $d_i - 1$ spin systems and one quantum emitter each are needed, where d_i with $i \in \{1, \ldots, n\}$ denotes the depth of the tree code where qubit i from the outer code is encoded in. Each of this processors is required to perform the re-encoding procedure for every tree of depth d_i per repeater station. Thus, the encoding into trees at the sending station and the re-encoding of each of the n trees at a repeater station is performed completely analogously to the tree-code-based re-encoding.

Next, in order to include stabilizer measurements of the outer code in our repeater chain we describe the procedure to perform such measurements in code concatenation.

3.3 Stabilizer measurements in trees

The goal of performing stabilizer measurements in code concatenation is to be able to detect and correct errors on the outer code with their outcomes. Nevertheless, the focus in this section is to understand how the stabilizers are measured instead of the correction of errors. The latter is specified in the last two sections of this chapter.

In order to do this, we need to learn how a stabilizer operator on the outer code can be implemented on the inner code. To do so, let us consider again the example of Shor's code as concatenated code, as seen in 2.3.3, which uses the phase flip code, seen in 2.3.2, as outer code and the bit flip code, seen in 2.3.2, as inner code. The stabilizers of the phase flip code are, $X_a X_b$ and $X_b X_c$, where here a, b and c denote the three qubits that form the code. Then each of those qubits is encoded into a 3-qubit bit flip code, such that a is encoded into three qubits labeled 1, 2, and 3, b in qubits labeled 4, 5 and 6 and c in qubits labeled 7, 8 and 9. All these nine qubits form Shor's code. To "translate" $\overline{X}_a \overline{X}_b$ and $\overline{X}_b \overline{X}_c$, where the line over the operator denotes that is the operator that corresponds to the outer code in code concatenation, into the bit flip code or to know which form

these stabilizers take once the two codes have been concatenated, one needs to look at the logical operators of the inner code. Logical X on the bit flip code is $X_1X_2X_3$, so that $\overline{X}_a\overline{X}_b = X_1X_2X_3X_4X_5X_6$ and $\overline{X}_b\overline{X}_c = X_4X_5X_6X_7X_8X_9$.

Following the same idea, one can "translate" the stabilizer operators of a given outer code to the ones of the inner code, now represented by a tree code. For this reason it is important to consider the logical operators of the tree code, which have been previously introduced in 2.4.3.

In what follows the different kinds of stabilizers measurements are presented, depending on which Pauli operations compose the operators. First stabilizers with only Pauli-Z are considered. Later stabilizers that only include Pauli-X are introduced. Finally, codes with both kinds of stabilizers and "mixed" stabilizers are presented. Note that in this section neither losses nor errors in the qubits in the trees are considered.

3.3.1 Z...Z stabilizer measurement

To begin with, we will consider as outer codes n-qubit stabilizer codes whose generators, as defined 2.3.2, are only formed by Pauli-Zs and identities. Now, the bit flip code, as a simple example of this kind of codes will be examined. This code, as presented in 2.3.2, is composed by three qubits, which each of them will be encoded in a tree of a certain branching vector $\bar{t}_l$, $l \in \{1,2,3\}$. Now, the outer code stabilizers are $\overline{Z}_1\overline{Z}_2$ and $\overline{Z}_2\overline{Z}_3$, where $\overline{Z}_l$ corresponds to the logical Pauli-Z in tree l, which was presented in equation 2.51, such that,

$$\overline{Z}_m\overline{Z}_n = \left(X_i^m \prod_{j=1}^{b_1^m} Z_{ij}^m\right) \otimes \left(X_{i'}^n \prod_{j'=1}^{b_1^n} Z_{i'j'}^n\right) \tag{3.1}$$

for $m \neq n$ and where the super-indices indicate the tree where the operation needs to be performed and the branching vectors are $\bar{t}_l = [b_0^l, b_1^l, \ldots, b_d^l]$, so that $i \in \{1, \ldots, b_0^m\}$ and $i' \in \{1, \ldots, b_0^n\}$. In general, an operator O_i^l stands for an operator O applied on qubit i, which denotes a first-level qubit in tree l. Similarly, O_{ij}^l stands for an operator O applied on qubit ij in tree l, which denotes the leaf or second-level qubit j of the first-level qubit i. The procedure for performing a measurement of these stabilizers goes as follows:

1. At first, two first-level qubits from the two involved trees, qubit i in tree m and qubit i' in qubit n, are picked. Then a non-destructive stabilizer measurement is performed between those two qubits, $X_i^m X_{i'}^n$ and the outcome of it is saved.

2. After this, the second-level qubits are measured in Z in both sub-trees from the two trees. The outcomes of these measurements are saved, and the product of them with the outcome of the previous measurement gives the result of the $\overline{Z}_m\overline{Z}_n$ stabilizer.

3. Moreover, the outcomes of these Z measurements will affect the total state as seen previously in 2.4.1. For example, the product of these measurements in tree m is -1, such that a correction needs to be applied. The correction can be done on the first-level qubit picked for the stabilizer measurement in tree m. Same procedure

is done for the other tree. These Z measurements have an additional role, they are also used to remove the qubits from the tree, as seen in 2.4.1.

4. If the depth of the tree is larger than 2, the qubits on the third level and below are also used to measure an indirect Z on the second-level qubits on them, as described in 2.4.2. This way the Z outcomes on the second-level are protected against losses, due to the redundancy of the tree state, and errors, due to the fact that a majority voting between the direct and indirect Z measurements can be performed.

5. Lastly, the used first-level qubits are detached from the tree by measuring them in Z and later on applying the corrections due to the outcomes at the corresponding tree.

Here an example involving only two trees has been presented. For a stabilizer of Z on n qubits, the step 1 would be a $X \dots X$ stabilizer measurement between all the n first-level qubits. The other steps are done in the same way as in the aforementioned method but for all the n sub-trees. From this first approach to the stabilizer measurements in trees one can see that in a repeater chain, these should take place at the repeater stations before each of the trees are decoded for the teleportation into a new tree, since we first want to use the photons to potentially measure a stabilizer. Otherwise, these photons would just be measured out in the decoding.

3.3.2 X...X stabilizer measurement

The reason why the stabilizers containing X are treated differently lies on the differences between the logical operations on a tree, as seen in 2.4.3. Their main difference is their cost, where the cost of a logical operation in this text refers to the number of physical operations that need to be applied in order to perform a certain logical operation. In particular, to apply a logical Pauli-X in a tree at least one operation on every sub-tree needs to be applied. In contrast, to apply logical Pauli-Z in a tree a set of operations, which at least consists of three of them, needs to be applied on a single sub-tree. Consider now the measurement procedure for a stabilizer with Zs, if one wants to apply a similar procedure for Xs, then at least one qubit per sub-tree per tree will be measured. This is inconvenient, since a lot of qubits would be involved in the stabilizer measurement making it more difficult to succeed against losses. Hence, here the reason why stabilizers containing Xs need to be treated differently.

Consider as an outer code a n-qubit stabilizer code whose generators, as defined 2.3.2, are only formed by Pauli-Xs and identities. A simple example of such outer code would be the phase flip code, presented in 2.3.2. This outer code is composed by three qubits where each of them will be encoded in a tree of a certain branching vector $\bar{t}_l$, $l \in \{1, 2, 3\}$. The stabilizers are $\overline{X}_1\overline{X}_2$ and $\overline{X}_2\overline{X}_3$, where $\overline{X}_l$ corresponds to the logical Pauli-X in tree l, which is presented in equation 2.49, such that,

$$\overline{X}_m\overline{X}_n = \left(\prod_{i=1}^{b_0^m} Z_i^m\right) \otimes \left(\prod_{i'=1}^{b_0^n} Z_{i'}^n\right) \tag{3.2}$$

where the super-indices indicate the tree where the operation needs to be performed, and the branching vectors are $\bar{t}_l = [b_0^l, b_1^l, \ldots, b_d^l]$. Note that performing all these Z measurements is impossible because all first-level qubits are being used. Importantly, these measurements can also be performed indirectly on the qubits on the lower levels. Nevertheless, this still does not prevent us from performing at least one operation per sub-tree in each tree, thus having a higher cost than the $Z \ldots Z$ stabilizers. Additionally, the probability of failure of performing this operation due to loss is very high, since a lot of qubits are involved. To overcome this potential problem the following procedure has been developed.

Consider a three-qubit state to be the state of the encoded phase flip code $|\Psi\rangle$, with stabilizers $g_1 = X_1X_2$ and $g_2 = X_2X_3$. Before encoding these three qubits into trees, we can bring those qubits to the X basis by applying a Hadamard to them, $|\Psi'\rangle = U|\Psi\rangle$ where $U = H_1H_2H_3$. By doing this the stabilizers are also transformed following $g_i' = Ug_iU^\dagger$, such that $g_1' = Z_1Z_2$ and $g_2' = Z_2Z_3$, this property has already been studied in 2.3.2. Note that the transformed stabilizers are the same as the ones for the bit flip code. The next step is to encode these three qubits into trees and send them to the next repeater station. The rest of the method follows along the lines of the ZZ stabilizer measurement method. Once the stabilizers are measured, and the trees are decoded, thus having again three qubits, these are returned to the computational basis making use of Hadamard gates on all qubits, applying the same $U = H_1H_2H_3$.

In conclusion, when the stabilizers of the outer code are composed only of Pauli-X, before encoding the outer code qubits into trees, H gates are applied to all of them, then stabilizers are measured following the procedure for a $Z \ldots Z$ stabilizer. Finally after decoding the trees, H gates are again applied to the qubits of the outer code.

3.3.3 General X and Z stabilizer measurement

Now that stabilizer codes with stabilizers containing only Z or only X have been discussed, the case where stabilizers are formed by both X and Z can be considered. In the examples below no losses nor errors are considered and also a sufficiently large branching vector is assumed in order to be able to perform all the stabilizer measurements.

First, consider the 9-qubit Shor's code, that has eight generators as presented in Table 2.3. There are six of them which are only composed by Zs and the other two are composed by Xs. One can see that to be able to measure all these stabilizers at least two repeater stations are needed. Let us present a detailed structure on when and how this stabilizer measurements would take place in a repeater chain.

- **Sending station:** First, at the sending station a single qubit is encoded into 9 qubits of the Shor's code and those are encoded into trees. These trees are sent to the first repeater station.
- **First repeater station:** At the first repeater station, before the decoding in the re-encoding procedure, the stabilizer measurements are to be performed. On a first instance the heralded storage of a first-level qubit is performed to ensure the

possibility of a later successful re-encoding and then the stabilizers are measured. In this repeater station, only the six Z-type stabilizers can be measured, due to the fact that prior to the encoding the trees no Hadamard gates were applied. After performing the stabilizer measurements and decoding the trees, a H gate is applied to all the 9 qubits, which have been heralded-stored. Then those qubits are encoded into trees and sent to the next repeater station.

- **Second repeater station:** As in the first repeater station, before the decoding, the stabilizer measurements are to be performed. In this case, only the two X-type stabilizers can be measured, due to the fact that before encoding the trees Hadamard gates where applied to all 9 qubits. After performing the stabilizer measurements and decoding the trees, a H gate is applied to all 9 qubits. Then those qubits are encoded into trees and sent to the next repeater station.
- **Other repeater stations:** Now all the following pairs of repeater stations can follow the same structure as the first and second repeater stations, until the last one that will send the trees to the end node.
- **End node:** Finally, both the tree code and the 9-qubit code are decoded in order to recover a single qubit.

Here one can see clearly the reason why at least two repeater stations are needed. At least one where trees are sent into with the computational basis and another where the trees are sent into with the X basis, so all the sets of stabilizers can be measured.

Moreover, stabilizers that are composed both by Xs and Zs can also be measured using this basis switching. Specifically, one only switches the basis of the qubits that have an X on the stabilizer. Consider a stabilizer X_1Z_2, then only qubit 1 will be switched to the X basis, so that needs the unitary $U = H_1$. To portray this property in a stabilizer code with "mixed" stabilizers we present an example with the 5-qubit code as an outer code. This code has four stabilizers as presented in Table 2.4, which are formed by X and Z, but none of them have a similar structure, in such a way that it is impossible to create subsets of them based on the basis that each qubit needs. To explicitly state this take $g_1 = X_1Z_2Z_4X_5$, before encoding the five qubits into trees to measure this stabilizer at the next repeater station, one needs to apply $U_1 = H_1H_5$, but none of the other generators need the same unitary operation to be in the right basis to measure the stabilizer. The other generators are $g_2 = X_1X_2Z_3Z_5$, $g_3 = Z_1X_2X_3Z_4$ and $g_4 = Z_2X_3X_4Z_5$, and the respective unitary operations that they need are $U_2 = H_1H_2$, $U_3 = H_2H_3$ and $U_4 = H_3H_4$, such that $U_i \neq U_j$ if $i \neq j$. Therefore, in the case of the 5-qubit code, a repeater chain that is able to measure all stabilizers needs at least four repeater stations. If we consider a repeater chain with five links, the procedure to measure all stabilizers of the 5-qubit code would be,

- **Sending station:** First, at the sending station a single qubit is encoded into the 5-qubit code, $U_1 = H_1H_5$ is applied to them and they are encoded into trees. These trees are sent to the first repeater station.
- **First repeater station:** At the first repeater station, before the decoding in the re-encoding procedure, the stabilizer measurements are to be performed. On

a first instance the heralded storage of a first-level qubit is performed to ensure the possibility of a later successful re-encoding, then the stabilizers are measured. In this repeater station, only the g_1-type stabilizers can be measured, due to the fact that before encoding the trees U_1 was applied. After performing the stabilizer measurement and decoding the trees, $U_1U_2 = (H_1H_5)(H_1H_2) = H_2H_5$ is applied to all the 5 qubits, which have been heralded-stored. U_1 is applied to return qubits 1 and 5 to the computational basis and U_2 to bring qubits 1 and 2 to the X basis to perform the next stabilizer. Then those qubits are encoded into trees and sent to the next repeater station.

- **Second repeater station:** As in the first repeater station, before the decoding, the stabilizer measurements are to be performed. In this case, only the g_2-type stabilizers can be measured, due to the fact that before encoding the trees U_2. After performing the stabilizer measurements and decoding the trees, $U_2U_3 = (H_1H_2)(H_2H_3) = H_1H_3$ is applied. Then those qubits are encoded into trees and sent to the next repeater station.

- **Third repeater station:** As in the previous repeater stations, before the decoding, the stabilizer measurements are to be performed. In this case, only the g_3-type stabilizers can be measured. After performing the stabilizer measurements and decoding the trees, $U_3U_4 = (H_2H_3)(H_3H_4) = H_2H_4$ is applied. Then those qubits are encoded into trees and sent to the next repeater station.

- **Fourth repeater station:** As previously, before the decoding, the stabilizer measurements are to be performed. In this case, only the g_4-type stabilizers can be measured. After performing the stabilizer measurements and decoding the trees, $U_4 = H_3H_4$ is applied in order to bring all the state to the computational basis. Then those qubits are encoded into trees and sent to the end node.

