TECHNISCHE UNIVERSITÄT MÜNCHEN

LUDWIG-MAXIMILIANS UNIVERSITÄT MÜNCHEN

Elite Master Program

Theoretical and Mathematical Physics

MASTER THESIS

General Quantum Error Correction for MERA Codes

Candidate:
FRANCESCO BATTISTEL

Supervisor:
Prof. ROBERT KÖNIG

Academic Year 2016/2017

Declaration of Authorship

I hereby certify that this thesis has been composed by me and is based on my own work, unless stated otherwise. No other person's work has been used without due acknowledgement in this thesis. All references and verbatim extracts have been quoted, and all sources of information, including graphs and data sets, have been specifically acknowledged.

Date: Signature:

Acknowledgements

First I want to thank my advisor Robert König for all the support he gave me. He discussed with me any problem I had and he gave me very useful suggestions during the course of all this work.

I also want to thank the research groups of Robert König and Michael Wolf, who hosted me during these months. Thanks also to Motohisa Fukuda who kindly accepted to be my second examiner at the defence.

A special thanks goes to my family who has always helped and supported me.

Contents

Introduction

There is a great effort nowadays to build a working quantum computer. One of the main practical limitations, however, comes from the fact that quantum states are very susceptible and sensitive to noise (thermal noise, preparation or measurement errors, photon loss...). The purpose of the theory of quantum error correction is to define a framework and the tools needed to passively or actively suppress the errors, at any stage of a computation. In this work we will model the noise by a *quantum channel* (see appendix A), which is well suited for many of the experimental conditions mentioned above.

The general idea of quantum error correction is to take "logical" states of interest on a quantum system and encode them into "physical" states of a larger one, in order to make their information-content somehow redundant and, hopefully, more easily protectable from the noise. The encoder (definition 1.1.2) is represented by an isometric quantum channel, mapping the logical states to states supported on a subspace of the physical Hilbert space, which is called the *code subspace* or *quantum error correcting code* (definition 1.1.1). Then, the task of quantum error correction is to find another quantum channel, called the *recovery map* (definition 1.1.3), which inverts the noise channel on the code subspace and gives back the original state. Whenever this is not exactly possible, it is still acceptable for most applications to recover a state which is only close to the original one, in terms of some measure. In this work we use the *worst-case entanglement fidelity* (definition 1.3.2) studied by Bény and Oreshkov [1], which leads to the so-called "approximate Knill-Laflamme conditions" (theorem 1.3.1).

An important class of quantum error correcting codes is the one of stabilizer codes (section 1.5). The *toric code* [2, 3] is an example showing many convenient and interesting properties: all operations can be performed locally, the recovery algorithm (definition 1.5.7) can be realized efficiently (i.e. in polynomial time) and there is a threshold probability (definition 1.5.8). The latter means that the probability for an uncorrectable error to happen goes to zero in the limit of large system sizes, as long as the physical error rate per single-qubit is below the threshold. However, the toric code requires interactions between the qubits which are not easy to realise, and the quest for better quantum error correcting codes is an active research topic.

In this work we study codes whose corresponding encoder has the structure of a tensor network called MERA (Multi-scale Entanglement Renormalization

Ansatz) [4, 5], which is made up of levels organised from bottom to top. In condensed matter applications, MERA usually represents a single state and it is used as a *coarse-graining* transformation from a lattice at the bottom level to smaller lattices at higher levels (see section 2.2). For our purposes, we consider MERAs representing entire subspaces, instead of a single state, and we consider the top and bottom levels to be the logical and physical ones, respectively. Therefore, for us MERA is just a quantum circuit with a peculiar local structure, serving as an encoder for input states at the top (see also section 2.2). A relevant example is again the toric code, which was later found to admit a MERA construction [6].

Lower bounds on the error-correcting capabilities of MERA codes have been recently derived by Kim & Kastoryano [7], in the case of erasure noise. Erasures are simply errors whose location is known, which is a precious information to construct a recovery map. However, such a knowledge is not generally available in a realistic situation. Our motivation for this thesis is then twofold: on the one hand, we would like to generalise those bounds to the case of more general noise models (it is not straightforward because we are considering *approximate* quantum error correction, see remark 4.0.1), and on the other hand we look for new examples of codes showing interesting properties, like the toric code, but starting from a MERA construction instead of building it in a second step (see section 3.5).

In chapters 1 and 2 we give an introduction to quantum error correction and MERA, respectively. In chapter 3 we first present the assumptions and the results of Kim & Kastoryano, identifying and correcting a number of formal details. Second, starting from the intuition we develop by studying the Ising chain and the toric code (section 3.4), we use numerics to look for stabilizer MERA codes with powerful capabilities: we find examples (see figure 3.5.2) for which we have evidence of a pretty high (19%) threshold probability w.r.t. Pauli noise and the smallest-weight recovery algorithm, with the drawback that such codes involve non-local interactions between the qubits. We did not manage to construct an efficient recovery algorithm admitting a threshold probability, but the possibility that it exists is still open.

In chapter 4 we investigate whether for a generic MERA code it is possible to guarantee correctability against arbitrary noise models. In the case in which the given code satisfies the K&K assumptions and the density-matrix assumption (definitions 3.1.4 and 4.2.2), we prove (theorem 4.2.1) that, if the Kraus operators ("errors") of a given channel are non-trivially supported on a number of qubits which is upper-bounded by a constant independent of the system size, then such a noise channel is approximately correctable (definition 1.3.3) with any desired accuracy if the system size is sufficiently large. We could not extend these results further, because of some limitations in our proof coming from the structure of the MERA itself. However, the existence of a threshold probability would mean that for stabilizer MERA codes and Pauli noise it is possible to correct arbitrary errors (not just erasures) supported on a *linear* number of sites, in the limit of large system sizes.

Chapter 1

Quantum Error Correction

In this chapter we review the main ideas of the theory of quantum error correction. More comprehensive books or lecture notes dealing with this subject are [8, 9, 10, 11]. Most of the results that will follow in this chapter are discussed in those references. We also cite the original work whenever we could find it. Here in the first few paragraphs we give an informal introduction and then in section 1.1 we start to define more precisely the concepts that we will need in the rest of this thesis.

Quantum information and quantum computing are very active topics of research nowadays. The main reasons are two: on the one hand, there are quantum algorithms, like the one invented by Shor for integer factorization [12], which are able to outperform any (known) classical one, and on the other hand, it is believed that quantum chemistry and material science will greatly benefit from the possibility of simulating quantum systems on a quantum hardware, as it was originally proposed by Feynman.

Ideally, a quantum computer operates on a quantum system S with a sequence of quantum gates, and the final state is appropriately measured. One of the main practical problems is represented by decoherence, which is the phenomenon for which S interacts with the environment in an uncontrolled way, such that the final state of the system is not the one that we would expect in the ideal case. Notice that to measure the system we must make it interact with our measurement device, which is part of the environment for S, but this is done in some controlled way. There is actually a second source of errors in the computation, which comes from the fact that quantum gates are unitary operators depending on continuous parameters, so that realising them in a perfect way is hopeless. Moreover, the preparation of the state or the measurement process are not error-free in general. However, it turns out that we can use the same formalism and the same concepts to correct all those types of errors, so we do not have to care too much about their origin.

We will model errors by a completely positive trace-preserving (CPTP) linear map or *quantum channel* $\mathcal{N}$ with Kraus operators $\{E_i\}$ (definition A.0.1):

$$\mathcal{N}(\rho) = \sum_i E_i \rho E_i^\dagger, \quad \sum_i E_i^\dagger E_i = \mathbb{1} \tag{1.1}$$

where ρ is the initial state of the system and $\mathcal{N}(\rho)$ is the state after the noise has disturbed it. It seems natural to view the noise as some process happening continuously in time, but here there is no concept of time involved, apparently. The idea behind this model is that we look at the evolution of our system after some time Δt, and $\mathcal{N}$ represents the cumulative action of decoherence during such a time interval. Moreover, every Δt we perform some error correcting procedure, which will succeed if the rate at which errors accumulate is lower than the rate at which we are able to correct them. Ultimately, Δt is limited by our experimental capabilities. As long as we perform quantum computations which take a small $t \ll \Delta t$, we do not have to care too much about errors because there will be only few or none of them, but if we want to build larger and larger quantum computers and perform longer and longer computations (which roughly corresponds to solving really complex and interesting problems), the importance of quantum error correction will grow more and more, both at the theoretical level of finding better procedures and at the experimental level of realising them in practice.

1.1 Quantum error correcting codes

We want to make clearer what we mean by quantum error correction (QEC) and errors. First we may define a recovery map $\mathcal{R}$ as a quantum channel such that $\mathcal{R} \circ \mathcal{N} = \mathrm{id}$, i.e. a quantum channel which reverses the action of the noise channel. However, the channels which are invertible are only the isometric ones (for a discussion see chapter 3 of [8]), whereas $\mathcal{N}$ it is not, in general. Physically, this means that $\mathcal{N}$ "degrades" some of the information which is contained in a state, so that the states that we will be able to recover will be those for which their information-content is somehow redundant. Hence, we cannot expect to be able to protect any state.

Definition 1.1.1 (Quantum Error Correcting Code)**.** Given a quantum system with Hilbert space $\mathcal{H}_S$, we define a quantum error correcting code (QECC) as a subspace $\mathcal{C}_S \subseteq \mathcal{H}_S$. We call it the *code subspace* and we refer to states in $\mathcal{C}_S$ as *codewords* or *code states*.

Remark 1.1.1. There exists also subsystem codes (see [9] for example) where the information is not encoded in a subspace but in a subsystem $\mathcal{H}_A$, i.e., supposing that $\mathcal{H}_S$ has a decomposition $\mathcal{H}_S = \mathcal{H}_A \otimes \mathcal{H}_B \oplus \mathcal{K}$ for some Hilbert spaces $\mathcal{H}_A, \mathcal{H}_B$ and $\mathcal{K}$, then $\mathcal{H}_A$ is called a subsystem code. In the following we will consider only codes as defined in 1.1.1.

Definition 1.1.2 (Encoder)**.** Given another quantum system with Hilbert space $\mathcal{H}_{S'}$, $\dim \mathcal{H}_{S'} \leq \dim \mathcal{H}_S$, we define an encoder as an isometric quantum channel $\mathcal{E} \colon \mathcal{B}(\mathcal{H}_{S'}) \to \mathcal{B}(\mathcal{H}_S)$, i.e. $\mathcal{E}(\rho_{S'}) = W \rho_{S'} W^\dagger \ \forall \rho_{S'} \in \mathcal{B}(\mathcal{H}_{S'})$, where $W \colon \mathcal{H}_{S'} \to \mathcal{H}_S$ is an isometry ($W^\dagger W = \mathbb{1}_{\mathcal{H}_{S'}}$).

Note that the image of $\mathcal{E}$ is a QECC.

Definition 1.1.3 (Exact QECC and recovery map)**.** We say that a QECC is *exact* w.r.t. a quantum channel $\mathcal{N}$ (called the *noise channel*) if there exists another quantum channel $\mathcal{R}$, that we call the *recovery map*, such that

$$\mathcal{R}(\mathcal{N}(\rho)) = \rho \tag{1.2}$$

for every code state ρ supported in $\mathcal{C}_S$, or, equivalently, such that

$$\mathcal{R} \circ \mathcal{N} \circ \mathcal{E} = \mathrm{id}_{\mathcal{H}_{S'}} \tag{1.3}$$

where $\mathcal{E} \colon \mathcal{B}(\mathcal{H}_{S'}) \to \mathcal{B}(\mathcal{H}_S)$ is an encoder whose image is the given QECC. We will also say that the recovery map corrects against $\mathcal{N}$ (over the code subspace).

Remark 1.1.2. Note that talking about $\mathcal{E}$ or $\mathcal{C}_S$ is basically the same thing, so one can refer directly to the subspace and do not specify what is the encoder, or alternatively one can specify the encoder in order to implicitly characterize the subspace (this is what we will do with MERA in the following chapters). Moreover, the reason why we consider only isometric encoders is that a non-isometric channel would map at least some orthogonal states to non-orthogonal ones, but this means making them irreparably less distinguishable. We expect then that the performance of such a code cannot be better than using an isometric encoder.

Remark 1.1.3. In some cases one may relax the above condition requiring that $\mathcal{R} \circ \mathcal{N} \circ \mathcal{E} \propto \mathrm{id}_{\mathcal{H}_{S'}}$, i.e. one can take $\mathcal{R}$ (or also $\mathcal{N}$) trace non-increasing instead of trace preserving. It is possible to justify this choice by the following. If $\rho = \mathcal{R}(\mathcal{N}(\rho)) = \mathcal{R}(\sum_i E_i \rho E_i^\dagger) = \sum_i \mathcal{R}(E_i \rho E_i^\dagger)$ for ρ supported in $\mathcal{C}_S$, then it is easy to show that $\mathcal{R}(E_i \rho E_i^\dagger) \propto \rho$. The claim follows from the fact that $\sum_i |\psi_i\rangle \langle\psi_i| = |\psi\rangle \langle\psi|$ for $\{|\psi_i\rangle\}$ sub-normalized non-zero vectors if and only if $|\psi_i\rangle \propto |\psi\rangle \ \forall i$, which holds because the $\{|\psi_i\rangle \langle\psi_i|\}$ are positive operators (it is a corollary of proposition 2.4 in [13]). So if one excludes some of the E_i, then $\sum_i E_i^\dagger E_i \leq 1$ and $\mathcal{R} \circ \mathcal{N} \circ \mathcal{E} \propto \mathrm{id}_{\mathcal{H}_{S'}}$.

Suppose that we want to find a recovery map correcting against the channel $\mathcal{N}(\rho) = \sum_i E_i \rho E_i^\dagger$ with Stinespring dilation $U_{\mathcal{N}} \colon |\psi\rangle \otimes |0\rangle \mapsto \sum_i E_i |\psi\rangle \otimes |i\rangle$ (definition A.0.1). This second form *may* suggest some operational way in which we can realise the recovery map: if we have control over the environment, in the sense that we are able to perform measurements on it, then when we measure in the $\{|i\rangle\}$ basis we get the outcome i with probability $\|E_i |\psi\rangle\|^2$ and the state after the measurement is $E_i |\psi\rangle \,/\, \|E_i |\psi\rangle\|$. Because we are assuming that the recovery map works, this means that E_i has to be invertible on the code subspace (in section 1.2 we will make this discussion more precise and formal). However, the procedure just described is not a recovery map in the sense of QECC: $\mathcal{R}$ in the definition is a quantum channel acting on the system *only*. We are not allowed to act also on the environment! We always assume this last condition because this is what makes experimentally sense: we have some sources of noise that we do not control (otherwise we

could just prevent the noise from occurring) and so we have to invent clever methods to understand what is i acting only on the system (or to restrict the possible values of i to some subset), but without destroying the encoded state. Indeed, this is one of the hard tasks that one should accomplish while designing a recovery map for a QECC, and we will see in section 1.2 under which conditions it is possible.

Parameters characterizing a QECC

In quantum error correction, one usually assumes some error model, that is, a noise channel $\mathcal{N}$ with some type of general structure, and then one wants to find the best combination of encoder and recovery map able to protect against $\mathcal{N}$. This is a double optimization problem, which is pretty hard in general, so that one usually makes an ansatz about what can be a good encoder $\mathcal{E}$, or some class of encoders, fixes that and then tries to optimize only over recovery maps, hoping at the same time that the ultimately chosen recovery map will perform well also again other noise channels.

Generally speaking, it is desirable to define some parameters characterizing a code and its ability to protect from errors. Usually one consider systems where the basic degrees of freedom are qubits, similarly to the classical case in which one can focus on binary codes where the smallest units of information are bits. We will do that throughout all this chapter and the rest of this thesis, unless otherwise stated. Recall that a system made up of n qubits has a Hilbert space $\mathcal{H}_S = (\mathbb{C}^2)^{\otimes n}$ with dimension $\dim \mathcal{H}_S = 2^n$.

Definition 1.1.4. We characterize a QECC $\mathcal{C}_S$ by a set of parameters $[[n, k, d]]$, where:

- $n = \log_2 \dim \mathcal{H}_S$ is called the number of *physical qubits*. We will refer to n also as the *system size*;
- $k = \log_2 \dim \mathcal{H}_{S'} = \log_2 \dim \mathcal{C}_S$ is the number of *logical qubits*. The ratio between logical and physical qubits is called the *rate* of the code;
- d is the *distance* of the code, defined as the smallest number of physical qubits which can support an operator able to map a codeword to another one (a *logical operator*). More formally:
  - $O \in \mathcal{B}(\mathcal{H}_S)$ is a logical operator if $\Pi O \Pi \neq \lambda \Pi$ for any $\lambda \in \mathbb{C}$, where Π is the projector such that $\Pi(\mathcal{H}_S) = \mathcal{C}_S$.
    Note that if $[O, \Pi] = \mathbf{0}$, then O is a logical operator according to our definition if it is different from the identity on the code subspace;
  - We say that a non-zero operator O has an *identity tensor product factor* at position j if it can be written as

$$O = \tilde{O}_{S\setminus\{j\}} \otimes I_j \tag{1.4}$$

where $\tilde{O}_{S\setminus\{j\}}$ is an operator acting on all the qubits except for the j-th one, otherwise we say that O has a *non-identity tensor product factor* at position j. Define

$$S_O = \{j \,|\, O \text{ has a non-identity tensor product factor at position } j\}. \tag{1.5}$$

We call it the *set of the faulty locations* of O. Then the distance d is defined as

$$d := \min_{\{O \in \mathcal{B}((\mathbb{C}^2)^{\otimes n}) \,|\, \Pi O \Pi \not\propto \Pi\}} |S_O| \,. \tag{1.6}$$

Lemma 1.1.1. *Let $\{B_i\}_{i=1}^4$ be a basis of $\mathcal{B}(\mathbb{C}^2)$ where $B_1 = I$, and let $\mathfrak{B}$ be a basis of $\mathcal{B}((\mathbb{C}^2)^{\otimes n})$ defined as $\mathfrak{B} := \{B_{i_1} \otimes \ldots \otimes B_{i_n}\}_{i_1,\ldots,i_n=1}^4$. Then*

$$d = \min_{\{B \in \mathfrak{B} | \Pi B \Pi \not\propto \Pi\}} |S_B| \,. \tag{1.7}$$

Therefore, the minimization in (1.6) *can be computed just on a finite set.*

Proof. Writing a generic operator $O \in \mathcal{B}((\mathbb{C}^2)^{\otimes n})$ on the basis $\mathfrak{B}$,

$$O = \sum_{i_1,\ldots,i_n=1}^{4} c_{i_1,\ldots,i_n} B_{i_1} \otimes \ldots \otimes B_{i_n}, \tag{1.8}$$

it is clear that $\left|S_{B_{i_1}\otimes\ldots\otimes B_{i_n}}\right| \leq |S_O| \;\; \forall i_1, \ldots, i_n$ such that $c_{i_1,\ldots,i_n} \neq 0$. The condition $\Pi O \Pi \not\propto \Pi$ holds only if there exists at least one term in the previous sum with $c_{j_1,\ldots,j_n} \neq 0$ such that $\Pi(B_{j_1} \otimes \ldots \otimes B_{j_n})\Pi \not\propto \Pi$. Therefore, $\forall O \in \mathcal{B}((\mathbb{C}^2)^{\otimes n})$ such that $\Pi O \Pi \not\propto \Pi$, there exists at least one $B \in \mathfrak{B}$ such that $|S_B| \leq |S_O|$. It follows that we can restrict the minimization in (1.6) just to elements in $\mathfrak{B}$. □

Remark 1.1.4. Usually in QEC one does not focus on a single "special" value of n, but one would like to construct a family of codes with the same structure for many values of n, where k and d are understood as functions of n. In particular, we would like a "good" code to have

1. $\lim_{n\to\infty} \frac{k(n)}{n} > 0$ i.e. *finite limiting rate*
2. $\lim_{n\to\infty} \frac{d(n)}{n} > 0$ i.e. *finite limiting distance*
3. efficient *recovery algorithm* (see definition 1.5.7).

The first condition may even be dropped in many cases, the second one is actually too strong and not really necessary to have a good code (with theorem 1.5.3 we will see how we can relax it considering an "effective" distance), whereas by the third one we mean that the classical algorithm which computes the recovery map can be executed in polynomial time (in section 1.5 we will discuss this issue more broadly). However, it is very difficult to get codes with all those three properties, in particular with the last two.

Notice that when a logical operator acts on an encoded state, we have no way of detecting if that happened or not, even when a codeword is mapped to an orthogonal one. This may sound like a false statement, because orthogonal states are precisely the only states for which we can design a procedure to perfectly distinguish among them. The point is that in QEC we assume that we do not know what the encoded state is: for example, it may be the result of an encoded quantum computation. Because we know which gates we are implementing, we may in principle follow the evolution of the state we originally prepared, using a classical computer. However, the reason why we would like to build a quantum computer is precisely to avoid to simulate classically the evolution, which is too hard for the problems of interest. Hence, if we end up with a codeword we may be happy with this fact, but if it is the wrong one and we cannot realise it, then we would draw the wrong conclusion from our computation.