- **End node:** Finally, both the tree code and the 5-qubit code are decoded in order to recover a single qubit.

This scheme clarifies the fact that for the 5-qubit code at least four repeater stations are needed. In general, in this section we have been how different kinds of stabilizers are measured, such that we are able to measure Z-type, X-type and "mixed"-type stabilizers.

3.4 Possible outer codes

After seeing the main idea behind the code concatenation in the one-way quantum repeater many questions arise, for example, what is the trade-off of qubits in the tree used to decode and used for the stabilizer measurement, what size should the branching vector be, how does these encoding and stabilizer measurements behave and perform under losses and errors and many more. To answer these questions and later on have a more determined protocol, first we need to fix one of the many variables that this encoding presents, the outer code, which needs to be a stabilizer code, some examples can be seen in 2.3.2.

From the stabilizer measurement procedure we can draw two conclusions on which conditions the stabilizer code should fulfil. First, the stabilizers to be measured in one repeater station need to involve each tree a similar amount of times. The reason lies in the cost of the stabilizers, which is a sub-tree per involved tree. If in a single repeater station many stabilizers are measured and those involve a certain tree considerably more times than the other trees, then this tree needs to have more sub-trees than the other ones to have a similar success probability when dealing with losses. Therefore, one would not have a uniform branching vector for all the trees and for all the repeater stations or sets of stabilizers. Moreover, if the branching vectors are different in depth then the number of memory qubits needed in a repeater station would also vary between trees. Thus, having different branching vectors for different trees would would make the analysis of this encoding more challenging. Since in this thesis we aim for a simpler investigation, a stabilizer code that allows for a uniform branching vector will be considered. Second, the stabilizers need to have small weight. The larger the weight of a stabilizer is, the larger the number of first-level qubits required to perform said stabilizer. This fact may come as a challenge when dealing with losses, as the probability that one of the selected first-level qubits is lost increases and thus the probability that the stabilizer measurement is not performed successfully also increases. So, one should look for stabilizer codes with stabilizers with the smallest weight possible.

Moreover, it makes sense to look for a stabilizer code able to at least correct arbitrary single-qubit errors, so that it is a code of distance $d = 3$, as seen in 2.3.2. Additionally, it is also important to consider outer codes that will not imply a huge number of repeater stations to measure a single set of stabilizers. Finally, in this thesis we aim for an initial simpler analysis of this code-concatenated encoding, therefore, small stabilizer codes are considered.

From all these constraints we believe that the 5-qubit code [42] is the best option to explore in this thesis. Following, a detailed protocol considering losses and errors within the code concatenated one-way quantum repeater with the 5-qubit code. In particular, at the sending station, to encode the chosen outer code, the 5-qubit code, one can follow the circuit drawn in Figure 3.2. The same circuit should be followed in order to decode the outer code at the end node, but in the opposite direction, from right to left.

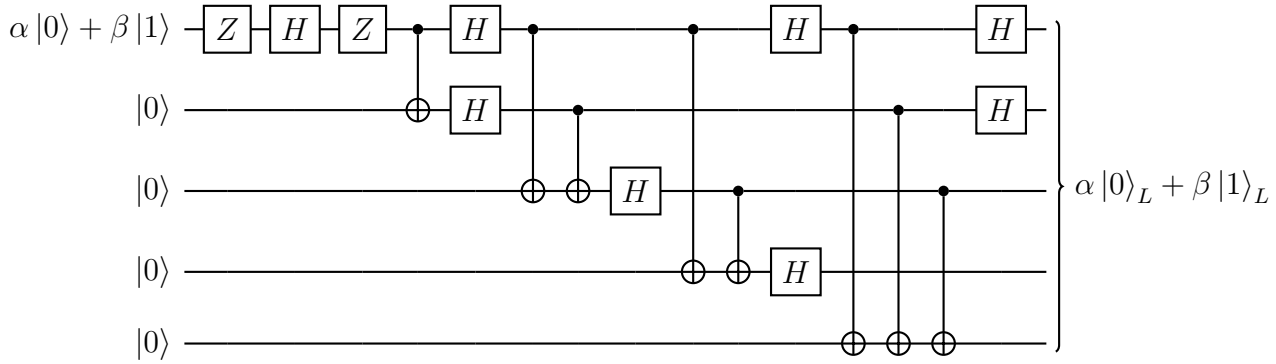


Figure 3.2: Quantum circuit for the encoding procedure of the 5-qubit code, where the state $\alpha|0\rangle + \beta|1\rangle$ represents the state of the message qubit being encoded. The final logical state corresponds to $\alpha|0\rangle_L + \beta|1\rangle_L$, where the sub-index L indicates that is the logical state.

3.5 Protocol 1 for the code concatenated one-way quantum repeater

In this section, we want to give the necessary tools to answer many questions regarding the one-way quantum repeater with code concatenation of a 5-qubit code, as the outer code, and a tree code, as the inner code. Particularly, the main focus is on the proposed loss and noise channels that the photons in the trees will be subject to when they are sent to a repeater station and how both losses and errors are analyzed and treated in this particular encoding. Although this protocol is presented for the 5-qubit code as outer code it can be easily generalized for any other outer code.

The approaches considered in this section define the *protocol 1* of the one-way quantum repeater with code concatenation with the 5-qubit code. This protocol uses as a basis the preliminary protocol presented in 3.2 and adds the constraints and methods on how and when to perform stabilizer measurements, how to deal with losses and errors in the photons in the trees and how and when to perform corrections on the outer code due to the stabilizer outcomes. Finally the proposed protocol in this section is a stepping stone for the next and final protocol, which will be presented in the next section.

3.5.1 Protocol 1: Structure of the repeater chain

Previously we have seen that for the 5-qubit code each stabilizer needs a different repeater station to be measured. Therefore, a repeater chain of at least four links is necessary in order to measure all the stabilizers. However, there are three questions that arise from this. First, how many times will a stabilizer be measured on a given repeater station. Second, which should be the frequency with which the stabilizers will be measured in the repeater chain stabilizer. Third, when are the corrections on the outer code performed. In this protocol the answers to these questions are presented and they are stated as follows:

> **Stabilizers in a repeater station:** In protocol 1 we will consider that at each repeater station one stabilizer is measured. Therefore, to have at least a set of stabilizer measurements outcomes, the chain needs to be composed of at least four links.

> **Frequency of stabilizers in a repeater chain:** In protocol 1 we will consider that there are n initial links in which no stabilizers are measured. Therefore, these links have the structure described in the preliminary protocol in 3.2, we will refer to these as the *free evolution links*. After these, there are four links in which the stabilizers are measured, which will be referred to as the *stabilizer links*. Now we have a set of $n + 4$ links in the repeater chain. Finally, a full repeater chain will be formed by k sets, such that the total number of links is $k(n + 4)$.

> **Corrections on the 5-qubit code:** In protocol 1 we will consider that after the four stabilizer links in each of the sets of $n+4$ links the corrections due to the syndrome outcomes on the 5-qubit code are applied. Specifically, in each of the last links of each set after measuring the stabilizer and decoding the trees, when there are five qubits is the moment where the corrections will be applied.

These answers are motivated and explained in the following sections. The first one in 3.5.2 and the other two in 3.5.3.

3.5.2 Protocol 1: Losses

First we want to shed some light upon how this protocol operates against losses. We will consider that all the photons in the trees are subject to an erasure channel, defined in 2.2.1, which considers the loss of a qubit with probability μ when they are sent to a next station.

Losses in free evolution links

In general one can view a free evolution link as five independent simultaneous links of the one-way quantum repeater based on the tree code, as basically the re-encoding procedure is done five times instead of one. Then in terms of losses the five trees and re-encodings behave just as in the protocol for the one-way quantum repeater with the tree code presented in 2.5. To be able to study the loss tolerance of these links, an analytical expression for the success probability of a free evolution link of the 5-qubit code concatenated with a tree code with branching vector $\bar{t} = [b_0, b_1, \ldots, b_d]$ for all trees can be determined. To obtain it, we depart from an expression for the tree code that can be found in (1) in [49], as well as (5) in [15]. From [49], the expression for the probability of having a successful indirect Z measurement on a qubit in the kth level is defined for $k \leq d$, as

$$R_k = 1 - \left(1 - (1-\mu)(1-\mu+\mu R_{k+2})^{b_{k+1}}\right)^{b_k} \tag{3.3}$$

where $R_{d+1} = 0$, $b_{d+1} = 0$. Then, the probability of a successful Z measurement in a qubit in the kth level directly is $1-\mu$ and indirectly is μR_k. Next the probability for a tree to have a successful heralded storage first-level qubit, the second-level qubits of that sub-tree (in)directly measured in Z and the rest of first-level qubits (in)directly measured in Z is also defined in [49] and [15], and is,

$$\eta_1 = \left((1-\mu+\mu R_1)^{b_0} - (\mu R_1)^{b_0}\right)(1-\mu+\mu R_2)^{b_1}, \tag{3.4}$$

where the first term, $(1-\mu+\mu R_1)^{b_0} - (\mu R_1)^{b_0}$, is the probability of having one first-level qubit and the other ones measured in Z and the second term is the probability of measuring in Z all the second-level qubits of one sub-tree. Given that a free evolution link is five concurrent tree-code-based links the success probability for one free evolution link is η_1^5 and for n links is $(\eta_1^5)^n$.

Losses in stabilizer links

In a stabilizer measurement, the Z measurements on the second-level qubits that are performed directly and indirectly are rather protected against losses. However, for the $X \dots X$ stabilizer measurement between first-level qubits, if any of the involved photons is lost, the measurement becomes impossible. Thus, the losses affect the probability of success of measuring the stabilizers.

> **Stabilizers in a repeater station:** In protocol 1 we consider that at each repeater station one stabilizer is measured. If this measurement is unsuccessful due to losses in a repeater chain the protocol is aborted making the whole repeater chain fail. The chain needs to be composed of at least four links, such that at least a set of stabilizer measurements can be performed.

The choice of measuring one stabilizer per repeater station is motivated by the fact that the success probability of a stabilizer link is lower as more stabilizers are considered to be measured. To overcome the losses and be able to perform successfully a stabilizer measurement per repeater station, this protocol assumes a *trial-and-error* approach. This is such that if the $X_i^m X_{i'}^n$ stabilizer measurement fails due to losses, the sub-trees involved in those measurements are detached from the tree as detailed in Table 2.6 and the corresponding corrections are applied. Then one tries again selecting other first-level qubits. The proposed approach has limited attempts, since the number of first-level qubits in a tree is finite and defined by b_0 in the branching vector. Additionally, as we said before, we will consider that the first-level qubit that is initially used to perform a heralded storage for the re-encoding procedure can not be used to perform the stabilizer.

> **Losses in stabilizers:** In protocol 1 the considered approach to deal with losses in terms of measuring stabilizers is the trial-and-error approach. Such that the used unsuccessfully sub-trees are detached from the trees. This is limited to the number of available first-level qubits.

In terms of resources, this protocol needs five processors with $d-1$ spin qubits and a quantum emitter per repeater station just as in the preliminary protocol. Thus, the stabilizer measurements in protocol 1 do not require additional resources.

3.5.3 Protocol 1: Errors

Moreover, in this protocol we will also consider that the photons in the trees are exposed when they are sent to a depolarizing channel, defined in 2.2.1, with a probability ϵ. Again, the effect of these errors can be mostly overcome by applying the majority voting in the redundant measurements.

> **Majority voting:** In protocol 1 when doing a majority voting for a measurement, if no conclusion can be retrieved, that specific measurement will be considered as failed.

Errors in free evolution links

As seen in the preliminary protocol a free evolution link has the same behaviour than five simultaneous tree-based links, defined in 2.5. Therefore, the errors are corrected exclusively by majority voting at the re-encoding procedure. Moreover, each of the five trees may have an error on the first-level qubit used for the re-encoding that leads to a logical error on the next new tree, which will cause an error on the outer code. For these links, there is neither detection nor correction of errors on the outer code but instead there is the possibility of leading to an error on the outer code. Therefore, one can say that the free evolution links let the error on the outer code build up and that is why we have given them that particular name.

Errors in stabilizer links

In the case of the stabilizer links the errors are corrected by majority voting as much as possible. However, in these links there is the addition of stabilizer measurements, which are made to detect and correct errors on the outer code and not for errors in the inner code. Nevertheless, the last two are connected since a certain sequence on certain qubits in a tree leads to an error on the logical state of the tree. In this case these depolarizing errors can trigger a faulty stabilizer outcome. To clarify this statement, let us present an example, consider the stabilizer $g_1 = X_1 Z_2 Z_4 X_5$ and a tree of depth 2, then the stabilizer measurement after applying $U_1 = H_1 H_5$ and encoding into trees is $\overline{g}_1 = \overline{Z}_1 \overline{Z}_2 \overline{Z}_4 \overline{Z}_5$, so that

$$\overline{g}_1 = \left(X_i^1 \prod_{j=1}^{b_1} Z_{ij}^1 \right) \otimes \left(X_{i'}^2 \prod_{j=1}^{b_1} Z_{i'j}^2 \right) \otimes \left(X_{i''}^4 \prod_{j=1}^{b_1} Z_{i''j}^4 \right) \otimes \left(X_{i'''}^5 \prod_{j=1}^{b_1} Z_{i'''j}^5 \right). \tag{3.5}$$

Then, consider that qubit $i'j$ in tree 2 has an X error, this error will give a wrong outcome when measuring $Z_{i'j}^2$ and thus, the outcome of $\overline{g}_1$ will be -1. This outcome denotes an error on the outer code, when in reality is just an error in one of the trees that has no logical error in the state of the tree. Importantly, this error in the case of the unconcatenated one-way quantum repeater would be detected and corrected thanks to the majority voting.

Of course is important to note that an error like the one in the example can be avoided by adding a third level so that when measuring $Z_{i'j}^2$ the qubits below can also be measured in X and then apply a majority vote and if there is no final agreement the measurement is aborted. In the case of having an error on one of the first-level qubits in the stabilizer measurement there is no solution.

The reason why there are some initial n free evolution stations is precisely to build up an initial error on the outer code which can probably be corrected with the stabilizer measurements in the four stabilizer links. In the case where $n = 0$, the errors on the photons in the trees need to correspond exactly to the combinations of operations on the trees to give a logical error on the outer code otherwise they could either trigger a faulty stabilizer outcome or be corrected by majority voting.