Code distance and i.i.d. noise

The distance is a concept that actually makes sense as a characterization of how good a code is (and we will show that even in such a case it is quite a bad and rough measure) only when we consider what we will call a

Definition 1.1.5 (Single-qubit i.i.d. error model)**.** If the noise channel $\mathcal{N}$ acting on a system made up of n qubits factorizes into $\mathcal{N} = \mathcal{N}_1^{\otimes n}$, where $\mathcal{N}_1$ is a single-qubit channel, we say that we are considering a single-qubit i.i.d. error model.

Notice that the Hilbert space $\mathcal{H}$ for n qubits is $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$ and we will consider this case throughout the following.

In the above model we are assuming that each qubit undergoes independently the same type of noise process, which means that we assume that each qubit is interacting with a different environment, or, put it in a different way, that each qubit is interacting with uncoupled parts of the same environment. We may say that the environment acts *locally* whenever the interaction is described by a single-qubit i.i.d. error model.

Ideally, one would like to protect against any kind of noise channel. We will see in section 1.2 that once a recovery map is able to correct against a noise channel $\mathcal{N}$ (over a code subspace), then it is able to correct against all the channels (over the same subspace) whose Kraus operators are linear combinations of the Kraus operators of $\mathcal{N}$. We will say that $\mathcal{R}$ is able to correct the *error subspace* associated with $\mathcal{N}$:

Definition 1.1.6 (Error subspace)**.** Given a noise channel $\mathcal{N}$ with Kraus operators $\{E_i\}$, $E_i \in \mathcal{B}(\mathcal{H})$ $\forall i$, the error subspace associated to $\mathcal{N}$ is given by $\text{span}\{E_i\} \subseteq \mathcal{B}(\mathcal{H})$. We call *error* any operator in the error subspace.

Therefore, by linearity, if we find a recovery map able to correct against a channel whose Kraus operators form a basis of $\mathcal{B}((\mathbb{C}^2)^{\otimes n}) \simeq \mathcal{M}_{2^n}(\mathbb{C})$ (the

vector space of $2^n \times 2^n$ matrices), then we have found a recovery map able to correct against any kind of noise on the given code subspace. For a system of 1 qubit, a basis of $\mathcal{B}(\mathbb{C}^2)$ is given by the Pauli matrices X (*bit flip* error), Z (*phase flip* error) and Y ($= iXZ$, i.e. a bit flip and a phase flip error), plus the identity I. If we have n qubits, then a basis of $\mathcal{B}((\mathbb{C}^2)^{\otimes n})$ is given by the set of all

Definition 1.1.7 (Pauli operators)**.** A Pauli operator over n qubits is an operator P that can be written as $P_1^{(1)} \otimes \cdots \otimes P_n^{(1)}$ where $P_i^{(1)} \in \{I, X, Y, Z\}$, $i = 1, \ldots, n$. We call $\mathcal{G}_n^+$ the set of all Pauli operators over n qubits.

We also define:

Definition 1.1.8 (Weight of a Pauli operator)**.** The *weight* $w(P)$ of a Pauli operator P the number of its non-identity tensor product factors.

Corollary 1.1.1. *The distance d of a QECC with code subspace projector Π is given by*

$$d = \min_{\{P \in \mathcal{G}_n^+ | \Pi P \Pi \not\propto \Pi\}} w(P). \tag{1.9}$$

Proof. Immediate from lemma 1.1.1. □

Remark 1.1.5. The Pauli operators do not play any special role in the definition of distance except for the fact that they form a basis of $\mathcal{B}((\mathbb{C}^2)^{\otimes n})$. We may use any other one instead of $\mathcal{G}_n^+$.

Definition 1.1.9 (Pauli noise)**.** We define a *Pauli channel* over n qubits as a single-qubit i.i.d. error model $\mathcal{N}_P = (\mathcal{N}_P^1)^{\otimes n}$ where $\mathcal{N}_P^1$ has a Kraus representation in which all Kraus operators are Pauli operators.

With the result of section 1.2 one can prove (see chapter 7 of [8] for a discussion):

Theorem 1.1.1. *Let F be a family of channels, $F = \{\mathcal{N}_i\}_{i=1}^N$, where each $\mathcal{N}_i$ can be represented by a single Pauli operator, i.e. $\mathcal{N}_i(\rho) = P_i \rho P_i$ for some Pauli operator P_i. Consider a QECC $\mathcal{C}$ with distance d. Then*

- *there exists a recovery map (definition 1.1.3) which is able to correct against any $\mathcal{N}_i$ (over $\mathcal{C}$) if $w(P_i) \leq \left\lfloor \frac{d-1}{2} \right\rfloor \ \forall i = 1 \ldots N$.*

Suppose now that $w(P_i) \leq d-1 \ \forall i = 1 \ldots N$. Assume that one channel among the $\{\mathcal{N}_i\}$ is applied to a given code state ρ of $\mathcal{C}$, chosen according to some probability distribution. Hence assume that the state has become $\mathcal{N}_j(\rho)$ for some j. Suppose that we do not know which $\mathcal{N}_j$ was applied. Then, independently of ρ, we have the following:

- *there exists a measurement which can unambiguously determine whether $\mathcal{N}_j(\rho) \equiv \rho$ or not;*

- *suppose that we are given the information about which are the non-identity tensor product factors of the Pauli operator P_j corresponding to $\mathcal{N}_j$, but without knowing if at each location there is an X, Y or Z. In terms of* (1.5)*, we are given the set S_{P_j} of the faulty locations of P_j. Then there exists a measurement and a recovery map, chosen depending on the measurement outcome and the above information, which corrects against $\mathcal{N}_j$.*

Single-qubit error probability

Here we want to define a concept of "error probability (or error rate) per single qubit" in the quantum channel framework. We can start with an example, which is known as the (completely) *depolarizing channel*, with Kraus operators

$$E_0 = \sqrt{1-p}\, I, \qquad E_1 = \sqrt{\frac{p}{3}}\, X, \qquad E_2 = \sqrt{\frac{p}{3}}\, Y, \qquad E_3 = \sqrt{\frac{p}{3}}\, Z \tag{1.10}$$

and with Stinespring dilation:

$$U\colon\ |\psi\rangle\otimes|0\rangle \mapsto \sqrt{1-p}\,|\psi\rangle\otimes|0\rangle + \sqrt{\frac{p}{3}}X\,|\psi\rangle\otimes|1\rangle + \sqrt{\frac{p}{3}}Y\,|\psi\rangle\otimes|2\rangle + \sqrt{\frac{p}{3}}Z\,|\psi\rangle\otimes|3\rangle\,. \tag{1.11}$$

In the interpretation that we gave previously, both if we would be able to measure the environment in the $\{|i\rangle\}_{i=0}^{3}$ basis or if we would be able to cleverly infer i from measurements on the system which do not disturb the encoded state (in this case we cannot really do the latter because we are considering just one qubit), then, according to Born's rule, it follows that we would find the state $|\psi\rangle$ with probability $1-p$, $X\,|\psi\rangle$ with probability $p/3$ and the same for Y and Z. It does not really matter how p is distributed among the possible errors, so we make the following definition:

Definition 1.1.10 (Single-qubit error probability)**.** Given a single-qubit channel $\mathcal{N}_1$, we say that there is a probability p of having an error if there exists a minimal (see theorem A.0.3 and definition A.0.3) Kraus representation $\{E_i\}$ of $\mathcal{N}_1$ such that $E_0 = \sqrt{1-p}\, I$. If $\mathcal{N} = \mathcal{N}_1^{\otimes n}$ we say that we have a probability p of error per single qubit.

Remark 1.1.6. Note that we choose a minimal representation in such a way that all the other Kraus operators are linearly independent of the identity, otherwise the definition would be ambiguous because for example we may have $E_0 = c_0\, I$ and $E_1 = c_1\, I$. Moreover, there are channels for which we *cannot* assign a value to p, for example the amplitude-damping channel (defined for example in [8]), because the identity is not in the linear span of its Kraus operators.

For $\mathcal{N} = \mathcal{N}_1^{\otimes n}$, the Kraus operators of $\mathcal{N}$ are all the possible tensor products of the Kraus operators of $\mathcal{N}_1$. For single-qubit Pauli noise (definition 1.1.9), with not-necessarily equal probabilities for X, Y, Z errors to

happen, those Kraus operators are all the Pauli operators for n qubits ("errors" in the following). More precisely, each Kraus operator is proportional to a Pauli operator P, where the proportionality is of the order $\mathcal{O}(p^t)$ and t is the weight of P. Hence, $\mathcal{N}$ contains errors with arbitrarily high weight. The proportionality constant can be viewed as the probability that P occurs and affects a code state, as we shall explain towards the end of section 1.2.

One may think that if $p \ll 1$, then p^2 is negligible and so the errors with weight 2 have a probability to occur which is also negligible. However, it depends on n, for fixed p, because the number of errors with weight 2 is $3^2\binom{n}{2}$, which may be very large in such a way that it can be very probable that some error of weight 2 occurs. Indeed, according to a binomial distribution the expected weight is np, and therefore most errors will have a weight which differs from np by at most a few standard deviations. If we consider a QECC with distance d and we fix also n, then how "good" such a code is depends a lot (but not completely) on the relationship between d and np. If $d \gg np$, then the probability that an error with weight $w \geq \lfloor (d-1)/2 \rfloor$ happens is (basically) negligible, and from theorem 1.1.1 it follows that there exists a recovery map (basically) correcting against $\mathcal{N}$, otherwise the performance of such a recovery map depends on the specific QECC. We will better quantify this discussion in section 1.5 in the case of stabilizer codes.

1.2 Knill-Laflamme conditions

At this point we would like to have a characterization of a correctable set of errors $\{E_i\}$, i.e., given some noise channel $\mathcal{N}$ with Kraus operators $\{E_i\}$, we would like to have necessary and sufficient conditions under which there exists a recovery map $\mathcal{R}$ such that $\mathcal{R}(\mathcal{N}(\rho)) = \rho$ for every state supported in some given code subspace $\mathcal{C}$.

Theorem 1.2.1 (Knill-Laflamme conditions [14])**.** *Given a noise channel $\mathcal{N}(\rho) = \sum_i E_i \rho E_i^\dagger$ and a QECC $\mathcal{C}$, necessary and sufficient conditions for the existence of a recovery map $\mathcal{R}$ correcting against $\mathcal{N}$ are given by:*

$$\Pi E_i^\dagger E_j \Pi = \lambda_{ij} \Pi \tag{1.12}$$

where Π is the projector onto $\mathcal{C}$ and $\lambda = (\lambda_{ij})_{ij}$ is a density matrix, $\lambda_{ij} \in \mathbb{C}$. If $\mathcal{N}$ is trace non-increasing, then λ is a positive operator with $\operatorname{Tr} \lambda \leq 1$.

Before proceeding to the proof, we notice that it follows from (1.12) that $\langle \psi | E_i^\dagger E_j | \psi \rangle = \lambda_{ij}$ for any $|\psi\rangle \in \mathcal{C}$, from which we see that on the r.h.s. there is no dependence on $|\psi\rangle$, which means that for each pair of errors E_i, E_j, the combination $E_i^\dagger E_j$ is a multiple of the identity on $\mathcal{C}$. If instead we pick a basis $\{|a\rangle\}$ of $\mathcal{C}$, then we observe from (1.12) that $\langle a | E_i^\dagger E_j | b \rangle = \lambda_{ij} \delta_{ab}$, which means that orthogonal codewords remain orthogonal under the action of the noise.

Proof. Necessity. Assume that there exists a recovery map $\mathcal{R}$ with Kraus operators $\{R_j\}$ and Stinespring dilation $U_{\mathcal{R}} \colon |\psi\rangle \otimes |0\rangle_A \mapsto \sum_j R_j |\psi\rangle \otimes |j\rangle_A$,

where A is some *ancilla* system (an environment that we control). Then correctability requires that

$$U_{\mathcal{R}}U_{\mathcal{N}}\colon\ |\psi\rangle\otimes|0\rangle_E\otimes|0\rangle_A \mapsto \sum_{ij} R_j E_i\,|\psi\rangle\otimes|i\rangle_E\otimes|j\rangle_A \tag{1.13}$$

$$\equiv \sum_{ij}\tilde{\lambda}_{ij}\,|\psi\rangle\otimes|i\rangle_E\otimes|j\rangle_A \tag{1.14}$$

$$= |\psi\rangle\otimes|\text{junk}\rangle_{EA} \tag{1.15}$$

for $|\psi\rangle\in\mathcal{C}$. We then have that

$$\Pi E_i^\dagger E_j\Pi = \Pi E_i^\dagger\Big(\sum_k R_k^\dagger R_k\Big)E_j\Pi \tag{1.16}$$

$$= \Pi\sum_k \tilde{\lambda}_{ik}^*\tilde{\lambda}_{jk}\Pi \equiv \lambda_{ij}\Pi. \tag{1.17}$$

The fact that λ is a density matrix is a simple fact which follows from the normalization of the Kraus operators and from taking the hermitian conjugate of (1.12).

Remark 1.2.1. Physically this means that by using some ancilla we move the entanglement between system and environment, induced by the noise channel, to entanglement between environment and ancilla. Note that, because entanglement corresponds to entanglement entropy looking at the marginal states, we can say that we are extracting entropy from the system and putting it into the ancilla. If we then want to reuse the same ancilla for a second round of QEC, we have to first reinitialise it, which by Landauer's principle costs energy (see chapter 1 of [8] for a discussion): error correction is not for free!

Sufficiency. Suppose now that (1.12) holds. Being λ a density matrix, it follows in particular that it is diagonalizable, which means that there exists a unitary U such that $U\lambda U^\dagger$ is diagonal, i.e.

$$(U\lambda U^\dagger)_{ij} = \sum_{kl} U_{ik}\lambda_{kl}U_{lj}^\dagger = \lambda_i\delta_{ij} \tag{1.18}$$

for some $\lambda_i\geq 0$ (being λ a positive operator). Using (1.12), we also have that

$$\sum_{kl} U_{ik}\lambda_{kl}U_{lj}^\dagger = \sum_{kl} U_{ik}\Pi E_k^\dagger E_l\Pi U_{lj}^\dagger = \Pi\Big(\sum_k U_{ik}E_k^\dagger\Big)\Big(\sum_l U_{lj}^\dagger E_l\Big)\Pi. \tag{1.19}$$

Defining $F_j \coloneqq \sum_l U_{lj}^\dagger E_l$ it is easy to see that they form another valid set of Kraus operators for $\mathcal{N}$ (see also lemma A.0.1). Hence the two above equations imply that (1.12) is equivalent to

$$\Pi F_i^\dagger F_j\Pi = \lambda_i\delta_{ij}\Pi. \tag{1.20}$$

From this expression it follows that:

- $\lambda_i \neq 0$ implies that there exists a unitary U_i such that $F_i|_{\mathcal{C}} = \sqrt{\lambda_i}\, U_i|_{\mathcal{C}}$, i.e. F_i behaves like a unitary on the code subspace, so we can correct it applying $U_i^\dagger$, and F_i occurs with probability λ_i.

- $\lambda_i = 0$ implies that $F_i|_{\mathcal{C}} = \mathbf{0}$, i.e. the error F_i never occurs (it corresponds to the previous case with probability of occurring equal to 0) and so we do not have to worry to correct it. Remark: F_i is generally non-zero on other subspaces.

Moreover, we have $0 = \langle\psi|F_i^\dagger F_j|\psi\rangle = \sqrt{\lambda_i\lambda_j}\,\langle\psi|U_i^\dagger U_j|\psi\rangle$ for $i \neq j$ and $|\psi\rangle \in \mathcal{C}$, which means that

- if λ is full rank, then $\Pi U_i^\dagger U_j \Pi = \delta_{ij}\,\Pi$, i.e. the $\{U_i\}$ map $\mathcal{C}$ to orthogonal subspaces $\{\mathcal{C}_i\}$ (orthogonal among them, and w.r.t. $\mathcal{C}$ itself if $U_0 = \mathbb{1}$, which is the case one often assumes, corresponding to some probability that no error occurs).

- if λ is not full rank, i.e. some of its eigenvalues are 0, ordering them in descending order $\{\lambda_0, \ldots, \lambda_k, 0, \ldots, 0\}$, we have that $\Pi U_i^\dagger U_j \Pi = \delta_{ij}\delta_{i>k}\,\Pi$, meaning that for $i > k$ the given expression is always $\mathbf{0}$ (not because for $i, j > k$ the two of unitaries map to orthogonal subspaces but because one of them acts as $\mathbf{0}$ on $\mathcal{C}$).

Based on the above observations, we can construct a recovery procedure: we first make a measurement which projects onto such a family of subspaces $\{\mathcal{C}_i\}$; because they are orthogonal, we can perfectly distinguish among them; finally, based on the measurement result i, we apply $U_i^\dagger$ and renormalise the result dividing by $\sqrt{\lambda_i}$. Explicitly, the quantum channel $\mathcal{R}$, $\mathcal{R}(\cdot) = \sum_j R_j(\cdot)R_j^\dagger + \bar{R}(\cdot)\bar{R}^\dagger$, doing what just described, is given by the following Kraus operators:

$$\begin{aligned} R_j &= \frac{1}{\sqrt{\lambda_j}}\Pi F_j^\dagger, \quad \lambda_j \neq 0 \\ \bar{R} &= \mathbb{1} - \sum_{j\colon \lambda_j \neq 0} \frac{1}{\lambda_j} F_j \Pi F_j^\dagger \end{aligned} \tag{1.21}$$

where $\bar{R}$ is just added to make $\mathcal{R}$ trace-preserving (it is the projector onto the orthogonal subspace to the span of the subspaces to which $\mathcal{C}$ is mapped by the errors). It is immediate to verify that the above operators form a valid set of Kraus operators and, using (1.12), that $\mathcal{R}(\mathcal{N}(\rho)) = \rho$ for all code states ρ. □

Definition 1.2.1. (Knill-Laflamme recovery map) Given a noise channel $\mathcal{N}$ satisfying the Knill-Laflamme conditions on a code subspace with projector Π, we call the *Knill-Laflamme recovery map* the quantum channel $\mathcal{R}$ described by the Kraus operators (1.21), where $\{F_j\}$ is a Kraus representation of $\mathcal{N}$ such that $\Pi F_i^\dagger F_j \Pi = \lambda_i \delta_{ij} \Pi$.

Remark 1.2.2. The Knill-Laflamme recovery map formalizes the idea that once we are able to correct some set of errors, then we can correct also all their

linear combinations, i.e. we can correct the full error subspace associated to $\mathcal{N}$, as we shall explain here.

Thanks to the Knill-Laflamme conditions, the Kraus operators of $\mathcal{R}$ have the following action:

$$R_j F_i \Pi = \delta_{ij} \Pi, \tag{1.22}$$

that is, the combination $R_j F_i$ is a multiple of the identity on the code subspace (in this case either 1 or 0). If we now consider an operator $\tilde{E} = \sum_i c_i F_i$ we have that

$$R_j \tilde{E} \Pi = R_j \sum_i c_i F_i \Pi = \sum_i c_i R_j F_i \Pi = c_j \Pi \tag{1.23}$$

so that $R_j \tilde{E}$ is a multiple of the identity on $\mathcal{C}$ (where c_j can take arbitrary values). Consider now a channel $\tilde{\mathcal{N}}$ with Kraus operators $\tilde{E}_k = \sum_i c_{ik} F_i$, i.e. a channel whose Kraus operators are taken from $\text{span}\{F_i\} = \text{span}\{E_i\}$, with the only requirement that $\sum_k \tilde{E}_k^\dagger \tilde{E}_k = \mathbb{1}$, which corresponds to $\sum_{ik} \lambda_k |c_{ik}|^2 = 1$. With this it is easy to see that

$$\mathcal{R} \circ \tilde{\mathcal{N}}(\rho) = \rho \tag{1.24}$$

for any code state. Hence, once $\mathcal{R}$ corrects against $\mathcal{N}$, then it also corrects against any $\tilde{\mathcal{N}}$ constructed as above.

Remark 1.2.3. The discussion after remark 1.1.3 in section 1.1 shows that if we are given the ability to perform measurements on the environment, then we can construct a recovery procedure for which the Knill-Laflamme conditions are not necessary, because in that case it is enough that the $\{E_i|_{\mathcal{C}}\}$ are unitaries but they do not have to map to orthogonal subspaces. The above theorem still holds, however, because that is a recovery map which is not acting *only* on the system, as required in the hypothesis.