Following this train of thought, the most logical option to perform the corrections due to the stabilizer outcomes is to correct right after the set of stabilizers has been successfully measured so a syndrome can be determined. This way we can correct the errors that have been probably built up in the n free evolution links, with the restriction that the 5-qubit code is only able to correct arbitrary single-qubit error, so only a single error on one of the five qubits. The specific correction to be applied after a set of stabilizer is measured is presented in Table 3.1.

g_1	g_2	g_3	g_4	Syndrome	Correction
+	+	+	+	No error	None
-	+	+	+	Z error on qubit 5	Z_5
+	-	+	+	X error on qubit 3	X_3
+	+	-	+	X error on qubit 1	X_1
+	+	+	-	Z error on qubit 4	Z_4
-	-	+	+	Z error on qubit 1	Z_1
-	+	-	+	X error on qubit 4	X_4
-	+	+	-	X error on qubit 2	X_2
+	-	-	+	Z error on qubit 2	Z_2
+	-	+	-	X error on qubit 5	X_5
+	+	-	-	Z error on qubit 3	Z_3
-	-	-	+	Y error on qubit 1	Y_1
-	-	+	-	Y error on qubit 5	Y_5
-	+	-	-	Y error on qubit 4	Y_4
+	-	-	-	Y error on qubit 3	Y_3
-	-	-	-	Y error on qubit 2	Y_2

Table 3.1: Error syndrome and correction table for the 5-qubit code, where the generators are $g_1 = X_1Z_2Z_4X_5$, $g_2 = X_1X_2Z_3Z_5$, $g_3 = Z_1X_2X_3Z_4$ and $g_4 = Z_2X_3X_4Z_5$, as in Table 2.4. The $\pm$ signs represent the ± 1 outcomes of the stabilizer measurements.

3.6 Protocol 2 for the code concatenated one-way quantum repeater

In this section, we develop an alternative protocol for the code concatenated one-way quantum repeater with the 5-qubit code and the tree code, which will be called *protocol 2*. Although this is presented for the 5-qubit code as outer code it can be easily generalized for any other outer code. Protocol 2 uses as a basis protocol 1 and improves it in terms of the error-less success probability. Therefore, for protocol 2 the structure and the constraints and considerations for the errors and corrections due to syndrome measurements are the same than for protocol 1, presented in 3.5.1 and 3.5.3. Therefore, the answer to the three questions asked in protocol 1 are the same. Here we highlight the main features shared between the two protocols:

Stabilizers in a repeater station: In protocol 1 and 2 we will consider that at each repeater station one stabilizer is measured. Therefore, to have at least a set of stabilizer measurements outcomes, the chain needs to be composed of at least four links.

Frequency of stabilizers in a repeater chain: In protocol 1 and 2 we will consider that there are n initial links in which no stabilizers are measured. Therefore, these links have the structure described in the preliminary protocol in 3.2, we will refer to these as the *free evolution links*. After these, there are four links in which the stabilizers are measured, which will be referred to as the *stabilizer links*. Now we have a set of $n + 4$ links in the repeater chain. Finally, a full repeater chain will be formed by k sets, such that the total number of links is $k(n + 4)$.

Majority voting: In protocol 1 and 2 when doing a majority voting for a measurement, if no conclusion can be retrieved, that specific measurement will be considered as failed.

Corrections on the 5-qubit code: In protocol 1 and 2 we will consider that after the four stabilizer links in each of the sets of $n + 4$ links the corrections due to the syndrome outcomes on the 5-qubit code are applied. Specifically, in each of the last links of each set after measuring the stabilizer and decoding the trees, when there are five qubits is the moment where the corrections will be applied.

These statements answers were motivated and explained in previous sections in protocol 1. The first one in 3.5.2 but is also motivated in this protocol with the same argument in 3.6.1 and the other three in 3.5.3. As we will see in this section the only difference between the two protocols is the approach that protocol 2 has for improving the success probability in the stabilizer links.

3.6.1 Protocol 2: Losses

As in the previous protocol, we will consider that all the qubits in the trees are subject to an erasure channel with probability μ when they are sent to a next station. As explained in the previous section, these losses will affect both the success probability of the free evolution links and the stabilizer links. The losses on the former are treated as presented in protocol 1 and the losses on the latter are treated differently. Here we give a detailed explanation on how the stabilizer links deal with losses in protocol 2.

Stabilizers in a repeater station: As in protocol 1, in protocol 2 we will also consider that at each stabilizer links one stabilizer is measured. If this measurement is unsuccessful due to losses in a repeater chain the protocol is aborted making the whole repeater chain fail. The chain needs to be composed of at least four links, such that at least a set of stabilizer measurements can be performed.

The choice of measuring one stabilizer per repeater station is motivated by the fact that the success probability of a stabilizer link is lower as more stabilizers are considered to be measured. The simplistic approach to deal with losses in the stabilizer links presented in protocol 1 is not ideal in terms of the success probability. Therefore, another scheme for tackling the losses and measuring stabilizers is proposed. Instead of using a trial-and-error approach we will make sure that the involved first-level qubits are not lost without perturbing the tree.

This is done in a similar way as for the re-encoding procedure by means of the method of heralded storage. By using the 5-qubit code as an outer code in each repeater station only one of the four stabilizers can be measured. Moreover, the stabilizers of this code have weight 4, hence we will need four first-level qubits per repeater station to measure a stabilizer. Then, a second heralded storage procedure will be performed in all the four involved trees in the stabilizer measurement, so that if it is successful we have four no-lost first-level qubits from each of the involved trees. Then the success of the stabilizer measurement will depend on the measurements second-level qubits, but if the depth of the tree is three or more these measurements are quite redundant and loss-proof.

To perform this extra heralded storage also an extra processor is needed. This heralded storage procedure for the stabilizer measurements is better in terms of success probability but worse on terms of resources, since it requires an extra processor. For this extra chip, two approaches are proposed, both of them are presented next. Following an analytical expression for the success probability for both approaches is derived.

> **Losses in stabilizers:** In protocol 2 the considered approach to deal with losses in terms of measuring stabilizers is the heralded storage approach, which means that before measuring the stabilizers, a heralded storage of the needed first-level qubits is performed to ensure the success of the stabilizer. If the heralded storage fails, it means that there is no available first-level qubits to measure the stabilizer. Moreover, the stabilizer can also fail if the (in)direct Z measurements on the involved second-level qubits are unsuccessful due to losses or failed majority voting.

Protocol 2: parallel stabilizer measurement

First the most straightforward method would be to consider that at the repeater station where the a stabilizer is to be measured an extra processor with five spin systems is needed. From these five, four would be the first-level qubits and the fifth one would act as an ancilla qubit. The latter is required in order to perform the $XXXX$ stabilizer between the first-level qubits. A diagram of the measurement protocol of this stabilizer measurement is presented in Figure 3.3. The circuit presented is derived from the general case of an indirect measurement of an M operator presented in 2.3.2.

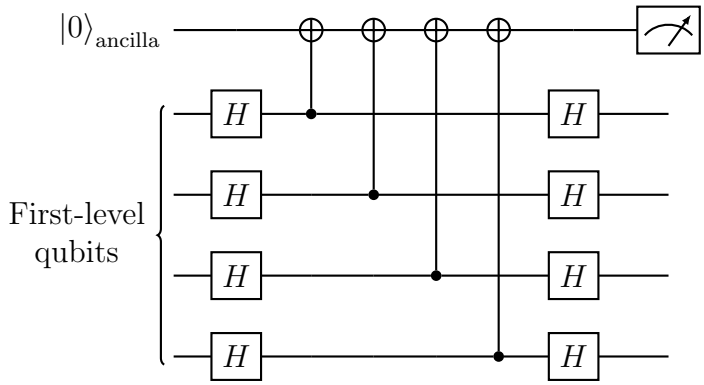


Figure 3.3: Quantum circuit for the $XXXX$ stabilizer measurement of four first-level qubits in the extra processor with five spin qubits.

Protocol 2: sequential stabilizer measurements

A more "cheap" approach on terms of the resources is instead of having five spin qubits in the extra processor we can have only two, one for the ancilla qubit and the other for one of the first-level qubits. The process to measure the stabilizer is the same as for the previous approach, but now the method is sequential, so not all the X measurements will be done at the same time. To start with, for one first-level qubit the circuit in Figure 3.4 is followed.

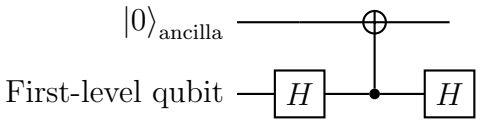


Figure 3.4: Quantum circuit for the indirect X for one of the four first-level qubits.

Then this first-level qubit is discarded and another first-level qubit is sent to the spin qubit and the circuit in Figure 3.4 is done again, and so on for all the four first-level qubits. After this indirect X is performed for all four the ancilla qubit is measured giving the result of the $XXXX$ stabilizer measurement. For any of the two approaches the number of needed first-level qubits to be able to decode and measure the stabilizer successfully are the same.

Protocol 2: analytical success probability

To be able to study the loss tolerance of the stabilizer links, an analytical expression for the success probability of one of them for the 5-qubit code concatenated with a tree code with branching vector $\bar{t} = [b_0, b_1, \ldots, b_d]$ for all trees can be determined. In this expression we will consider the success of the stabilizer measurement and the success of the decoding of all the trees. To get this analytical expression, we depart from the one for the tree code, this expression has already been used in the previous protocol for the success probability of a free evolution link, that is η_1 as presented in equation 3.4, as mentioned it can also be found in [49] and [15]. The expression for the probability

of having a successful indirect Z measurement on a qubit in the kth level was given in equation 3.3 and denoted R_k. Then, the probability of a tree to have two successful heralded stored first-level qubits, whose second-level qubits are measured in Z and that the other first-level qubits are measured (in)directly in Z is,

$$\eta_2 = \left((1-\mu+\mu R_1)^{b_0} - (\mu R_1)^{b_0} - b_0(\mu R_1)^{b_0-1}(1-\mu)\right)\left((1-\mu+\mu R_2)^{b_1}\right)^2. \tag{3.6}$$

This equation can be broken into several terms to clarify its meaning. The first term, which we will rename as $\gamma = (1-\mu+\mu R_1)^{b_0} - (\mu R_1)^{b_0} - b_0(\mu R_1)^{b_0-1}(1-\mu)$ is the probability of having at least two non-lost first-level qubits and that all the others are directly or indirectly measured in Z. This is the same as the sum of the probabilities of having i non-lost first-level qubits and the rest $b_0 - i$ first-level qubits being measured (in)directly in Z for $i = 2, \ldots, b_0$. This is the same as considering the probability of having *all* non-lost first-level qubits measured directly and indirectly, defined as γ_{all}, minus the probability of having all-lost first-level qubits measured in Z indirectly, which is γ_0, and minus the probability of having one non-lost first-level qubit measured directly and that all the others are indirectly measured in Z, which will be γ_1. So that γ is,

$$\gamma = \gamma_{\text{all}} - \gamma_0 - \gamma_1. \tag{3.7}$$

Then, the probability of a non-lost first-level qubit is the same as the probability of being able to measure it directly, so $1-\mu$. Next, the probability of measuring a first-level qubit both directly and indirectly is the sum of them $(1-\mu)+\mu R_1$. Then,

$$\gamma_{\text{all}} = (1-\mu+\mu R_1)^{b_0}, \tag{3.8}$$

so that γ_{all} is the probability that all the first-level qubits are not lost and can be measured directly and indirectly. Next,

$$\gamma_0 = (\mu R_1)^{b_0}, \tag{3.9}$$

which is the probability of the outcome where all the first-level qubits are lost and they can only be measured indirectly. Finally,

$$\gamma_1 = b_0(1-\mu)(\mu R_1)^{b_0-1}, \tag{3.10}$$

where we consider all the possible combinations, b_0, where a first-level qubit is not lost with probability $1-\mu$, and all the rest are lost and they can only be Z measured indirectly with probability $(\mu R_1)^{b_0-1}$. Then the second term of η_2 is just the probability of the successful (in)direct Z measurements on the second-level qubits of two sub-trees. For one sub-tree the probability of making an (in)direct Z measurement of all the second-level qubits is $(1-\mu+\mu R_2)^{b_1}$, then for two is only its square power, so $\left((1-\mu+\mu R_2)^{b_1}\right)^2$.

We also wish to know the success probability of a stabilizer link, this is such that four trees will need two heralded storage qubits and one tree only one, so that the probability of success in one stabilizer link is $\eta_1\eta_2^4$. Next, since we have four stabilizer links the probability is given by $(\eta_1\eta_2^4)^4$. If we now consider a set of $(n+4)$ where n is the number of free evolution links, which each of them has a success probability of η_1^5, the success probability is $\eta = (\eta_1^5)^n (\eta_1\eta_2^4)^4$. Finally, if we have k sets, so $k(n+4)$ links, the success probability is η^k.

Measuring stabilizers: In protocol 2 the considered approach to deal with losses in terms of measuring stabilizers is the heralded storage approach. This can be performed in parallel or sequentially. Both ways have the same success probability η^k and same number of required processors, five to generate the trees and one to measure the stabilizer per repeater station. They differ in the number of spin qubits in the extra processor, the parallel requiring five spin qubits and the sequential only two.

CHAPTER 4

Numerical investigation of the 5-qubit code

In the previous chapter we have analyzed the code concatenation of the 5-qubit code and the tree code as the encoding of the one-way quantum repeater. Moreover, two specific protocols, 1 and 2, for this particular encoding and its corresponding stabilizer measurements have been presented, such that to measure all four generators of the 5-qubit code at least four repeaters are needed, one per generator.

In this chapter the goal is to investigate the behaviour of this encoding presented in protocol 2, as explained in Section 3.6, and compare it to the one-way quantum repeater with the tree code, previously seen in Section 2.5. The two compared one-way quantum repeaters will be referred to as *concatenated* and *unconcatenated* correspondingly. The comparison is done in terms of error tolerance, meaning that we want to know whether using code concatenation as in protocol 2 the one-way quantum repeater is more resilient against errors. The reason why losses are not so deeply considered in this study is because both the concatenated and the unconcatenated one-way quantum repeaters are found to protect well against losses in the space parameter considered. Therefore, the emphasis of the investigation will be on error tolerance, which we hope to improve using code concatenation.

In this chapter, the following content is presented. First, in Section 4.1 we motivate and constrain the simulation of the code concatenated one-way quantum repeater. Next, in Section 4.2 we introduce the tools that are used in order to develop the simulation. Later in Section 4.3, a detailed explanation on the choice of the branching vector and loss probability for the simulation is presented. As a continuation, in Section 4.4 a verification of the simulation is presented followed by the results of a first approach in Section 4.5. Finally, an analysis of a better strategy, its analytical study and some preliminary results are given in Section 4.6.

4.1 Motivation and constraints for the simulation

To study the error tolerance of the concatenated and the unconcatenated one-way quantum repeaters one must acknowledge that the two systems have many variables that influence their behaviour. Moreover, the considered systems are composed by a large number of qubits with probabilistic errors and losses. Importantly the interplay between losses and errors and all the constrains of the protocol makes an analytical study a rather complicated path to follow. This interplay is clearly seen in the success probability of the repeater chain, since on the one hand it depends on the depolarizing errors of the photons due to the assumption that a majority voting of a Z measurement fails and no conclusion can be drawn making the measurement fail too. On the other hand, the success probability is also affected by losses, as it has been studied in 3.6.1. However, the complex relation between those two is unclear and it is the main motivation for developing a simulation for these systems. Moreover, a simulation can also help to study the optimization of certain variables and interpret the results of the protocol. Finally, a simulation is also useful to get an intuition on how the two systems behave under similar scenarios.