We can prove necessity of the Knill-Laflamme conditions also using more general principles of quantum information (see [8] for a discussion). If the environment would be able to acquire information about the encoded state, it would be able to get a copy of it (at least some copies, asymptotically), but then this would violate the no-cloning theorem by the hypothesis of correctability on the system. Hence, we need that the density matrix ρ^E for the environment is independent of all states ρ supported on $\mathcal{C}$. Such ρ_E is given by the complementary channel $\widehat{\mathcal{N}}$ (definition A.0.4), as explained in appendix A:

$$\rho^E = \widehat{\mathcal{N}}(\rho). \tag{1.25}$$

Lemma 1.2.1. *ρ^E as defined above is independent of any ρ supported on a QECC $\mathcal{C}$ if and only if $\mathcal{N}$ satisfies the Knill-Laflamme conditions on $\mathcal{C}$.*

Proof. Independence of ρ holds if and only if there exists a basis $\{|i\rangle\}$ of $\mathcal{H}_E$ such that all the matrix entries $\rho^E_{ij} \in \mathbb{C}$ of $\rho^E = (\rho^E_{ij})_{ij}$ in this basis are equal to some $\lambda_{ij} \in \mathbb{C}$ independent of ρ (then it holds in any basis). Writing

$$\rho^E = \sum_{ij} \text{Tr}(E_j \rho E_i^\dagger) \, |j\rangle \langle i| \,, \tag{1.26}$$

and noticing that $\rho^E_{ij} = \mathrm{Tr}(E_j \rho E_i^\dagger) = \mathrm{Tr}(\Pi E_i^\dagger E_j \Pi \rho)$ for code states ρ, we have that

$$\mathrm{Tr}(\Pi E_i^\dagger E_j \Pi \rho) = \lambda_{ij} \quad \iff \quad \Pi E_i^\dagger E_j \Pi = \lambda_{ij} \Pi, \tag{1.27}$$

which holds because $\mathrm{Tr}(\Pi E_i^\dagger E_j \Pi \rho) = \langle \Pi E_i^\dagger E_j \Pi, \rho \rangle$ is a scalar product (the Hilbert-Schmidt inner product). □

It follows that the Knill-Laflamme conditions are necessary for the existence of a recovery map correcting against $\mathcal{N}$.

Definition 1.2.2 ((Non-)degenerate code)**.** An exact QECC for which the Knill-Laflamme conditions hold is said to be *non-degenerate* if the λ in (1.12) is full rank, otherwise it is *degenerate*.

A channel $\mathcal{N}$ acting over n physical qubits has dimension (definition A.0.3) at most 2^{2n}, whereas, when we restrict it to act on a subspace encoding k logical qubits, that trivial bound becomes $\dim \mathcal{N}|_{\mathcal{C}} \leq 2^{n+k}$ (such a reduction can be understood from the fact that many linearly independent Kraus operators will become linearly dependent when restricted on $\mathcal{C}$). The Knill-Laflamme conditions tell us that when the code is degenerate, the dimension of the channel is even smaller. The non-zero elements of $\{F_i|_{\mathcal{C}}\}$ are clearly linearly independent because they map to orthogonal subspaces, hence $\dim \mathcal{N}|_{\mathcal{C}} = N$ if there are N of them. If the code is highly degenerate, then $N \ll 2^{n+k}$ and one can consider a much smaller "effective" environment interacting with the system.

Degenerate codes are quite important and interesting because, generally speaking, they show that it is possible to design codes which are able to correct more errors than those that one can unambiguously identify. Usually (see chapter 7 of [8] for a discussion) one says that when a code is degenerate there are many errors which "look the same" on the code subspace (and so can be corrected by the same procedure), meaning that they are proportional to the same operator when they are restricted on the code subspace. Indeed, whenever there are two Kraus operators E_i, E_j such that $E_i|_{\mathcal{C}} = E_j|_{\mathcal{C}}$, then λ in (1.12) is clearly non-full rank because it has two equal rows (and two equal columns).

Stochastic noise

Consider the following intuition about a quantum channel $\mathcal{N}, \mathcal{N}(\rho) = \sum_i E_i \rho E_i^\dagger$. W.l.o.g. we can assume that $\mathrm{Tr}(E_i^\dagger E_i \rho) \neq 0 \ \forall i$. Then we can write it as:

$$\mathcal{N}(\rho) = \sum_i E_i \rho E_i^\dagger \tag{1.28}$$

$$= \sum_i \mathrm{Tr}(E_i^\dagger E_i \rho) \, \frac{E_i \rho E_i^\dagger}{\mathrm{Tr}(E_i^\dagger E_i \rho)}. \tag{1.29}$$

The second form suggests that we can understand a quantum channel as a map which takes ρ and maps it to an ensemble of states

$$\rho_i' \coloneqq \frac{E_i \rho E_i^\dagger}{\mathrm{Tr}(E_i^\dagger E_i \rho)} \tag{1.30}$$

(they are clearly positive and the denominator makes the trace equal to 1), where each of these states has been "prepared" with probability

$$p_\rho(i) \coloneqq \mathrm{Tr}(E_i^\dagger E_i \rho) \tag{1.31}$$

by the environment causing the noise. Note an important point: in general such a probability distribution is dependent on the state and on the chosen Kraus representation.

We make the following non-standard definition and prove the subsequent theorem.

Definition 1.2.3 (Stochastic noise)**.** Given a code subspace with projector Π, a channel $\mathcal{N}$ is stochastic w.r.t. Π and w.r.t. a Kraus representation of $\mathcal{N}$ with Kraus operators $\{E_i\}$ if $p_\rho(i) \coloneqq \mathrm{Tr}(E_i^\dagger E_i \rho)$ is independent of any code state ρ for all i.

As an example, a channel with a representation in which all Kraus operators are proportional to a unitary (e.g. Pauli channels in definition 1.1.9) is clearly stochastic w.r.t. such a representation for every Π. However, it is interesting to realize that stochasticity w.r.t. some Kraus representation does not imply stochasticity w.r.t. other representations of the same channel.

Theorem 1.2.2. *Given a code subspace with projector Π, a channel $\mathcal{N}$ is stochastic w.r.t. Π and w.r.t. any Kraus representation of $\mathcal{N}$ if and only if $\mathcal{N}$ satisfies the Knill-Laflamme conditions* (1.12) *on* Π.

Proof. As explained in lemma 1.2.1, the density matrix of the environment in the basis $\{|i\rangle\}$ of $\mathcal{H}_E$ is given by

$$\rho^E = \sum_{ij} \mathrm{Tr}(E_j^\dagger E_i \rho) \, |i\rangle \langle j| \, . \tag{1.32}$$

Stochasticity corresponds to the case in which the diagonal of ρ^E in the $\{|i\rangle\}$ basis is independent of ρ. However, it is possible that the off-diagonal terms are not, hence via a change of basis we can move the dependence on ρ from the off-diagonal terms to the diagonal, getting a Kraus representation of $\mathcal{N}$ w.r.t. which $\mathcal{N}$ is not stochastic. As an example, consider the case in which the environment E is a qubit with density matrix

$$\rho^E = \frac{1}{2}\mathbb{1} + \beta(\rho) X = \begin{pmatrix} \frac{1}{2} & \beta(\rho) \\ \beta(\rho) & \frac{1}{2} \end{pmatrix} \tag{1.33}$$

where $0 \leq \beta(\rho) \leq 1/2$. Then if we change basis using a Hadamard transformation we get

$$H \begin{pmatrix} \frac{1}{2} & \beta(\rho) \\ \beta(\rho) & \frac{1}{2} \end{pmatrix} H = \begin{pmatrix} \frac{1}{2} + \beta(\rho) & 0 \\ 0 & \frac{1}{2} - \beta(\rho) \end{pmatrix}. \tag{1.34}$$

Hence, in order to have stochasticity in any basis we need all the matrix elements of ρ^E to be independent of ρ. Then the statement follows from lemma 1.2.1. □

1.3 Approximate quantum error correction

It will prove to be useful, and actually compelling, to extend the notion of quantum error correction from the exact case, that we treated in the previous sections, to an approximate case, where we do not require to be able to recover perfectly the original state, but we just require to be close to it in terms of some measure. Indeed, exact quantum error correction is a very special case of approximate error correction: in general, an exact QECC (definition 1.1.3) w.r.t. some noise channel $\mathcal{N}$ is an approximate QECC (as we are going to define) w.r.t. other channels, even close or similar to $\mathcal{N}$, because they may not satisfy the Knill-Laflamme conditions (1.12) as well. It follows that developing a theory for AQEC is very important for experiments, where the physical noise channel is never precisely equal to an idealized model. Moreover, there are known examples [15] in which approximate error correction (AQEC) outperforms exact QEC.

Here we repeat the definition of entanglement fidelity (definition B.0.4), which is discussed more in detail in appendix B.

Definition 1.3.1 ((Generalized) entanglement fidelity)**.** Given quantum channels $\mathcal{N}, \mathcal{M} \colon \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H})$, for a state $\rho \in \mathcal{B}(\mathcal{H})$ with purification $|\psi\rangle \in \mathcal{H} \otimes \mathcal{H}_R$, where R is some reference system, we define the *entanglement fidelity* as

$$F_\rho(\mathcal{N}, \mathcal{M}) \coloneqq F\Big(\mathcal{N} \otimes \mathrm{id}_R(|\psi\rangle\langle\psi|), \mathcal{M} \otimes \mathrm{id}_R(|\psi\rangle\langle\psi|)\Big) \tag{1.35}$$

where F is the fidelity (definition B.0.2) and id_R is the identity channel on $\mathcal{B}(\mathcal{H}_R)$.

Definition 1.3.2. Given some code subspace $\mathcal{C}$ with projector Π, and two channels $\mathcal{N}, \mathcal{M}$, we define the *worst-case* entanglement fidelity of the code as

$$F(\mathcal{N}, \mathcal{M}) \coloneqq \min_\rho F_\rho(\mathcal{N}, \mathcal{M}) \tag{1.36}$$

where the minimum is taken over all the density matrices supported in the code subspace, i.e. over all the ρ such that $\rho = \Pi\rho\Pi$.
The entanglement fidelity is a measure of how "far apart" $\mathcal{N}$ and $\mathcal{M}$ are, but

it is not a metric in the mathematical sense. For that we define the *Bures distance* between channels as

$$\mathfrak{B}(\mathcal{N},\mathcal{M}) = \sqrt{1-\sqrt{F(\mathcal{N},\mathcal{M})}}, \tag{1.37}$$

which has the properties of a metric [1].

Definition 1.3.3 (Approximate Quantum Error Correcting Code)**.** We define an approximate quantum error correcting code (AQECC) w.r.t. a noise channel $\mathcal{N}$, with *approximation parameter* ϵ, as a QECC given by an encoder $\mathcal{E}$ for which there exists a recovery map $\mathcal{R}$ such that

$$\mathfrak{B}(\mathcal{R}\circ\mathcal{N}\circ\mathcal{E},\mathrm{id}) \leq \epsilon. \tag{1.38}$$

Notice that this corresponds to the requirement for the entanglement fidelity that $\min_\rho \sqrt{F_\rho(\mathcal{R}\circ\mathcal{N}\circ\mathcal{E},\mathrm{id})} \geq 1-\epsilon^2$, where the minimum is taken over density matrices supported on the code subspace. In the case of exact QEC (definition 1.1.3) the requirement could be restated as $F_\rho(\mathcal{R}\circ\mathcal{N}\circ\mathcal{E},\mathrm{id}) = 1\ \forall\rho$.

The main result that we will use in following chapters has been derived by Bény and Oreshkov in [1]:

Theorem 1.3.1 (Approximate Knill-Laflamme conditions)**.** *Given $\epsilon > 0$, $\mathcal{N}$ and $\mathcal{E}$ as above with a corresponding projector Π on the code subspace, there exists a recovery map satisfying* (1.38) *if and only if there exists a density matrix $\lambda = (\lambda_{ij})_{ij}$, with matrix elements $\lambda_{ij} \in \mathbb{C}$ w.r.t. a standard basis $\{|i\rangle\}$, such that, with*

$$\Pi B_{ij}\Pi \coloneqq \Pi E_i^\dagger E_j \Pi - \lambda_{ij}\Pi \tag{1.39}$$

$$\Lambda(\rho) \coloneqq \sum_{ij} \lambda_{ij}\,\mathrm{Tr}(\rho)\,|i\rangle\langle j| \tag{1.40}$$

$$\Lambda'(\rho) \coloneqq \Lambda(\rho) + \sum_{ij} \mathrm{Tr}(\rho B_{ij})\,|i\rangle\langle j|\,, \tag{1.41}$$

we have

$$\mathfrak{B}(\Lambda,\Lambda') \leq \epsilon. \tag{1.42}$$

In the following we will refer to equation (1.39) as the *approximate* Knill-Laflamme conditions in the sense that, if it holds that $\Pi B_{ij}\Pi = \mathbf{0}\ \forall i,j$, then equation (1.39) would reduce to the Knill-Laflamme conditions (theorem 1.2.1) and (1.42) would hold with $\epsilon = 0$, which means that the given code subspace would be an exact QECC.

Corollary 1.3.1. *A sufficient condition for* (1.42) *to be satisfied is that there exists a λ such that*

$$\max_{ij} \|\Pi B_{ij}\Pi\| \leq \frac{2}{N^2}\epsilon^2 \tag{1.43}$$

where N is the number of Kraus operators in a given representation of the channel $\mathcal{N}$. In the case in which one picks a minimal representation, $N =$

$\dim \mathcal{N}$ *(definition A.0.3). Considering* $\mathcal{N}$ *restricted over states supported on the code subspace* $\mathcal{C}$*, then* N *can be taken as small as* $\dim \mathcal{N}|_{\mathcal{C}}$*. Note that the l.h.s. depends on* λ *via the defining equation* (1.39)*, and that it depends also on the chosen Kraus representation.*

Proof. This result has also been derived in [16]. Lemma (B.0.1) states that $\sqrt{F(\rho,\sigma)} \geq 1 - \frac{1}{2}\left\|\rho - \sigma\right\|_1$. For a code state ρ and a purification $|\psi\rangle$ of ρ we have that:

$$\sqrt{F_\rho(\Lambda,\Lambda')} \geq 1 - \frac{1}{2}\left\|\Lambda \otimes \mathrm{id}(|\psi\rangle\langle\psi|) - \Lambda' \otimes \mathrm{id}(|\psi\rangle\langle\psi|)\right\|_1 \tag{1.44}$$

$$= 1 - \frac{1}{2}\left\|\sum_{ij} \mathrm{Tr}(B_{ij}|\psi\rangle\langle\psi|)\,|i\rangle\langle j|\right\|_1 \tag{1.45}$$

$$\geq 1 - \frac{1}{2}N^2 \max_{ij}\left|\mathrm{Tr}(B_{ij}|\psi\rangle\langle\psi|)\right| \tag{1.46}$$

$$\geq 1 - \frac{1}{2}N^2 \max_{ij}\left\|\Pi B_{ij}\Pi\right\| \tag{1.47}$$

where we used the fact that in general $\|M\|_1 \leq N^2 \max_{kl}|M_{kl}|$ for a $N\times N$ matrix M, and the matrix Hölder inequality. The statement follows immediately taking the minimum over ρ. □

1.4 Local approximate quantum error correction

Other notions of AQEC have been developed apart for the one studied by Bény and Oreshkov [1]. Here we discuss a notion of *local* approximate quantum error correction (LAQEC) which has been proposed in [17]. In this framework we will consider recovery maps acting only on some neighbourhood of a region where an error has occurred. There is an important assumption hidden here: we need to know which are the faulty qubits in order to apply a recovery map on a neighbourhood, which we need to be error-free. Hence in this section we will assume that we know the faulty locations, but that we do not know what specific error acted (e.g. an X or Z error), otherwise we could use also this information to develop an even better recovery map. To be more precise, in the following we will not need to know what is the exact location of an error, but just that it is contained in some region and not outside.

Moreover, the natural candidates for LAQEC are for example local stabilizer codes (definition 1.5.3), however, this notion of LAQEC invokes locality only for recovery maps, whereas we do not necessarily have to consider local stabilizer codes and we do not have to restrict to stabilizer codes at all.

As an example, the typical situation in which one knows the location of an error is the case of photon loss. Suppose that Alice is sending some qubits to Bob, in the form of photons. If Bob fails to detect some of them (knowing that he should have received one, for example by communicating classically with Alice), then we say that those photons have been *erased*. Another way

in which we may model this situation is by saying that Alice communicates to Bob what are the erased qubits by sending a quantum state orthogonal to all those that she may have sent. Then Bob can just perform an orthogonal measurement, revealing unambiguously to him if a photon was erased or not. More formally we define

Definition 1.4.1 (Erasure channel)**.** Consider an Hilbert space $\mathcal{H}$ with a decomposition $\mathcal{H} = (\mathcal{H}_A \oplus \mathcal{H}_A^\perp) \otimes \mathcal{H}_B$. Let $\sigma^A \in \mathcal{B}(\mathcal{H}_A^\perp)$ be a fixed state. We call a quantum channel $\Delta_A \colon \mathcal{B}(\mathcal{H}_A \otimes \mathcal{H}_B) \to \mathcal{B}(\mathcal{H}_A^\perp \otimes \mathcal{H}_B)$ the *erasure channel* on A if $\Delta_A(\rho^{AB}) = \sigma^A \otimes \rho^B$ for any $\rho^{AB} \in \mathcal{B}(\mathcal{H}_A \otimes \mathcal{H}_B)$.

Hence, the erasure channel traces out system A and replaces its original state with some fixed state.

Remark 1.4.1. Knowing the location of an error for some noise channel $\mathcal{N}$ but without knowing specifically which error occurred is equivalent to an erasure of that location: the faulty qubits end up in some unpredictable state, so we can just trace them out and replace their state with some fixed one.

To conform to the notation of [17], we define the Bures distance between density matrices ρ, σ as $\mathfrak{B}(\rho, \sigma) = (1 - \sqrt{F(\rho, \sigma)})^{1/2}$ where F is the fidelity. Moreover, for any code state we can consider its purification onto some reference system R.

Definition 1.4.2 (Local approximate correctability.)**.** Consider a code subspace $\mathcal{C}$ with projector Π, and assume that the noise is described by the erasure channel Δ_A on A. Partition the system into ABC, where A is the erased region, B is some region completely surrounding A, and C is the rest of the system. We say that the region A is ϵ-correctable from B (i.e. "locally") if there exists a recovery map $\mathcal{R}_B^{AB} \colon \mathcal{B}(\mathcal{H}_B) \to \mathcal{B}(\mathcal{H}_A \otimes \mathcal{H}_B)$ such that

$$\mathfrak{B}(\mathcal{R}_B^{AB}(\rho^{BCR}), \rho^{ABCR}) \leq \epsilon \tag{1.48}$$

for any purified code state ρ^{ABCR}.

In terms of the notation of the previous section we may write the above condition as $\mathfrak{B}(\mathcal{R}_B^{AB} \circ \Delta_A \circ \mathcal{E}, \mathrm{id}) \leq \epsilon$, in order to easily compare it with (1.38), where $\mathcal{E}$ is an encoder mapping to states in $\mathcal{C}$. Recall that in (1.38) we were using the entanglement fidelity in our definition, however, the entanglement fidelity of a state is the same as the fidelity for the purified state, which is what we are considering in this other definition.

Notice that, according to the definition, C can be empty and so the recovery map would not be really local. One may introduce a locality parameter specifying how big B can be, but we will not do that here (see [17]).

Definition 1.4.3 (ϵ-distance)**.** We define the ϵ-distance of a code with projector Π as the largest integer d_ϵ such that any region A of size $|A| < d_\epsilon$ is locally ϵ-correctable, where the size of A is the number of qubits in region A.

It is clear from the definition that if $\epsilon < \epsilon'$ then $d_\epsilon \leq d_{\epsilon'}$. Moreover, $\epsilon = 0$ correspond to the case of exact QEC, so that d_0 is the distance d for an exact QECC (definition 1.1.3) and therefore $d = d_0 \leq d_\epsilon$ for any ϵ.

The following theorem, proved in [17], is fundamental to derive the results of chapter 3:

Theorem 1.4.1. *Partitioning the system into regions ABC as above, we have*

$$\inf_{\mathcal{R}_B^{AB}} \sup_{\rho ABCR} \mathfrak{B}(\mathcal{R}_B^{AB}(\rho^{BCR}), \rho^{ABCR}) = \min_{\omega^A} \sup_{\rho ABCR} \mathfrak{B}(\omega^A \otimes \rho^{CR}, \rho^{ACR}) \quad (1.49)$$

where R is a purifying reference system and the suprema are taken over purified code states.

This theorem is important because it says that if the r.h.s. is small, then it is possible to find a quite good recovery map. Hence, this theorem gives us a tool to prove the existence of recovery maps achieving ϵ-correctability, for any chosen ϵ, without having to construct them explicitly. We just have to bound the correlations between A and CR, whose amount is expressed by the r.h.s. (recall that there are no correlations if $\rho^{ACR} = \rho^A \otimes \rho^{CR}$). It would feel more natural, then, to have on the r.h.s. the distance between ρ^{ACR} and $\rho^A \otimes \rho^{CR}$. Indeed, in [17] it is also proven that

$$\min_{\omega^A} \sup_{\rho ABCR} \mathfrak{B}(\omega^A \otimes \rho^{CR}, \rho^{ACR}) \leq \frac{1}{3} \sqrt{\sup_{\rho ABCR} \mathfrak{B}(\rho^A \otimes \rho^{CR}, \rho^{ACR})} \quad (1.50)$$

and so one can find upper bounds on $\mathfrak{B}(\rho^A \otimes \rho^{CR}, \rho^{ACR})$, instead.

Another important tool that is available when dealing with erasure noise is the so called union lemma, first proved in [18] for the case of exact QEC, and later proved in [17] for the approximate case:

Lemma 1.4.1 (Union lemma)**.** *Suppose that we have two erased regions A_1 and A_2, separated by a geometric distance x (in terms of lattice spacing), and suppose that A_1 is ϵ_1-correctable from a region B_1 which includes qubits at a geometric distance at most x from A_1. Moreover, assume that A_2 is ϵ_2-correctable (with B_2 possibly as big as the complement of A_2). Then $A_1 \cup A_2$ is $(\epsilon_1 + \epsilon_2)$-correctable.*

Clearly, this result would not apply for a general noise model because in that case the location of the errors is unknown and the statement has no meaning. However, in the case of erasure noise it is pretty useful, because it allows to prove correctability for regions possibly much bigger than the (simply) connected regions for which we may be able to prove correctability by other means.