Here, we present, to the best of our knowledge, the first fully general simulation framework for studying the performance of tree-code based one-way quantum-repeater chains. In this thesis, we use this framework to perform an analysis of the potential of concatenating the tree code with a stabilizer code. However, the framework can potentially be used to investigate a number of questions about tree-code based architectures. These questions include, but are not limited to,

- How well does the tree-code based one-way quantum repeater perform in the presence of realistic (correlated) noise?
- Which is the interplay of losses and errors for the tree-code based one-way quantum repeater?

Before being able to prepare the simulation one needs to think about which are the variables that protocol 2 and the unconcatenated one-way quantum repeater have. From all these variables one has to decide which ones should be fixed to a certain value or range and which ones should be left as independent variables of the system. The variables still undetermined by protocol 2 are the branching vector of the used tree code, which as stated in 3.4 we consider it to be the same for all the trees and all repeater stations in the concatenated case, the loss probability μ and the depolarizing rate ϵ that the photons in the trees are subjected to when they are sent and the number of links in the repeater chain.

The branching vector is a parameter that in principle can be chosen freely, although there is some resource trade-off between the number of qubits and the simulation time. The loss probability is a function of the inter-node spacing and thus we would like it to mimic realistic settings. Finally, in this thesis we have chosen a depolarizing channel with the same probability for all the photons in the trees, which is not a realistic approach since, for example, from the tree generation procedure one can see that a correlated error

model for the photons in trees would be more appropriate. Nevertheless, the depolarizing channel is also chosen in [15] as a convenient noise model to draw some initial conclusions on the tree-code based one-way quantum repeaters.

Note that these variables are independent from one another and all together define the success probability of the concatenated repeater chain. For example, as seen in the expression for the analytical success probability of a $k(n+4)$-linked repeater chain, η^k in 3.6.1, takes the values of the branching vector, the loss probability and the number of links into account. Also, the depolarizing rate has an impact on the success probability due to the fact that the majority vote of Z indirect and direct measurements fails in case of disagreement. Thus, it may be that some Z measurements fail resulting in an unsuccessful repeater chain.

As a first step we will consider the simplest repeater chain in protocol 2, the repeater chain with four links, so that $k = 1$ and $n = 0$. This approach is considered as a first stepping stone towards seeing the effects of the stabilizer measurements and their corrections.

Since the goal of this chapter is to study the behaviour of the system under the depolarizing channel, it makes sense to fix the value of the loss probability and the branching vector and leave the depolarizing noise probability as an independent variable. The depolarizing noise considered has the same depolarizing rate for all the qubits in the trees, ϵ. To be able to choose these values the simplest method would be to make use of a simulation of the system with the presented constraints.

Additionally, in order to determine the impact that a message qubit sent through a repeater chain as the one described is subjected to, we want to look at the secret-key rate and the QBERs. We want to use the secret-key rate to characterize the results because, as explained, the interplay of errors and losses in the code-concatenated one-way quantum repeaters is unclear and this parameter combines both the rate, which relates to the loss tolerance, and the QBER, which relates to the error tolerance, into a single metric. In particular, the widely used BB84 protocol, as presented in 2.2.2, is considered, so that the message qubits will be the BB84 states, $|0\rangle$, $|1\rangle$, $|+\rangle$ and $|-\rangle$. For this purpose, considering a simulation of the studied system appears to be the most adequate choice.

4.2 The NetSquid Simulator

For our simulations we use **NetSquid**, which stands for **Net**work **S**imulator for **Qu**antum **I**nformation using **D**iscrete events [50, 51]. This is a simulator for quantum networks developed at QuTech freely available for non-commercial users. NetSquid stands out as a software tool able to model the effects of time on the performance of quantum networks and quantum computing systems. Moreover, its modular approach is an essential step for large scale quantum systems. Additionally, the front end is entirely written in Python which makes it very easy to use.

NetSquid has a lot of powerful features and functions, however, on our simulations only the more general and simple components and models are used. Moreover, this tool allows to use different formalisms, including the stabilizer one. In our simulations we will use a

formalism named *GSLC* that represents a quantum state using a graph state with local Cliffords as described in [52]. We choose to use this one because we empirically found it to be the most efficient one.

Importantly, the simulation framework we have developed for studying the performance of tree-code based one-way quantum-repeater chains is an add-on to the NetSquid Simulator and has been entirely created for the purpose of this thesis. It allows to track the full quantum state and supports arbitrary noise and loss models, which makes it a tool to be used straightforwardly in the future for the study of physically realistic noise models in the one-way quantum repeater.

Finally, to process the data and plot the processed data from the simulations and be able to study our systems in terms of their secret-key rate and QBERs of the BB84 protocol, concepts which have been previously defined in 2.2.2, the snippets of NetSquid provide us with the tools to do so. These snippets are extensions for NetSquid created by users and developers. The specific snippet used to process and plot data of the simulation is the NetSquid Snippet NetSquid-SimulationTools [53, 54].

4.3 Choice of branching vector and loss probability

The goal of this section is to describe the procedure followed to determine the branching vector, assumed to be the same for all trees at all repeater stations, and the loss probability, μ, which are used on the following sections of this chapter. We aim for a branching vector and a loss probability such that the success probability without considering errors is almost deterministic in the code concatenation approach. The reason why we want such a combination of branching vector and loss probability is because we want to perform simulations relatively fast, so that the repeater chain almost always succeeds. This means that we would be looking into a regime where both the concatenated and unconcatenated one-way quantum repeaters protect well against losses, given that the focus of the investigation is on the error tolerance.

In order to find the proper combination of those two parameters, one can look at the analytic expression for the success probability derived in 3.6.1. The expression η^k denotes the success probability for $k(n+4)$ links in the code concatenated one-way quantum repeater with the 5-qubit code and a tree code with branching vector $\bar{t} = [b_0, \ldots, b_d]$, so that there are k sets of n free evolution links and 4 stabilizer links where one stabilizer is performed successfully.

Another thing to take into account is that in the unconcatenated approach, presented in [15], they show that a tree of depth 3, is sufficient for transmission distances up to 1000 km assuming telecom frequencies. Since in the scope of this thesis we will be looking at four links we will also consider a tree code of depth 3, so that the branching vector has three numbers, $\bar{t} = [b_0, b_1, b_2]$.

Finally, it is important to note that increasing any of the numbers of the branching vector makes the number of qubits in a tree grow. For a single encoded tree the number of qubits is $\sum_{i=0}^{d} \prod_{j=0}^{i} b_j$, so for our case where we have five trees of depth 3 the number of qubits

is $5(b_0 + b_0b_1 + b_0b_1b_2)$. Using the GSLC formalism in the simulation makes the memory scale as $\mathcal{O}(nd + n)$, where n is the number of qubits in the quantum state and d is the average amount of edges per vertex in the graph state [51] and for large quantum states is the best formalism to use in terms of memory. Nevertheless, from our simulations we have empirically seen that increasing the number of qubits does not have a drastic impact on the simulation time. Therefore the number of qubits will also be considered in the decision for a branching vector for the simulation.

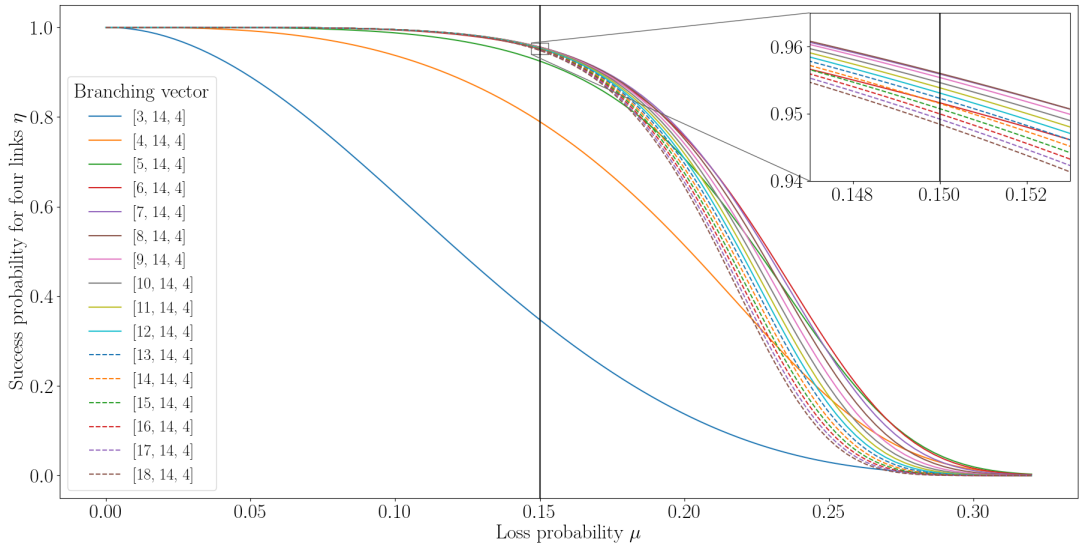


Figure 4.1: Analytical success probability (3.6.1) for a 5-qubit code concatenated one-way quantum repeater with four stabilizer links, η, in the in terms of the loss probability, μ, for different number of first-level qubits, b_0, for an error-less protocol 2. The vertical black line denotes $\mu = 0.15$, around this value a zoom is presented.

Considering all the presented values and with a first scan of parameter space we decided to explore branching vectors $\bar{t} = [b_0, b_1, b_2]$ such that b_0, b_1 and b_2 take values from 3 to 18. Moreover, we will look for which of those values the success probability is higher for a value of loss probability of $\mu = 0.15$. This value has been chosen taking into account all the previous considerations. Specifically, if we fix a higher μ the branching vector needed to have a success probability close to one would be bigger. Additionally, we consider that a considerably lower loss probability would be too close to zero in order to see the interplay of losses and errors.

Several plots for the value of the success probability η in terms of the loss probability for different values of the branching vector can be seen in Figures 4.1, 4.2 and 4.3. In particular, we analyze each of the three values of the branching vector separately. So that, the two values that are not being scanned are set to a tentatively final branching vector. In Figure 4.1 we see that for $b_0 \geq 6$ the success probability for $\mu = 0.15$ is very similar. Since we want to provide a scheme that works with minimal required resources and also taking into account the simulation time we have chosen the smallest option, so $b_0 = 6$. Next, in the Figure 4.2, one can observe that increasing b_1 the success probability increases, although once again taking into consideration minimal resources and the simulation time we opt for $b_1 = 14$. However, if we increased even more b_1 we could target for a higher loss probability, but for as argued before that would increase too much the simulation time.

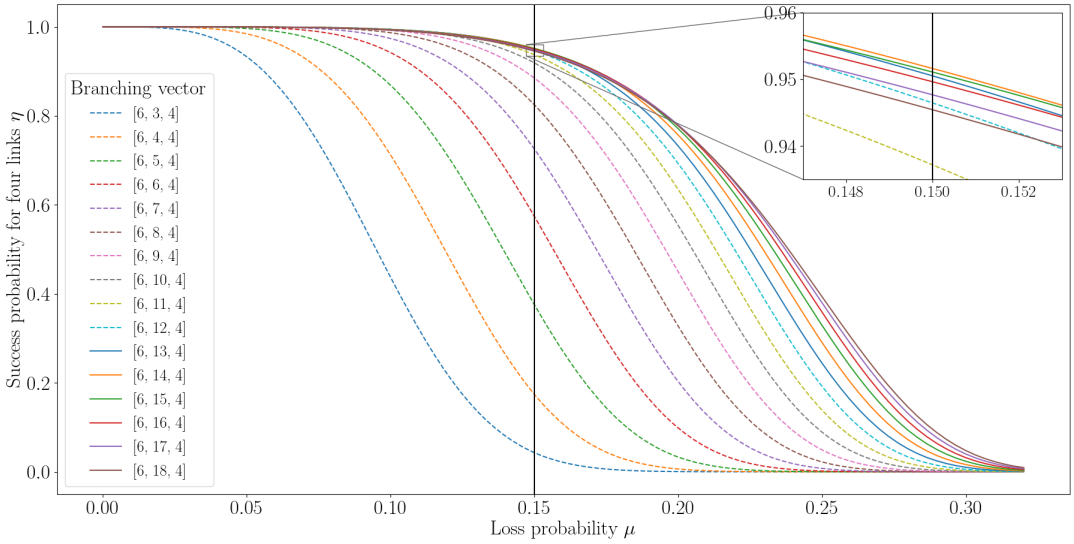


Figure 4.2: Analytical success probability (3.6.1) for a 5-qubit code concatenated one-way quantum repeater with four stabilizer links, η, in the in terms of the loss probability, μ, for different number of second-level qubits, b_1, for an error-less protocol 2. The vertical black line denotes $\mu = 0.15$, around this value a zoom is presented.

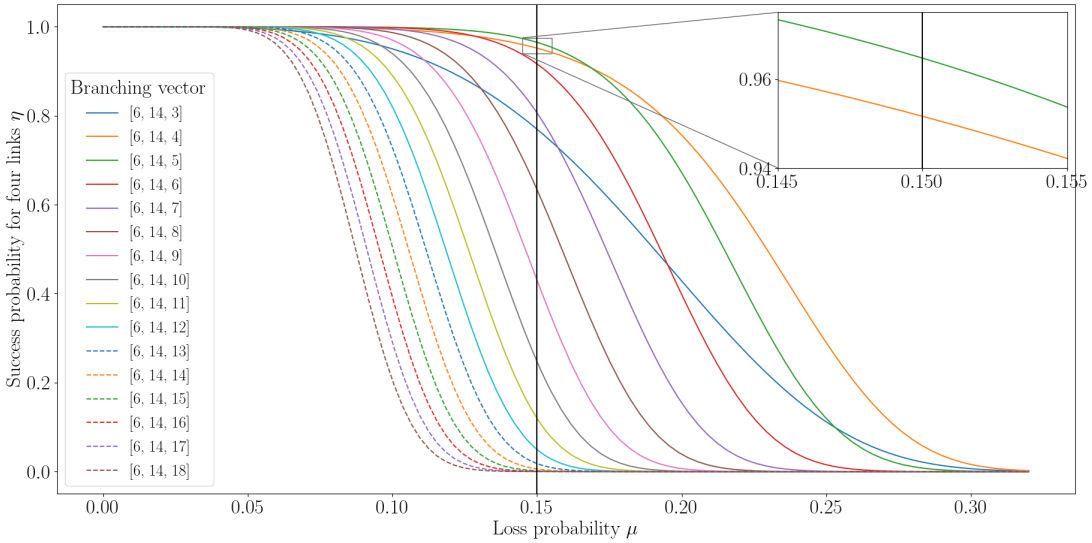


Figure 4.3: Analytical success probability (3.6.1) for a 5-qubit code concatenated one-way quantum repeater with four stabilizer links, η, in the in terms of the loss probability, μ, for different number of third-level qubits, b_2, for an error-less protocol 2. The vertical black line denotes $\mu = 0.15$, around this value a zoom is presented.