1.5 Stabilizer codes and threshold probability

An important class of quantum error correcting codes is given by stabilizer codes. We give a few preliminary definitions:

Definition 1.5.1 (Pauli group)**.** The Pauli group $\mathcal{G}_n$ of n qubits is defined as the group with elements given by all the possible n-fold tensor products of Pauli matrices, where each tensor-product factor can be independently an I, X, Y or Z, together with multiplicative factors $\pm 1, \pm i$, where the group operation is the matrix multiplication. We refer only to those elements with a multiplicative factor $+1$ as *Pauli operators* (to conform to definition 1.1.7).

As an example, $\mathcal{G}_1 = \{\pm I, \pm iI, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}$,

$$\begin{aligned}\mathcal{G}_2 = \{ &\pm I\otimes I,\ \pm iI\otimes I,\ \pm X\otimes I,\ \pm iX\otimes I,\ \pm Y\otimes I,\ \pm iY\otimes I,\ \pm Z\otimes I,\ \pm iZ\otimes I,\\ &\pm I\otimes X,\ \pm iI\otimes X,\ \pm X\otimes X,\ \pm iX\otimes X,\ \pm Y\otimes X,\ \pm iY\otimes X,\ \pm Z\otimes X,\ \pm iZ\otimes X,\\ &\pm I\otimes Y,\ \pm iI\otimes Y,\ \pm X\otimes Y,\ \pm iX\otimes Y,\ \pm Y\otimes Y,\ \pm iY\otimes Y,\ \pm Z\otimes Y,\ \pm iZ\otimes Y,\\ &\pm I\otimes Z,\ \pm iI\otimes Z,\ \pm X\otimes Z,\ \pm iX\otimes Z,\ \pm Y\otimes Z,\ \pm iY\otimes Z,\ \pm Z\otimes Z,\ \pm iZ\otimes Z\}\end{aligned} \tag{1.51}$$

Definition 1.5.2 (Stabilizer group)**.** We say that a subgroup $\mathcal{S}_n$ of the Pauli group $\mathcal{G}_n$ is a *stabilizer group* if it is abelian and it does not contain $-I^{\otimes n}$.

A stabilizer group can be compactly specified by $m \leq n$ independent *generators* $S_1, \ldots, S_m$ (i.e. the minimal number of elements such that any other one can be expressed as a product of them) and we use the notation $\mathcal{S}_n = \langle S_1, \ldots, S_m \rangle$ to mean that $\mathcal{S}_n$ is generated by the set on the r.h.s.

Definition 1.5.3 (Local stabilizer code)**.** We say that a family of stabilizer codes with increasing system size n, where the qubits are distributed on a lattice, is *local* if $\forall n$ there exist sets of generators $G_n = \{S_1^{(n)}, \ldots, S_{m_n}^{(n)}\}$ such that each $S \in G_n$ is *geometrically local*, i.e. S acts with an X, Y or Z over qubits which are at a geometric distance from each other (in terms of lattice spacing) upper bounded by a constant independent of n.

Regarding the following definition, note that every two elements in the Pauli group can either commute or anticommute.

Definition 1.5.4 ((Error) syndrome)**.** Given a Pauli operator P (in the following, we will often refer to Pauli operators as *errors*, except for the identity $I^{\otimes n}$), we define its (error) syndrome w.r.t. a given set of stabilizer generators $\{S_i\}_{i=1}^m$ as the vector $\boldsymbol{k} = (k_1, \ldots, k_m)$, where k_i is 1 if $[S_i, P] = \mathbf{0}$ and -1 if $\{S_i, P\} = \mathbf{0}$. We will also say that P is *compatible* with the syndrome $\boldsymbol{k}$.

Definition 1.5.5 (Stabilizer code)**.** Let $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$ and $\mathcal{S}_n$ be a stabilizer group for n qubits. A stabilizer (quantum error correcting) code is given by the simultaneous $+1$-eigenspace Π of the elements in $\mathcal{S}_n$.

One can show (see [9] for a discussion) that, if $\mathcal{S}_n$ has m independent generators, it encodes $n-m$ logical qubits, and that the distance d is given by the smallest-weight operator which commutes with all the generators of $\mathcal{S}_n$ and which is not contained in $\mathcal{S}_n$.

There are two main types of recovery map for stabilizer codes: one is the *smallest-weight* recovery map (sometimes referred to as the minimal-matching recovery map) and the other one is the *maximum-likelihood* recovery map. The second one can be easily understood: if the noise channel is an i.i.d. Pauli channel (definition 1.1.9), we can assign to every Pauli operator a probability to happen, and then it makes sense to suppose that the error which actually happened was the most likely one compatible with the observed syndrome. However, a smallest-weight recovery map is often used: for example, in the case of the toric code [2, 3] by applying a long error-correction chain there is a higher risk that we close non-trivial loops around the torus, hence it is convenient to apply a small-weight one. Here we will describe and analyse this second choice.

Definition 1.5.6 (Smallest-weight recovery map for stabilizer codes)**.** Given a stabilizer code $\mathcal{S}_n = \langle S_1, \ldots, S_m \rangle$, defined over n qubits, suppose that for each syndrome $\boldsymbol{k}$ there exists a unique smallest-weight Pauli operator $P_{\boldsymbol{k}}$ compatible with it. Call $M_{\boldsymbol{k}}$ the projector onto the common k_i-eigenspace of the S_i, $i = 1 \ldots m$, i.e.

$$M_{\boldsymbol{k}} = 2^{-m} \Big(\mathbb{1} + k_1 S_1\Big) \ldots \Big(\mathbb{1} + k_m S_m\Big). \tag{1.52}$$

Then we define the *smallest-weight recovery map* $\mathcal{R}$ as the map:

$$\mathcal{R} \colon \mathcal{B}((\mathbb{C}^2)^{\otimes n}) \to \mathcal{B}((\mathbb{C}^2)^{\otimes n}) \tag{1.53}$$

$$\mathcal{R}(\rho) = \sum_{\boldsymbol{k}} P_{\boldsymbol{k}} M_{\boldsymbol{k}} \rho M_{\boldsymbol{k}} P_{\boldsymbol{k}}. \tag{1.54}$$

If the smallest-weight Pauli operator compatible with $\boldsymbol{k}$ is not unique, then we choose one and define the smallest-weight recovery map w.r.t. that one. In the following, assume for simplicity that it is unique.

It is easy to check that the operators $\{P_{\boldsymbol{k}} M_{\boldsymbol{k}}\}$ form a valid set of Kraus operators and that therefore $\mathcal{R}$ as defined is a quantum channel. First we have that

$$(P_{\boldsymbol{k}} M_{\boldsymbol{k}})^\dagger (P_{\boldsymbol{k}} M_{\boldsymbol{k}}) = M_{\boldsymbol{k}} P_{\boldsymbol{k}}^2 M_{\boldsymbol{k}} = M_{\boldsymbol{k}} \geq 0, \tag{1.55}$$

being $M_{\boldsymbol{k}}$ a projector. Second, we can see that

$$\sum_{\boldsymbol{k}} (P_{\boldsymbol{k}} M_{\boldsymbol{k}})^\dagger (P_{\boldsymbol{k}} M_{\boldsymbol{k}}) = \sum_{\boldsymbol{k}} M_{\boldsymbol{k}} = \mathbb{1}. \tag{1.56}$$

Notice also that $M_{\mathbf{1}} = \Pi$, where $\mathbf{1} \coloneqq (1, \ldots, 1)$ and Π is the code subspace projector corresponding to $\mathcal{S}$.

Lemma 1.5.1. *The smallest-weight recovery map can be expressed as:*

$$\mathcal{R}(\rho) = \sum_{\boldsymbol{k}} \Pi P_{\boldsymbol{k}} \rho P_{\boldsymbol{k}} \Pi \tag{1.57}$$

Proof. Recall from the definition of the syndrome that if $P_{\boldsymbol{k}}$ is compatible with $\boldsymbol{k}$ it means that $[P_{\boldsymbol{k}}, S_i] = \mathbf{0}$ if $k_i = 1$ and $\{P_{\boldsymbol{k}}, S_i\} = \mathbf{0}$ if $k_i = -1$. These two equations can be expressed respectively as $P_{\boldsymbol{k}} S_i P_{\boldsymbol{k}} = S_i$ and $P_{\boldsymbol{k}} S_i P_{\boldsymbol{k}} = -S_i$, or equivalently as

$$P_{\boldsymbol{k}} S_i P_{\boldsymbol{k}} = k_i S_i. \tag{1.58}$$

It follows that

$$P_{\boldsymbol{k}} M_{\boldsymbol{k}} P_{\boldsymbol{k}} = 2^{-m} \left(\mathbb{1} + k_1 P_{\boldsymbol{k}} S_1 P_{\boldsymbol{k}}\right) \dots \left(\mathbb{1} + k_m P_{\boldsymbol{k}} S_m P_{\boldsymbol{k}}\right) \tag{1.59}$$

$$= 2^{-m} \left(\mathbb{1} + k_1^2 S_1\right) \dots \left(\mathbb{1} + k_m^2 S_m\right) = \Pi. \tag{1.60}$$

Therefore

$$P_{\boldsymbol{k}} M_{\boldsymbol{k}} = \Pi P_{\boldsymbol{k}}. \tag{1.61}$$

□

Given a noise channel $\mathcal{N}$, $\mathcal{N}(\rho) = \sum_i E_i \rho E_i^\dagger$, if ρ is a code state ($\Pi\rho\Pi = \rho$), then the action of the smallest-weight recovery map on $\mathcal{N}(\rho)$ can be expressed as

$$\mathcal{R}\Big(\mathcal{N}(\rho)\Big) = \sum_{i\boldsymbol{k}} \Big(\Pi P_{\boldsymbol{k}} E_i \Pi\Big) \rho \Big(\Pi E_i^\dagger P_{\boldsymbol{k}} \Pi\Big). \tag{1.62}$$

We will refer to this equation in the following subsection.

The problem with the smallest-weight recovery map (and not so differently for a maximum-likelihood one), is that it is difficult to compute it, in the sense that finding the smallest-weight Pauli operator compatible with a given syndrome is exponentially hard in the system size n (notice that it is a classical problem). The practical solution is to use a classical algorithm which finds a "small"-weight error (but not necessarily the smallest one) in a polynomial amount of time. However, the performance of the recovery map may be seriously compromised, so that it is really difficult in general to find a good compromise, and it is not clear even when it is actually possible.

Definition 1.5.7 (Recovery algorithm)**.** Given a stabilizer code, we call *recovery algorithm* any classical algorithm taking an error syndrome $\boldsymbol{k}$ (definition 1.5.4) as input and giving a Pauli operator compatible with $\boldsymbol{k}$ as output.