Finally, in Figure 4.3, strangely the best behaviour corresponds to $b_2 = 4$ and not to larger values. The reason why the third level shows this behaviour is because the qubits on that level, for example, in the decoding procedure, are measured in Z when performing an indirect Z measurement on a first-level qubit, but these direct Z measurements on the third level are not protected against losses, they would only be protected if there was a fourth level. Then we see that there is a trade-off between loss protection of the qubits in

different levels and it seems like the optimum value for the third-level qubits when they are not protected against losses is $b_2 = 4$.

Considering all the presented factors we have chosen the branching vector $\bar{t} = [6, 14, 4]$ with loss probability $\mu = 0.15$, such that the system will have 2130 qubits and an error-less success probability of $\eta \approx 0.95$.

4.4 Verification of the simulation

Until now we have seen the tools and the parameters that will be used in the simulation. In this section, the goal is to present specific scenarios in order to contrast the data generated by the developed simulation and the analytical expressions that we know from the system. In particular, we consider two scenarios from which we know what should be the outcome, such that the simulation can be verified. For both scenarios a branching vector of $[6, 14, 4]$ is considered.

First, we take a code concatenated one-way quantum repeater as defined in 3.6, which is formed by four stabilizer links. From this we study the probability of success for the error-less case, $\epsilon = 0$, which has been analytically studied in 3.6.1. With this information we hope to compare the average number of attempts per success, which is the inverse of the success probability $1/\eta$, in terms of the loss probability, μ, given by the simulation and given by the analytical function, $1/\eta(\mu)$.

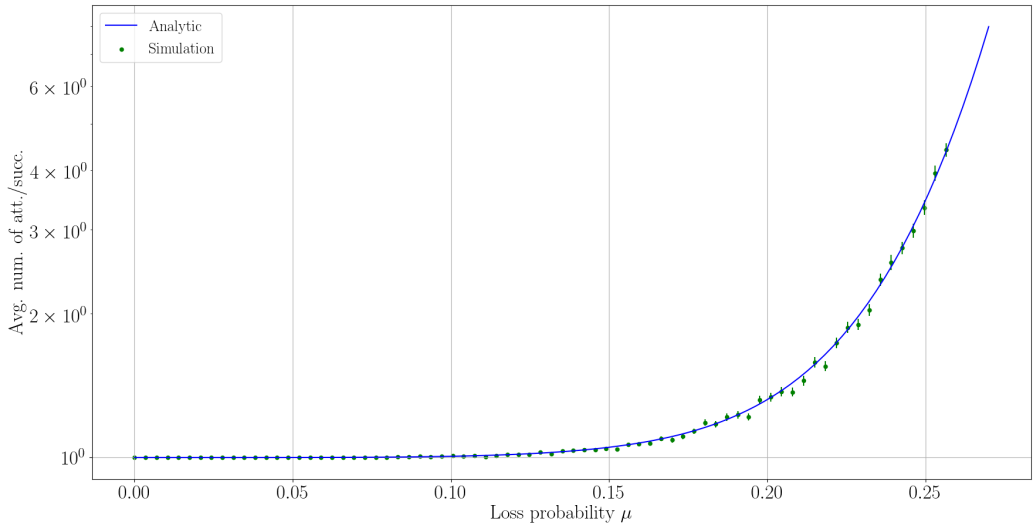


Figure 4.4: Average number of attempts per success in terms of the loss probability μ. This corresponds to four stabilizer links of a code concatenated one-way quantum repeater that follows protocol 2. The branching vector considered is $[6, 14, 4]$ and the depolarizing rate of the system is $\epsilon = 0$. The green points represent the data obtained from the simulation and the blue line represents the data obtained from the analytical expression $1/\eta(\mu)$. The number of samples taken per simulation data point is 600.

In Figure 4.4 we can clearly see that both the simulation points and the analytic line agree, therefore we consider that the simulation is verified in terms of losses.

Next, we also consider a code concatenated one-way quantum repeater as defined in 3.6, which is formed by four stabilizer links. To verify that the stabilizer measurements and the corresponding corrections work as intended, we assume that in the sending station before encoding the five qubits into trees only one of them is subjected to a depolarizing channel with probability 1. Meaning that I, X, Y or Z with same probability is applied to one of the five qubits that are encoded into trees. We consider this because in our case the outer code, the 5-qubit code, is able to correct arbitrary errors on a single qubit. Moreover, to verify the simulation results we consider that the four stabilizer links are free of error, meaning that $\epsilon = 0$. Therefore, forcing an error on one of the outer code qubits and assuming no errors on the photons in the trees must result in a final state with no errors, since the stabilizers measurements and the corresponding corrections must be able to correct the single-qubit error on the outer code.

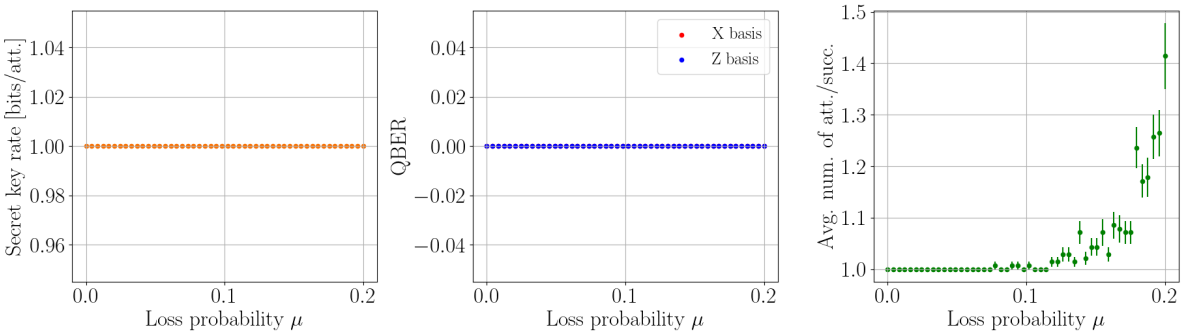


Figure 4.5: Starting from left to right, plot the secret-key rate, QBERs and averaged number of attempts per success in terms of the loss probability μ that the photons in the trees are subjected to. This corresponds to four stabilizer links of a code concatenated one-way quantum repeater that follows protocol 2. The depolarizing rate of the system is $\epsilon = 0$ and the branching vector is $[6, 14, 4]$. There is a depolarizing channel on one of the five qubits on the outer code with probability 1. The number of samples taken per data point is 140.

In Figure 4.5 we can see that the expected behaviour is obtained by the simulation, since both the QBER in the Z and in the X basis are zero always, meaning that the final state does not contain any error. Note that the X basis QBER is overlaid by the Z basis QBER, and hence is not seen in Figure 4.5. This verifies the proper behaviour of the stabilizer measurements and their corrections.

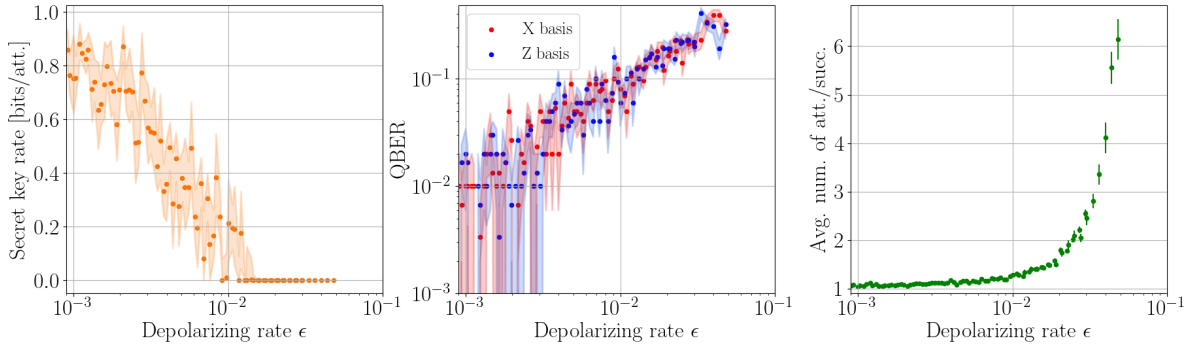
4.5 Protocol 2 with four links

In this section we present the results for the one-way quantum repeater using code concatenation of the 5-qubit code and the $[6, 14, 4]$ tree code considering protocol 2, seen in Section 3.6. The qubits in the trees when sent are subject to an erasure channel with loss probability $\mu = 0.15$ and to a depolarizing channel with probability ϵ, note that the depolarizing rate is the same for all the qubits. The considered repeater chain is formed by

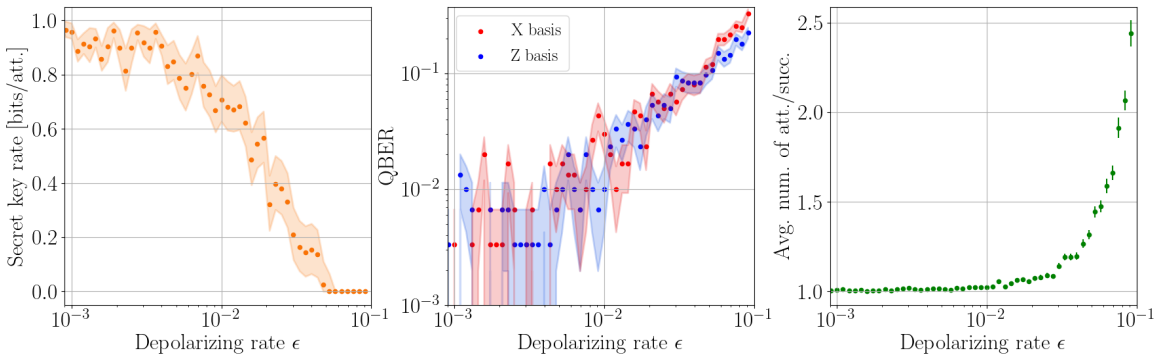
four links where all four stabilizers are measured once and if any of them fails the whole chain is said to fail. In the end node before decoding the 5-qubit code the syndrome corrections due to the stabilizer outcomes are applied. This is such that the number of sets is $k = 1$ and the number of free evolution links is $n = 0$, as defined in protocol 2.

To get an idea on whether this particular scenario behaves better than an unconcatenated one-way quantum repeater, we will also present the results for the unconcatenated one-way quantum repeater using the $[6, 14, 4]$ tree code. Again, the qubits in the tree when sent they are subject to an erasure channel with loss probability $\mu = 0.15$ and to a depolarizing channel with probability ϵ, note the depolarizing rate is the same for all the qubits. The considered unconcatenated repeater chain also consists in four links.

Both results have been generated using the simulation using NetSquid and the data has been processed and plotted using the NetSquid-SimulationTools Snippet.



(a) Concatenated one-way quantum repeater with four links



(b) Unconcatenated one-way quantum repeater with four links.

Figure 4.6: Both figures starting from left to right, plot the secret-key rate, QBERs and averaged number of attempts per success in terms of the depolarizing noise ϵ that the photons in the trees are subjected to. The number of samples taken per data point ranges from 200 to 600 in (a) and it is 600 in (b). (a) and (b) correspond to an one-way quantum repeaters of four links with fixed loss probability $\mu = 0.15$. (a) denotes a code concatenated one with a $[6, 14, 4]$ tree code and the 5-qubit code that has the constraints described in protocol 2, seen in Section 3.6. (b) represents the unconcatenated one with a $[6, 14, 4]$ tree code as studied in Section 2.5.

From Figure 4.6 one can see that the behaviour for the unconcatenated one-way quantum repeater, (b) in Figure 4.6, is better than the concatenated one, (a) in Figure 4.6. First,

in the rightmost plots in both (a) and (b), one can see that the concatenated has a higher average number of attempts per success than the unconcatenated one for large values of ϵ. Remember that we have considered a majority vote of the Z indirect and direct measurements that fails if there is no agreement, which affects in general the performance of the repeater chain as ϵ increases. In particular, it disturbs more the concatenated approach than the unconcatenated one, since in the former, which is defined by protocol 2, there is an additional constraint compared to the latter of having to measure successfully a stabilizer per repeater station. Moreover, another factor that explains this behaviour is that the concatenated one considers five trees being sent from a repeater station to another, where they need to be decoded besides measuring the stabilizer, so that the probability of all trees being successfully decoded taking into account the restrained majority vote must be less compared to the one tree being decoded in (b) for a high enough value of ϵ.

On another note, let us focus on the plot for the QBER for the unconcatenated case in the middle plot in Figure 4.6(b). One can see that in general the QBER in the X basis is higher than the one in the Z basis. This is an expected result, because if we recall the decoding procedure of the tree code is such that after every Z (in)direct measurement a correction needs to be applied on the final state. These corrections are Z operations on the first-level qubit that will be used to either re-encode or in the end node retrieve the information. Therefore, as ϵ increases the probability of making the wrong corrections due to this depolarizing noise on the photons on the tree increases too. Of course, this effect is slightly mitigated by the fact that we are majority voting the direct and indirect Z measurements on a qubit before performing the correction and also by the fact that an inconclusive majority voting will make the measurement fail thus preventing faulty corrections.

If now we look at the two plots on the left and middle in (a) and (b) in Figure 4.6, we can see that in (b) for all the range of ϵ, the secret-key rate is higher and the QBERs are smaller than the concatenated one, (a). Both of them share two sources of errors, the first one being the error that the first-level qubit used for the re-encoding may have and the second one being the error that is also caused by the depolarizing noise on the rest of the qubits. As explained previously in 2.4.2 this error is mostly overcome by making use of the redundant Z measurements on qubits and their majority vote. Nevertheless, it may happen that this is not enough and the wrong outcome from the Z measurements is given, leading to a wrong correction to the remaining state.

However, the concatenated repeater chain has an extra source of error, the stabilizer outcomes. These can be faulty and lead to a wrong correction on the 5-qubit code state before this is decoded in the end node. This faultiness of the stabilizer was presented in 3.5.3. Then, the noticeable difference between the two encoding clearly points out that the stabilizer outcomes are faulty very often and that this faultiness increases with ϵ. Although this result may not be encouraging it makes sense and reveals that using the code concatenation in longer repeater chains would be a better approach. Specifically, we would need to consider a number of free evolution links higher than zero, so $n > 0$, in order to have an error build up which can maybe corrected with the stabilizer outcomes and thus maybe improve the error tolerance of a set of $n + 4$ links. This preexisting or build-up error needs to be larger than the probability that your stabilizer measurement is faulty reducing the QBER of the resulting state after the correction is applied.

4.6 Improving the strategy

As seen in the previous section, the strategy of only having four stabilizer links in the code concatenated one-way quantum repeater with the 5-qubit code needs to be improved. Following the intuition that we have gained throughout the project, we have proposed that targeting longer repeater chains with $n > 0$ free evolution links may benefit a set of $n+4$ links. The goal of this section is precisely to investigate again protocol 2, presented in Section 3.6, considering $n > 0$.

First we look into how to mimic the free evolution links in the simulation. Next, we consider perfect stabilizers to properly define a regime for the number of free evolution links and get an intuition on the behaviour of the system. Afterwards, an analytical study of k sets of $n + 4$ links is performed. Finally, a first look into imperfect stabilizers is presented.

4.6.1 Error build-up for the free evolution links

The previously prepared simulation does not allow to perform long repeater chains in a relatively short period of time, therefore we have come up with an alternative to mimic the effect of a long repeater chain. To start with, let us consider a single set of $n + 4$ links in the repeater chain, only needing to imitate the effect of the error built up after n free evolution links. To get an intuition how the error is built up in the free evolution links we can look at the evolution of the unconcatenated one-way quantum repeater in Figure 4.6(b). As stated in 3.5.3, the behaviour of a free evolution link is the same as five simultaneous unconcatenated links. One can see a free evolution per tree as a single quantum channel on a single qubit, as a qubit is encoded at the beginning and then decoded at the end. Therefore, we can say that each of the qubits sent in a free evolution link will end with the same QBER as an unconcatenated link.

From Figure 4.6(b) we can see that the QBER in the X and Z basis are not the same. The difference between them can not be appreciated for small values of ϵ due to the large error bars caused by the inclusion of zero. However, for values close to $\epsilon = 10^{-1}$, the difference between Q_X and Q_Z is about 0.1. More precisely the X basis QBER tends to be higher than the Z one. Thus, this shows that the error after four non-simultaneous unconcatenated links does not correspond to a depolarizing channel, which would have equal X and Z QBERs, $Q_X = Q_Z$. Nevertheless, the difference between them is not drastic, thus we consider the approximation where these four unconcatenated links behave as a depolarizing channel on the message qubit in order to simulate the n free evolution links.

Taking the assumption that four unconcatenated links behave as a depolarizing channel, we can say that an unconcatenated link with depolarizing noise on the photons in the tree behaves like a depolarizing channel on the message qubit with depolarizing rate ϵ_I. Then, since a free evolution link corresponds to five simultaneous unconcatenated links we will consider a free evolution link to be five simultaneous depolarizing channels on each of the five qubits of the outer code all with depolarizing rate ϵ_I.

To be able to move forward we need to present a property from depolarizing channels. This is that if there is a depolarizing channel concatenated n times with depolarizing rate ϵ_I this corresponds to a single depolarizing channel with depolarizing rate ϵ'_I, such that,

$$\epsilon'_I = 1 - (1 - \epsilon_I)^n. \tag{4.1}$$

This property can be easily derived, by considering $n = 2$, where after the first depolarizing channel one is left with a state which will not have a depolarizing error with probability $1 - \epsilon_I$. If we now go through the second one, the probability that the final state does not have an error is the product of the probability of not having an error on the first channel and of not having an error on the second channel, so $(1-\epsilon_I)^2$. Therefore the depolarizing rate of the two depolarizing channels together is $1 - (1 - \epsilon_I)^2$. This procedure can be generalized for n depolarizing channels easily such that you get the stated property.

In our case, we are considering exactly what is proposed on this property. Each of five unconcatenated simultaneous links that form a free evolution link will be a depolarizing channel with depolarizing rate ϵ_I. Then for n free evolution links we will consider to have five simultaneous depolarizing channels with depolarizing rate ϵ'_I, such that the two depolarizing rates are related by equation 4.1.

Now, the question is how to relate ϵ_I with ϵ from the depolarizing rate that the photons in the trees are subjected to. In order to answer this we will look at the results of the simulation for four unconcatenated links in Figure 4.6(b). For a certain value of ϵ we look at the corresponding QBERs and take the one with higher value, since we want to overestimate error rates in order to get a lower bound on performance, usually the higher QBER will be the one in the X basis. Then since a depolarizing channel is such that the QBER in both bases are the same, $Q_X = Q_Z$ and $\epsilon'_I = Q_X + Q_Z$, we multiply the higher QBER by two to obtain ϵ'_I for four links, which we will denote as ϵ'_4, so that $\epsilon'_4 = 2Q_X(\epsilon, \mu)$, where $Q_X(\epsilon, \mu)$ denotes the QBER in X basis as a function of ϵ and μ. Then, the depolarizing rate for a single link, ϵ_I, is,

$$\epsilon_I = 1 - (1 - \epsilon'_4)^{1/4} = 1 - (1 - 2Q_X(\epsilon, \mu))^{1/4}. \tag{4.2}$$

Unfortunately, there is no analytical function to relate Q_X with ϵ and μ, but the simulation yields to this information. Finally, we can conclude that n free evolution links behave like a depolarizing channel with depolarizing rate ϵ'_I, such that it is a function of the number of free evolution links n, the depolarizing rate of the photons in the trees ϵ and the loss probability μ, so that,

$$\epsilon'_I = \epsilon'_I(n, \epsilon, \mu). \tag{4.3}$$

We have presented all this to explain how we are going to mimic the n free evolution links in a set of $n + 4$ links in the simulation. The way that we are going to do this is by applying a depolarizing channel with depolarizing rate ϵ'_I to each of the five qubits of the 5-qubit code in the sending station before they are encoded into trees and sent through 4 stabilizer links, where the qubits in the trees will be subjected to an erasure channel with loss probability $\mu = 0.15$ and a depolarizing channel with probability ϵ.

4.6.2 Perfect stabilizer measurements

Consider again one set of $n+4$ links, first, it is important to build an intuition on which regimes of ϵ'_I as the final depolarizing rate of n free evolution links can be beneficial for the entire repeater chain. In order for the concatenated one to be "better" or beneficial in any sense compared to the unconcatenated one, the QBER of the former must be lower than the QBER of the latter. This requirement is due to the fact that the rate of the concatenated one will never be "better" than the rate of the unconcatenated one. Thus in order to have a higher secret-key rate, the QBER must be lower as stated. Therefore, we say we have benefit when the QBER of the final state of the $n+4$ links from protocol 2 is smaller than the QBER of the unconcatenated repeater chain with also $n+4$ links.

In order to get this intuition we will assume a case with *perfect* stabilizers. This case is such that we consider the depolarizing rate in the last four links, where the stabilizers are measured in the concatenated case, to be zero. Therefore, the stabilizers are not subject to any errors so that not only can we ensure they will be non-faulty, but *perfect*. We want to compare once again the concatenated and the unconcatenated cases to be able to determine if the stabilizers are beneficial for the system.

For this specific case we do not need to use the simulation, since we can derive an analytical expression for the QBER for both the concatenated and the unconcatenated cases. In the concatenated case, we are assuming that the five qubits will have depolarizing noise, and since there is no error in the 4 stabilizer stations we can only focus on the 5-qubit code and ignore the tree code. Therefore, since the 5-qubit code is able to correct arbitrary errors on a single qubit, meaning that if we have more than one qubit with an error we will not be able to correct it. When the 5-qubit code is able to correct, the quantum state after the four error-less stabilizer links has $Q_X = Q_Z = 0$. In the case where the 5-qubit code is not able to correct, we will approximate the quantum channel to a depolarizing channel with probability κ, thus having $Q_X = Q_Z = \kappa/2$, where κ is the probability that the 5-qubit code cannot correct. Then, the probability that the 5-qubit code can correct, $(1-\kappa)$, is the sum of the probability that there is no error, which is $(1-\epsilon'_I)^5$, and the probability that there is one qubit with an error, which is $5\epsilon'_I(1-\epsilon'_I)^4$. Therefore,

$$\kappa = 1 - (1-\epsilon'_I)^5 - 5\epsilon'_I(1-\epsilon'_I)^4. \tag{4.4}$$

For the unconcatenated case, since we have assumed that the last four unconcatenated links have no error, the final error of the state will be the same as the initial, therefore $Q_X = Q_Z = \epsilon'_I/2$.

In Figure 4.7 we have plotted the QBER in terms for ϵ'_I of both analytical functions. This plot helps us analyze which is the regime for which ϵ'_I can be beneficial. One can clearly see that this regime ranges from $\epsilon'_I = 0$ to approximately 0.13.

4.6.3 Behaviour of a long chain

As we have just seen, in the very ideal case of perfect stabilizers we can benefit from code concatenation as presented in protocol 2 for a single set of $n+4$ links when ϵ'_I is from

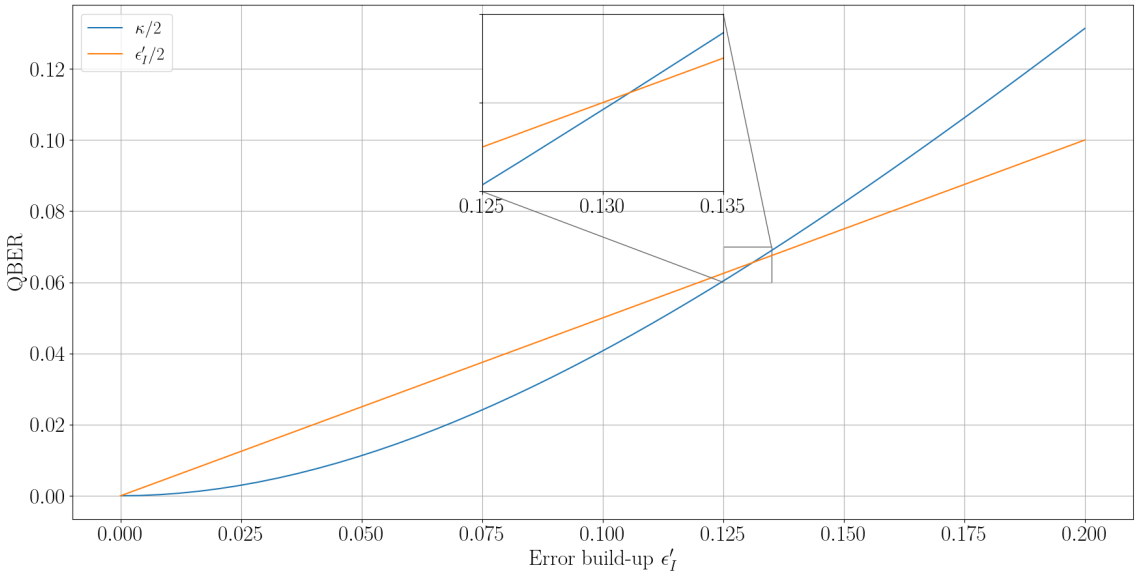


Figure 4.7: Plot for the QBER as a function of ϵ'_I. The orange line corresponds to the unconcatenated one-way quantum repeater with $n+4$ links where the n initial ones are modeled to have an error build-up of ϵ'_I and the next 4 links are error-less. In this case QBER $= \epsilon'_I/2$. The blue line represents the concatenated one-way quantum repeater with protocol 2 with $n+4$ links where the n free evolution links have an error build-up of ϵ'_I and the next 4 stabilizer links are error-less. In this case, QBER $= \kappa/2$, where κ is the probability that the 5-qubit code cannot correct.

0 to approximately 0.13. If we consider this regime, this means that after a set of $n+4$ links we "bring down" the QBER of the outer code state.

Consider a longer repeater chain with $k(n+4)$ links where a single set of $n+4$ links can be characterized so that all the sets can be treated independently. Then if the behaviour of a single set is better than the same number of unconcatenated links, then a protocol 2 chain of $k(n+4)$ links also should behave better than $k(n+4)$ unconcatenated links.

In order to characterize a set, we can interpret $n+4$ links as a depolarizing channel with probability $\epsilon''_I = 2\text{QBER}$, where QBER is $\kappa/2$, so $\epsilon''_I = \kappa$ and κ is defined in equation 4.4. If we consider a larger number of sets, such that $k > 1$, then we can understand the whole repeater chain as a concatenation of k depolarizing channels formed by $n+4$ links each. Next, by following the property of depolarizing channels, the depolarizing rate after all the $k(n+4)$ links is

$$\epsilon'''_I = 1 - (1 - \epsilon''_I)^k = 1 - (1 - \kappa)^k, \tag{4.5}$$

so that the QBER of the final state will be $\epsilon'''_I/2$. Moreover in protocol 2 we have derived an analytical expression for the error-less success probability of a set of $n+4$ links, η as presented in 3.6.1, and the total success probability of $k(n+4)$ links can be specified in terms of η, such that the final success probability is η^k.

Finally, we have seen that both the error probability and the success probability of k sets can be studied as separated k sets, therefore we can conclude that for a beneficial regime of ϵ'_I for a single link, the repeater chain is also beneficial and can be studied in terms of

the different sets of $n + 4$ links.

4.6.4 Imperfect stabilizer measurements

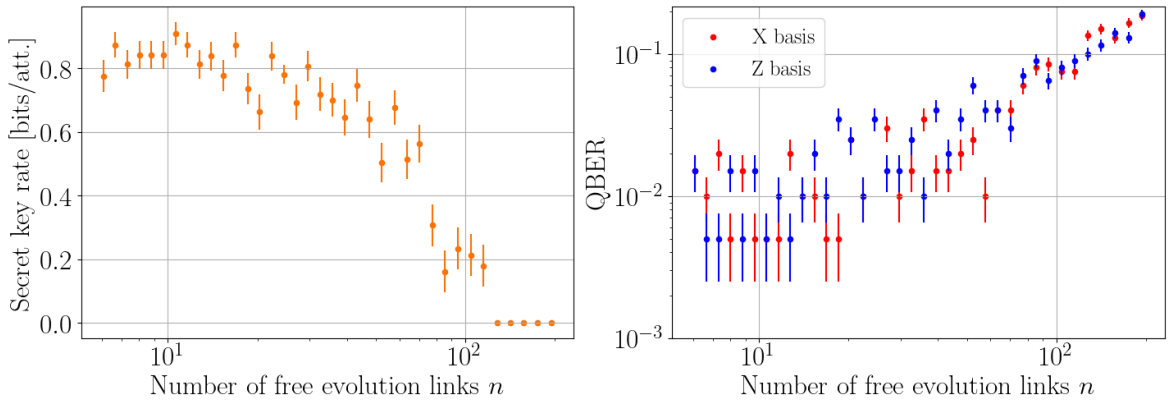
So far we have studied the $n+4$ links for the perfect stabilizers, but this is a non-realistic situation. Therefore, in this subsection we want to analyze the imperfect stabilizers, meaning that $\epsilon > 0$ in the last 4 stabilizer links of one set of protocol 2. We want to investigate the regime for which the error build-up of the free evolution links, ϵ_I', is beneficial[1] for the system with imperfect stabilizers.

In this case, to get results we will make use of the simulation to be able to see if noisy stabilizers can correct the initial error build-up. In order to know if the system is beneficial we also present the unconcatenated one-way quantum repeater for the same scenario. The following simulations are for four links which in the case of the concatenated one are stabilizer links. For the code concatenated one-way quantum repeater, just before encoding the 5-qubit code in the sending station a depolarizing noise with probability ϵ_I' is applied to all five qubits imitating the build-up error of the free evolution links before we simulate the 4 stabilizer links. For the unconcatenated one, the same structure is applied such that a depolarizing noise with probability ϵ_I' is applied to the message qubit before being encoded into a tree in the sending station, once again imitating the initial build-up error. For the following simulations the branching vector $[6, 14, 4]$ and the loss probability $\mu = 0.15$ are again considered, since we are again looking effectively at only four links. Now, the problem has become two-dimensional, since the values or range for ϵ and ϵ_I' need to be explored. Remember, that these two parameters are not independent from each other as seen in 4.6.1.

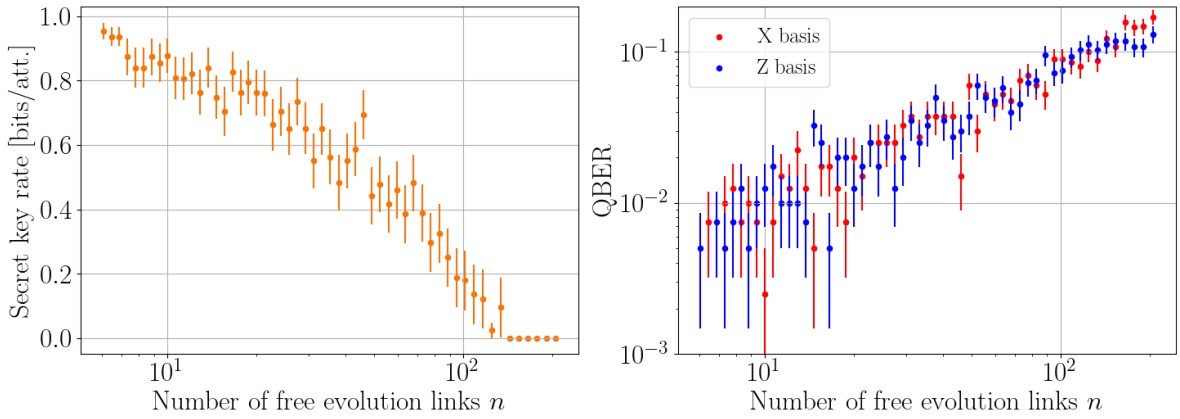
In order to find which regimes of ϵ_I' may be beneficial for a certain ϵ, we restrict the study to the regime of ϵ_I' where the perfect stabilizer are beneficial, as seen in 4.6.1 this is approximately $0 < \epsilon_I' < 0.13$. It is important to note that if ϵ_I' is very close to zero then in terms of the simulation we need to succeed at many attempts to actually see the impact of ϵ_I', since the probability that we get an error on the initial qubits is very low. Moreover, we are also interested in relatively high values of ϵ_I', such that the error build-up is larger than ϵ, because as seen in Section 4.5 if we consider $n = 0$ where there is no error build-up the faultiness of the stabilizer dominates the system and it is not beneficial at all.

In order for the concatenated approach to have a lower QBER than the unconcatenated and thus have a beneficial behaviour, there should be an optimal range for ϵ_I' for each ϵ, meaning an optimal range of n free evolution links in a set with a certain ϵ, μ and $\bar{t}$. On the one hand, if n is too low then the error build-up is too low in order for the stabilizer links to benefit the result, meaning that the faultiness of the stabilizer links results in a higher QBER. On the other hand, if n is too large the error build-up may be too large for the stabilizers of the 5-qubit code to correct it, since the 5-qubit code can only correct

[1]The benefit is defined such that if a set of protocol 2 is beneficial when the QBER of the final state of the $n + 4$ links from protocol 2 is smaller than the QBER of the unconcatenated repeater chain with also $n + 4$ links, for the same error build-up and loss probability.



(a) Concatenated one-way quantum repeater with four links and n free evolution links.



(b) Unconcatenated one-way quantum repeater with four links and n free evolution links.

Figure 4.8: Both figures starting from left to right, plot the secret-key rate, and QBERs in terms of the n free evolution links. (a) and (b) correspond to an one-way quantum repeaters of four links with fixed loss probability $\mu = 0.15$ and depolarizing rate $\epsilon = 10^{-3}$ that the photons in the trees are subjected to. The number of samples taken per data point is 400 (a) and 800 in (b).(a) denotes a code concatenated one with a $[6, 14, 4]$ tree code and the 5-qubit code that has the constraints described in protocol 2 in Section 3.6 with 4 stabilizer links. (b) represents the unconcatenated one with a $[6, 14, 4]$ tree code as studied in Section 2.5 with four links.

single-qubit arbitrary errors. Thus, this optimal regime is the area where $n + 4$ links of protocol 2 result in a lower QBER than $n + 4$ unconcatenated links.

First, the simulation for $\epsilon = 10^{-3}$ has been performed for a regime of $10^{-2} \leq \epsilon_I' \leq 2 \cdot 10^{-1}$. This regime for fixed $\epsilon = 10^{-3}$ corresponds to the number of number of free evolution links n, which can be obtained from equation 4.1, that is,

$$n = \frac{\ln(1 - \epsilon_I')}{\ln(1 - \epsilon_I)}, \tag{4.6}$$

where ϵ_I corresponds to the depolarizing rate of a single link and can be retrieved as in equation 4.2. Moreover, since ϵ_I' is a continuous function, the values for n are also continuous, meaning that the value satisfying this equation is not always an integer. From the simulation done in Section 4.5 we can get the value of Q_X for $\epsilon = 10^{-3}$ in the

unconcatenated one-way quantum repeater with four links, and that is $Q_X \approx 0.0033$. Then the regime for n is from 5 to 200 free evolution links.

The result of this simulation can be seen in Figure 4.8, where we can observe that the behaviours of secret-key rate and QBER of the concatenated (a) and unconcatenated (b) one-way quantum repeaters are very similar for the studied regime of ϵ'_I or n. The size of the error bars in the QBER vs. n plot does not allow us to get an exact idea if the code concatenation is beneficial like in the case of perfect stabilizers.

Unfortunately, the results of this simulation do not tell us if the code concatenation as presented in protocol 2 is beneficial or not. However, the differences between the concatenated and the unconcatenated approaches in Figure 4.6, where $n = 0$, are more drastic that the differences in Figure 4.8, where $n > 0$, as we had initially guessed. This approach opens the door to further simulations to try to find this optimal n for a certain ϵ, μ and $\bar{t}$. In order to draw a conclusion a further investigation with simulations for a variety of values of ϵ with more data points should be done.

CHAPTER 5

Conclusions and outlook

Here we conclude the thesis with a discussion of the results of this work and practical considerations. We additionally discuss the prospect of future work that may extend upon what has been presented.

5.1 Summary and conclusions

Throughout this thesis we have explored methods aimed towards achieving fault tolerance tree-code based one-way quantum repeater making use of code concatenation. Moreover, we did this keeping minimal the required resources and trying to maintain a fast transmission rate of the quantum repeater. For this reason, the core of our work has been focused on establishing a protocol for the code concatenated one-way quantum repeater using the 5-qubit code and the tree code. However, this may not be the optimal protocol for achieving fault-tolerance but a first step towards it regardless.

In Chapter 3, we have discussed and developed the specific constraints of our protocol in order to deal with an erasure channel with probability μ and a depolarizing channel with probability ϵ on photons sent between nodes. In order to have a fault-tolerant one-way quantum repeater chain we propose to measure the stabilizers of the 5-qubit code using the photons in the trees that in [15] were only used for the redundant decoding of tree. The proposed structure requires 5 processors with two memory qubits and a quantum emitter per repeater station and an extra processor with at least two memory qubits in each repeater station where stabilizers are measured.

Lastly, in Chapter 4 we focus on the error tolerance of protocol 2, presented in Section 3.6, in comparison to the unconcatenated protocol, seen in Section 2.5. From the results of protocol 2 with only four links, we conclude that the stabilizer links by themselves lead to a higher error on the final state than the unconcatenated links under same the circumstances. This introduces a search for an optimal number of free evolution links, $n > 0$, for which there is an error build-up that can be corrected by the stabilizer links. In order to mimic the free evolution links, we assume that each of them behaves like five simultaneous depolarizing channels. Under this assumption we consider perfect stabilizers

which shows an improvement to the error tolerance of the one-way quantum repeater and thus allows for individual treatment of the different k sets in a $k(n+4)$-linked repeater chain. Finally, we consider imperfect stabilizers with $n > 0$, which lead to an improvement compared to the imperfect stabilizers with $n = 0$, since the results for $n > 0$ of the concatenated and unconcatenated with same n, μ, $\bar{t}$ and ϵ show almost the same behaviour. In order to draw a stronger conclusion, a further investigation with simulations for a variety of values of ϵ with more data points should be done. Moreover, we consider this last investigation to open the door to further and improved simulations to get a very precise distinction between the concatenated and unconcatenated one-way quantum-repeater chains.

5.2 Future outlook

Many different approaches to several distinct problems have been explored, yet not all of them were possible to be studied deeply enough in the framework of this thesis. Nevertheless, we present them in the following paragraphs as ideas for a future work towards a fault-tolerant one-way quantum repeater using code concatenation of a stabilizer code and the tree code.

First, in our protocols we have always considered a structure where stabilizers are not always measured, but instead there are free evolution links. We have shown that they are necessary if we want to correct after successfully measuring a set of stabilizers, thus, having the possibility of treating the long repeater chain as k segments with a studied error and loss tolerance. However, if one did not correct every time the stabilizers are measured but after some or all sets are measured, one can see that the outcomes of the same stabilizer in different points of the repeater chain must be correlated between them and with the error build-up. Therefore, one should consider a smarter way of correcting. For example, we take as a base the presented structure for correcting errors in the surface code [55] using recurrent neural networks in the context of machine learning in [56], where they outperform the widely used minimum weight perfect matching [57] decoder. It is argued that this outperformance is due to the fact that using recurrent neural networks [58] for decoding can track correlated errors, something that with the common minimum weight perfect matching decoder is impossible to do. Thus, we can argue that in our case if we consider a final or smaller number of intermediate corrections, a decoding scheme for the outer code using recurrent neural networks would be a perfect fit.

However, always having stabilizer links can greatly affect the success probability, since the presented approach gives another purpose to the photons in the trees by using them to measure stabilizers. Therefore, one can also come up with alternative methods to measure the stabilizers making use of the sub-tree that is used in the heralded storage for the re-encoding at each repeater station. An example on how to do this would be to create a GHZ state $(|0\ldots0\rangle + |1\ldots1\rangle)$ between the quantum emitters in each processor in a repeater station. Then in order to perform the $X\ldots X$ stabilizer between first-level qubits that have been heralded stored for re-encoding, one would only need to perform a CNOT between the first-level qubit and one of the qubits that forms the GHZ state. Later the GHZ-qubits should be measured in order to perform a parity check and detect

if an error has happened and where. This approach has the advantage that no extra processors are needed at each repeater station and tentatively the success probability when measuring stabilizers would not decrease, since no extra qubits from the tree are required to not be lost. However, this approach does not allow measuring more than one stabilizer once per repeater station, whereas using the other photons in the tree to measure stabilizer allows measuring a stabilizer or multiple more than once per repeater station.

One of the main encountered challenges in order to simulate long repeater chains is the simulation time required. In order to avoid it, a smart approach would be to study a single link in detail so that a model of one link could be derived. Then, to study a long repeater chain the full simulation would not be necessary but one could simply take the derived model for a single link as many times as links in the chain. This approach would variate on the specific protocol on the links, since different kinds of links would require different models. This suggests that a protocol where all links behave the same would be the best problem to tackle in this fashion.

Moreover, if one considers a long repeater chain formed by stabilizer links in order to not perturb the success probability that much, one can relax the constraint of having to successfully measure a stabilizer per stabilizer link. Instead one can always try to measure a stabilizer once or multiple times per repeater station so that in the end of the chain one can have as many outcomes as possible without decreasing a lot the success probability of the chain. Additionally, if in one repeater station a single stabilizer is measured multiple times, then those outcomes can be majority voted and ensure a non-faulty stabilizer outcome. Nevertheless, the decoding in this strategy would not be as straightforward as the one presented as there are more outcomes to be considered, one would need to carefully study the correction step for this case. Moreover, this strategy may lead to a larger error rate compared to the presented one, where we require successful stabilizer measurements.

Note that due to the fact that in the presented protocol the 5-qubit code is used, we have restrained to a single stabilizer measured on a link, so that at least four links are necessary to measure a set of stabilizers. But if one desired to look at bigger outer codes one may need to measure multiple stabilizers in a link. For example, the surface code [55] has suitable stabilizers of the form $XXXX$ and $ZZZZ$ for the structure considered. So that in this case at least two links would be necessary, but many more stabilizers would need to be measured in a single link, rendering the strategy proposed in this thesis as potentially not adequate for this outer code. Moreover, the 5-qubit code was chosen mainly due to the fact that is small and easy to study. Then, looking at bigger codes could also improve the performance of the correction on the outer code, since some bigger codes protect against more than a single-qubit arbitrary error. The built simulation framework could be used as a tool to investigate this larger codes.

On another note, so far depolarizing channels have been considered on the photons, which are not a very physically realistic approach. Another branch related to this topic would be to derive a physically motivated error model based, for example, on the tree generation structure, which clearly points out to a correlated error model. One could go even further and try to find a fault-tolerant tree-cluster generation. From a correlated error model, one would expect that the re-encoding procedure would be more prone to map to logical

errors, while the stabilizer measurements would be no more affected than they are with a simple error model as the depolarizing noise.

All in all, we hope that this thesis will help with laying the groundwork for the fault-tolerant one-way quantum repeater using code concatenation, including a simulation framework to further help with analyzing the behaviour of tree-code based quantum-repeater chains.

References

[1] Frank Arute et al. "Quantum supremacy using a programmable superconducting processor". en. In: *Nature* 574.7779 (Oct. 2019). Number: 7779 Publisher: Nature Publishing Group, pp. 505–510. ISSN: 1476-4687. DOI: 10.1038/s41586-019-1666-5. URL: http://www.nature.com/articles/s41586-019-1666-5.

[2] *Quantum Technology — The future is Quantum*. en-US. URL: https://qt.eu/.

[3] kugleta. *What is Horizon 2020?* en. Text. Oct. 2013. URL: https://ec.europa.eu/programmes/horizon2020//en/what-horizon-2020.

[4] *Home*. en-US. URL: https://quantum-internet.team/.

[5] *QIA - Quantum Internet Alliance - European Quantum Internet Alliance*. en-US. URL: https://qt.eu/about-quantum-flagship/projects/european-quantum-internet-alliance/.

[6] Stephanie Wehner, David Elkouss, and Ronald Hanson. "Quantum internet: A vision for the road ahead". en. In: *Science* 362.6412 (Oct. 2018). Publisher: American Association for the Advancement of Science Section: Review. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.aam9288. URL: https://science-sciencemag-org.tudelft.idm.oclc.org/content/362/6412/eaam9288.

[7] Charles H. Bennett and Gilles Brassard. "Quantum cryptography: Public key distribution and coin tossing". In: *Theoretical Computer Science* 560 (Dec. 2014). arXiv: 2003.06557, pp. 7–11. ISSN: 03043975. DOI: 10.1016/j.tcs.2014.05.025. URL: http://arxiv.org/abs/2003.06557.

[8] Peter Kómár et al. "A quantum network of clocks". In: *Nature Physics* 10.8 (Aug. 2014). arXiv: 1310.6045, pp. 582–587. ISSN: 1745-2473, 1745-2481. DOI: 10.1038/nphys3000. URL: http://arxiv.org/abs/1310.6045.

[9] Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi. "Universal blind quantum computation". In: *2009 50th Annual IEEE Symposium on Foundations of Computer Science* (Oct. 2009). arXiv: 0807.4154, pp. 517–526. DOI: 10.1109/FOCS.2009.36. URL: http://arxiv.org/abs/0807.4154.

[10] Thomas B. Bahder. "Quantum Positioning System". In: *arXiv:quant-ph/0406126* (June 2004). arXiv: quant-ph/0406126. URL: http://arxiv.org/abs/quant-ph/0406126.

[11] W. K. Wootters and W. H. Zurek. "A single quantum cannot be cloned". en. In: *Nature* 299.5886 (Oct. 1982). Number: 5886 Publisher: Nature Publishing Group, pp. 802–803. ISSN: 1476-4687. DOI: 10.1038/299802a0. URL: http://www.nature.com/articles/299802a0.

[12] H.-J. Briegel et al. “Quantum Repeaters: The Role of Imperfect Local Operations in Quantum Communication”. In: *Physical Review Letters* 81.26 (Dec. 1998). Publisher: American Physical Society, pp. 5932–5935. DOI: 10.1103/PhysRevLett.81.5932. URL: https://link.aps.org/doi/10.1103/PhysRevLett.81.5932.

[13] W. J. Munro et al. “Quantum communication without the necessity of quantum memories”. en. In: *Nature Photonics* 6.11 (Nov. 2012). Number: 11 Publisher: Nature Publishing Group, pp. 777–781. ISSN: 1749-4893. DOI: 10.1038/nphoton.2012.243. URL: http://www.nature.com/articles/nphoton.2012.243.

[14] Sreraman Muralidharan et al. “Ultrafast and Fault-Tolerant Quantum Communication across Long Distances”. In: *Physical Review Letters* 112.25 (June 2014). Publisher: American Physical Society, p. 250501. DOI: 10.1103/PhysRevLett.112.250501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.112.250501.

[15] Johannes Borregaard et al. “One-way quantum repeater based on near-deterministic photon-emitter interfaces”. In: *Physical Review X* 10.2 (June 2020). arXiv: 1907.05101, p. 021071. ISSN: 2160-3308. DOI: 10.1103/PhysRevX.10.021071. URL: http://arxiv.org/abs/1907.05101.

[16] Nicolas Gisin and Rob Thew. “Quantum Communication”. In: *Nature Photonics* 1.3 (Mar. 2007). arXiv: quant-ph/0703255, pp. 165–171. ISSN: 1749-4885, 1749-4893. DOI: 10.1038/nphoton.2007.22. URL: http://arxiv.org/abs/quant-ph/0703255.

[17] T. D. Ladd et al. “Quantum computers”. en. In: *Nature* 464.7285 (Mar. 2010). Number: 7285 Publisher: Nature Publishing Group, pp. 45–53. ISSN: 1476-4687. DOI: 10.1038/nature08812. URL: http://www.nature.com/articles/nature08812.

[18] Nicolas Gisin et al. “Quantum cryptography”. In: *Reviews of Modern Physics* 74.1 (Mar. 2002). Publisher: American Physical Society, pp. 145–195. DOI: 10.1103/RevModPhys.74.145. URL: https://link.aps.org/doi/10.1103/RevModPhys.74.145.

[19] H. J. Kimble. “The quantum internet”. en. In: *Nature* 453.7198 (June 2008). Number: 7198 Publisher: Nature Publishing Group, pp. 1023–1030. ISSN: 1476-4687. DOI: 10.1038/nature07127. URL: http://www.nature.com/articles/nature07127.

[20] Zhen-Sheng Yuan et al. “Entangled photons and quantum communication”. en. In: *Physics Reports* 497.1 (Dec. 2010), pp. 1–40. ISSN: 0370-1573. DOI: 10.1016/j.physrep.2010.07.004. URL: https://www.sciencedirect.com/science/article/pii/S0370157310001833.

[21] L.-M. Duan et al. “Long-distance quantum communication with atomic ensembles and linear optics”. en. In: *Nature* 414.6862 (Nov. 2001). Number: 6862 Publisher: Nature Publishing Group, pp. 413–418. ISSN: 1476-4687. DOI: 10.1038/35106500. URL: http://www.nature.com/articles/35106500.

[22] J. S. Bell. “On the Einstein Podolsky Rosen paradox”. In: *Physics Physique Fizika* 1.3 (Nov. 1964). Publisher: American Physical Society, pp. 195–200. DOI: 10.1103/PhysicsPhysiqueFizika.1.195. URL: https://link.aps.org/doi/10.1103/PhysicsPhysiqueFizika.1.195.

[23] W. J. Munro et al. "Inside Quantum Repeaters". In: *IEEE Journal of Selected Topics in Quantum Electronics* 21.3 (May 2015). Conference Name: IEEE Journal of Selected Topics in Quantum Electronics, pp. 78–90. ISSN: 1558-4542. DOI: 10.1109/JSTQE.2015.2392076.

[24] Charles H. Bennett et al. "Purification of Noisy Entanglement and Faithful Teleportation via Noisy Channels". In: *Physical Review Letters* 76.5 (Jan. 1996). arXiv: quant-ph/9511027, pp. 722–725. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.76.722. URL: http://arxiv.org/abs/quant-ph/9511027.

[25] M. Żukowski et al. ""Event-ready-detectors" Bell experiment via entanglement swapping". en. In: *Physical Review Letters* 71.26 (Dec. 1993), pp. 4287–4290. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.71.4287. URL: https://link.aps.org/doi/10.1103/PhysRevLett.71.4287.

[26] Alexander M. Goebel et al. "Multistage Entanglement Swapping". In: *Physical Review Letters* 101.8 (Aug. 2008). arXiv: 0808.2972, p. 080403. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.101.080403. URL: http://arxiv.org/abs/0808.2972.

[27] F.D Rozpedek. *Building blocks of quantum repeater networks*. en. OCLC: 8144104098. 2019. ISBN: 978-94-6384-043-9.

[28] Michael M. Wolf, David Pérez-García, and Geza Giedke. "Quantum Capacities of Bosonic Channels". In: *Physical Review Letters* 98.13 (Mar. 2007). Publisher: American Physical Society, p. 130501. DOI: 10.1103/PhysRevLett.98.130501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.98.130501.

[29] Charles H. Bennett, David P. DiVincenzo, and John A. Smolin. "Capacities of Quantum Erasure Channels". In: *Physical Review Letters* 78.16 (Apr. 1997). arXiv: quant-ph/9701015, pp. 3217–3220. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.78.3217. URL: http://arxiv.org/abs/quant-ph/9701015.

[30] Koji Azuma, Kiyoshi Tamaki, and Hoi-Kwong Lo. "All-photonic quantum repeaters". en. In: *Nature Communications* 6.1 (Nov. 2015), p. 6787. ISSN: 2041-1723. DOI: 10.1038/ncomms7787. URL: http://www.nature.com/articles/ncomms7787.

[31] Austin G. Fowler et al. "Surface Code Quantum Communication". In: *Physical Review Letters* 104.18 (May 2010). Publisher: American Physical Society, p. 180503. DOI: 10.1103/PhysRevLett.104.180503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.104.180503.

[32] Michael A. Nielsen and Isaac L. Chuang. *Quantum computation and quantum information*. en. 10th anniversary ed. Cambridge ; New York: Cambridge University Press, 2010. ISBN: 978-1-107-00217-3.

[33] Gláucia Murta et al. "Key rates for quantum key distribution protocols with asymmetric noise". In: *Physical Review A* 101.6 (June 2020). arXiv: 2002.07305, p. 062321. ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.101.062321. URL: http://arxiv.org/abs/2002.07305.

[34] S. Pirandola et al. "Advances in Quantum Cryptography". In: *Advances in Optics and Photonics* 12.4 (Dec. 2020). arXiv: 1906.01645, p. 1012. ISSN: 1943-8206. DOI: 10.1364/AOP.361502. URL: http://arxiv.org/abs/1906.01645.

[35] Renato Renner. "Security of Quantum Key Distribution". In: *arXiv:quant-ph/0512258* (Jan. 2006). arXiv: quant-ph/0512258. URL: http://arxiv.org/abs/quant-ph/0512258.

[36] Venkatesan Guruswami. "Notes 1: Introduction, linear codes". en. In: (), p. 11.

[37] Keisuke Fujii. "Stabilizer Formalism and Its Applications". en. In: *Quantum Computation with Topological Codes: From Qubit to Topological Fault-Tolerance*. Ed. by Keisuke Fujii. SpringerBriefs in Mathematical Physics. Singapore: Springer, 2015, pp. 24–55. ISBN: 978-981-287-996-7. DOI: 10.1007/978-981-287-996-7_2. URL: https://doi.org/10.1007/978-981-287-996-7_2.

[38] Andrew M Steane. "A Tutorial on Quantum Error Correction". en. In: (), p. 24.

[39] Maarten Van den Nest, Jeroen Dehaene, and Bart De Moor. "The invariants of the local Clifford group". In: *Physical Review A* 71.2 (Feb. 2005). arXiv: quant-ph/0410035, p. 022310. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.71.022310. URL: http://arxiv.org/abs/quant-ph/0410035.

[40] Asher Peres. "Reversible logic and quantum computers". In: *Physical Review A* 32.6 (Dec. 1985). Publisher: American Physical Society, pp. 3266–3276. DOI: 10.1103/PhysRevA.32.3266. URL: https://link.aps.org/doi/10.1103/PhysRevA.32.3266.

[41] Peter W. Shor. "Scheme for reducing decoherence in quantum computer memory". In: *Physical Review A* 52.4 (Oct. 1995). Publisher: American Physical Society, R2493–R2496. DOI: 10.1103/PhysRevA.52.R2493. URL: https://link.aps.org/doi/10.1103/PhysRevA.52.R2493.

[42] Raymond Laflamme et al. "Perfect Quantum Error Correction Code". In: *arXiv:quant-ph/9602019* (Feb. 1996). arXiv: quant-ph/9602019. URL: http://arxiv.org/abs/quant-ph/9602019.

[43] Barbara M. Terhal. "Quantum Error Correction for Quantum Memories". In: *Reviews of Modern Physics* 87.2 (Apr. 2015). arXiv: 1302.3428, pp. 307–346. ISSN: 0034-6861, 1539-0756. DOI: 10.1103/RevModPhys.87.307. URL: http://arxiv.org/abs/1302.3428.

[44] M. Hein et al. "Entanglement in Graph States and its Applications". In: *arXiv:quant-ph/0602096* (Feb. 2006). arXiv: quant-ph/0602096. URL: http://arxiv.org/abs/quant-ph/0602096.

[45] Maarten Van den Nest, Jeroen Dehaene, and Bart De Moor. "Graphical description of the action of local Clifford transformations on graph states". en. In: *Physical Review A* 69.2 (Feb. 2004), p. 022316. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.69.022316. URL: https://link.aps.org/doi/10.1103/PhysRevA.69.022316.

[46] D. Schlingemann and R. F. Werner. "Quantum error-correcting codes associated with graphs". en. In: *Physical Review A* 65.1 (Dec. 2001), p. 012308. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.65.012308. URL: https://link.aps.org/doi/10.1103/PhysRevA.65.012308.

[47] Michael A. Nielsen. "Cluster-state quantum computation". In: *Reports on Mathematical Physics* 57.1 (Feb. 2006). arXiv: quant-ph/0504097, pp. 147–161. ISSN: 00344877. DOI: 10.1016/S0034-4877(06)80014-5. URL: http://arxiv.org/abs/quant-ph/0504097.

[48] Mihir Pant et al. “Rate-distance tradeoff and resource costs for all-optical quantum repeaters”. en. In: *Physical Review A* 95.1 (Jan. 2017), p. 012304. ISSN: 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.95.012304. URL: https://link.aps.org/doi/10.1103/PhysRevA.95.012304.

[49] Michael Varnava, Daniel E. Browne, and Terry Rudolph. “Loss Tolerance in One-Way Quantum Computation via Counterfactual Error Correction”. In: *Physical Review Letters* 97.12 (Sept. 2006). Publisher: American Physical Society, p. 120501. DOI: 10.1103/PhysRevLett.97.120501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.97.120501.

[50] *NetSquid – The Network Simulator for Quantum Information using Discrete events*. en-US. URL: https://netsquid.org/.

[51] Tim Coopmans et al. “NetSquid, a discrete-event simulation platform for quantum networks”. In: *arXiv:2010.12535 [quant-ph]* (Jan. 2021). arXiv: 2010.12535. URL: http://arxiv.org/abs/2010.12535.

[52] Simon Anders and Hans J. Briegel. “Fast simulation of stabilizer circuits using a graph state representation”. In: *Physical Review A* 73.2 (Feb. 2006). arXiv: quant-ph/0504117, p. 022334. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.73.022334. URL: http://arxiv.org/abs/quant-ph/0504117.

[53] *QINC Wehner / NetSquid Snippets / NetSquid-SimulationTools*. en. URL: https://gitlab.tudelft.nl/qinc-wehner/netsquid-snippets/netsquid-simulationtools.

[54] *Snippets – NetSquid*. en-US. URL: https://netsquid.org/snippets/.

[55] Austin G. Fowler et al. “Surface codes: Towards practical large-scale quantum computation”. In: *Physical Review A* 86.3 (Sept. 2012). arXiv: 1208.0928, p. 032324. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.86.032324. URL: http://arxiv.org/abs/1208.0928.

[56] P. Baireuther et al. “Machine-learning-assisted correction of correlated qubit errors in a topological code”. In: *Quantum* 2 (Jan. 2018). arXiv: 1705.07855, p. 48. ISSN: 2521-327X. DOI: 10.22331/q-2018-01-29-48. URL: http://arxiv.org/abs/1705.07855.

[57] Jack Edmonds. “Paths, Trees, and Flowers”. en. In: *Canadian Journal of Mathematics* 17 (1965). Publisher: Cambridge University Press, pp. 449–467. ISSN: 0008-414X, 1496-4279. DOI: 10.4153/CJM-1965-045-4. URL: http://www.cambridge.org/core/journals/canadian-journal-of-mathematics/article/paths-trees-and-flowers/08B492B72322C4130AE800C0610E0E21.

[58] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. “Recurrent Neural Network Regularization”. In: *arXiv:1409.2329 [cs]* (Feb. 2015). arXiv: 1409.2329. URL: http://arxiv.org/abs/1409.2329.