AQEC for stabilizer codes

In this section we will consider Pauli noise (definition 1.1.9), i.e. we take $\mathcal{N}_P \coloneqq (\mathcal{N}_P^1)^{\otimes n}$ as noise channel, where

$$\mathcal{N}_P^1(\rho) \coloneqq (1-p)\,\rho + p_X\, X\rho X + p_Y\, Y\rho Y + p_Z\, Z\rho Z. \tag{1.63}$$

and $p = p_X + p_Y + p_Z$ is the overall single-qubit error probability (definition 1.1.10). For simplicity, in the following we consider $p_X = p_Y = p_Z = p/3$.

Theorem 1.5.1. *Consider Pauli noise $\mathcal{N}_P$ as above. A stabilizer code over n qubits satisfies*

$$\min_{\rho} F\Big(\mathcal{R}(\mathcal{N}_P(\rho)), \rho\Big) \geq 1 - \delta_{n,p} \tag{1.64}$$

where the minimum is taken over code states ρ, where $\mathcal{R}$ is the smallest-weight recovery map and

$$\delta_{n,p} = \sum_{w > \frac{d-1}{2}} g_w(n) \binom{n}{w} p^w (1-p)^{n-w}. \tag{1.65}$$

The distance of the code is d and $g_w(n)$ is the (system-size dependent) fraction of errors with weight w which are not *correctable by the smallest-weight recovery map, i.e. it is the fraction of those Pauli operators P such that $\mathcal{R}(P\rho P) \neq \rho$ at least for some code state ρ.*

Remark 1.5.1. The validity of the formula (1.65) is actually not restricted to the smallest-weight recovery map but it applies to any recovery map whose Kraus operators can all be expressed as Pauli operators.

Proof. By definition of Pauli noise the Kraus operators $\{E_i\}_{i=0}^{4^n}$ of $\mathcal{N}_P$ are

$$E_i = \Big(\frac{p}{3}\Big)^{w_i} (1-p)^{n-w_i} P_i \tag{1.66}$$

where w_i is the weight of the Pauli operator P_i and i indexes all the Pauli operators over n qubits.
By joint concavity of the fidelity, the minimum in (1.64) is achieved over pure states $\rho = |\psi\rangle\langle\psi|$. In that case, for $|\psi\rangle$ in the code subspace, using (1.62) and (1.66):

$$F\Big(\mathcal{R}(\mathcal{N}_P(|\psi\rangle\langle\psi|)), |\psi\rangle\langle\psi|\Big) = \langle\psi|\mathcal{R}(\mathcal{N}_P(|\psi\rangle\langle\psi|))|\psi\rangle \tag{1.67}$$

$$= \sum_{i\boldsymbol{k}} |\langle\psi|\Pi P_{\boldsymbol{k}} E_i \Pi|\psi\rangle|^2 \tag{1.68}$$

$$= \sum_{i\boldsymbol{k}} \Big(\frac{p}{3}\Big)^{w_i} (1-p)^{n-w_i} |\langle\psi|\Pi P_{\boldsymbol{k}} P_i \Pi|\psi\rangle|^2 . \tag{1.69}$$

Then we have

$$\min_{|\psi\rangle} F\Big(\mathcal{R}(\mathcal{N}_P(|\psi\rangle\langle\psi|)), |\psi\rangle\langle\psi|\Big) \geq \sum_{i\boldsymbol{k}} \Big(\frac{p}{3}\Big)^{w_i} (1-p)^{n-w_i} \min_{|\psi\rangle} |\langle\psi|\Pi P_{\boldsymbol{k}} P_i \Pi|\psi\rangle|^2 \tag{1.70}$$

Regarding the last term, notice first that for each i the sum over $\boldsymbol{k}$ reduces to a single term because a Pauli operator is compatible with only one syndrome and in the other cases it happens that $\{P_{\boldsymbol{k}} P_i, \Pi\} = 0$, so that

$$\langle\psi|\Pi P_{\boldsymbol{k}} P_i \Pi|\psi\rangle = -\langle\psi|\Pi^2 P_{\boldsymbol{k}} P_i|\psi\rangle = -\langle\psi|\Pi P_{\boldsymbol{k}} P_i \Pi|\psi\rangle , \tag{1.71}$$

which implies that such a term is 0 (we used $\Pi^2 = \Pi$ and $\Pi|\psi\rangle = |\psi\rangle$). For the remaining term (the one for which P_i is compatible with syndrome $\boldsymbol{k}$) we have that $P_{\boldsymbol{k}}P_i$ commutes with the code subspace projector Π, and in particular

$$|\langle\psi|\Pi P_{\boldsymbol{k}}P_i\Pi|\psi\rangle| = 1 \qquad \text{if } P_{\boldsymbol{k}}P_i\Pi = \Pi \tag{1.72}$$

$$|\langle\psi|\Pi P_{\boldsymbol{k}}P_i\Pi|\psi\rangle| \leq 1 \qquad \text{if } P_{\boldsymbol{k}}P_i \text{ is a logical operator.} \tag{1.73}$$

The first case is the one in which the recovery map worked in the sense that $\mathcal{R}(P_i\rho P_i) = \rho$ for every code state ρ. In the second one we have that the value of $|\langle\psi|\Pi P_{\boldsymbol{k}}P_i\Pi|\psi\rangle|$ depends on $|\psi\rangle$: for example, if $P_{\boldsymbol{k}}P_i = \bar{Z}$ (we use bars for logical states and operators), then if $|\psi\rangle = |\bar{0}\rangle$ we get 1, but if $|\psi\rangle = |\bar{+}\rangle$ then we get a 0. Because for stabilizer codes a Pauli operator commuting with the code subspace projector and different from the identity on the code subspace is either an $\bar{X}, \bar{Y}$ or $\bar{Z}$ for at least one of the encoded qubits, then there is always a state $|\psi\rangle$ which is mapped to a state orthogonal to $|\psi\rangle$ itself, similarly to the previous example.
It follows that when we take the minimum in equation (1.70), then for each term i we get either 1 if P_i is correctable or 0 if it is not. So

$$\min_{|\psi\rangle} F\Big(\mathcal{R}(\mathcal{N}_P(|\psi\rangle\langle\psi|)), |\psi\rangle\langle\psi|\Big) \geq \sum_w \Big(\frac{p}{3}\Big)^w (1-p)^{n-w}\tilde{G}_w(n) \tag{1.74}$$

where $\tilde{G}_w(n)$ is the number of Pauli operators of weight w which are correctable. Using the fact that the total number of Pauli operators over n qubits for a given weight w is $3^w\binom{n}{w}$, we get the statement of the theorem. □

Threshold probability

We first give a general definition, then we give some results which apply in general and some which are specific (but not necessarily restricted) to stabilizer codes.

Definition 1.5.8 (Threshold probability)**.** Consider a family of QECCs with increasing system size n and an i.i.d. error model $\mathcal{N}_{n,p} = (\mathcal{N}_p^1)^{\otimes n}$ admitting a single-qubit error probability (definitions 1.1.5 and 1.1.10). We say that such a family admits a threshold probability $\bar{p}$ w.r.t. $\mathcal{N}_{n,p}$ if there exists a sequence of recovery maps $\mathcal{R}_n$ such that:

$$\lim_{n\to\infty} \min_{\rho} F\Big(\mathcal{R}_n(\mathcal{N}_{n,p}(\rho)), \rho\Big) = 1 \qquad \forall p < \bar{p} \tag{1.75}$$

$$\lim_{n\to\infty} \min_{\rho} F\Big(\mathcal{R}_n(\mathcal{N}_{n,p}(\rho)), \rho\Big) < 1 \qquad p = \bar{p} \tag{1.76}$$

where the minima are taken over code states ρ.

Corollary 1.5.1. *A family of stabilizer codes admits a threshold probability w.r.t. $\mathcal{N}_{n,p}$ if there exists a $\tilde{p}$ such that*

$$\lim_{n\to\infty} \delta_{n,p} = 0 \qquad \forall p < \tilde{p} \tag{1.77}$$

where $\delta_{n,p}$ is defined by equation (1.65)*.*

We can ask now under which conditions a threshold probability exists.

Theorem 1.5.2. *A family of QECCs with linear distance $2\tilde{p}n+1$ has a threshold probability w.r.t. $\mathcal{N}_{n,p}$ greater or equal to $\tilde{p}$.*

Proof. Note that $\mathcal{N}_{n,p}$ is not necessarily a Pauli channel but here we will use anyway the word weight for a Kraus operator (error) to indicate the number of its non-identity tensor product factors. The weight of such errors is distributed according to a binomial distribution with average value np and standard deviation $\sqrt{np(1-p)}$. For $p < \tilde{p}$ the interval $[np, n\tilde{p}]$ contains a number $\mathcal{O}(\sqrt{n})$ of standard deviations, so that the integral of the binomial distribution in $[n\tilde{p}, \infty]$ goes to 0 for $n \to \infty$, which means that the probability of having an error with weight exceeding the distance is asymptotically 0. □

Hence, a linear distance is sufficient for the existence of a threshold probability, but for example the toric code [2, 3] shows that it is not necessary.

Theorem 1.5.3. *Referring to equation* (1.65)*, given a stabilizer code with distance $2t(n)+1$, the following conditions are equivalent*

$$\lim_{n\to\infty} \delta_{n,p} = 0 \quad \forall p < \tilde{p} \tag{1.78}$$

$$\lim_{n\to\infty} \sum_{w=t(n)+1}^{n\tilde{p}} g_w(n) \binom{n}{w} p^w (1-p)^{n-w} = 0 \quad \forall p < \tilde{p}. \tag{1.79}$$

Hence a stabilizer code admits a threshold probability $\bar{p}$ such that $\bar{p} \geq \tilde{p}$ if equation (1.79) *holds for some $\tilde{p} > 0$.*

Proof. Necessity is clear because otherwise the entire sum up to n cannot go to 0. Sufficiency follows from the fact that the sum from $n\tilde{p}+1$ to n is anyway going to 0 for $p < \tilde{p}$ similarly to the case of the previous theorem. □

Definition 1.5.9 (Effective linear distance)**.** Whenever a stabilizer code satisfies (1.79) for some $\tilde{p} > 0$, we say that it has an *effective linear distance*.

The above result shows that a code with sub-linear distance can still admit a threshold probability if $g_w(n)$ "boosts" the code to an effective linear distance. That is one of the reasons why we claimed that the distance by itself is not a very significant parameter. In principle it is also possible that a linear distance code admits a higher effective linear distance.

Corollary 1.5.2. *A sufficient condition for the existence of a threshold probability for a stabilizer code with distance $d = 2t(n)+1$ is that there exists a $\tilde{p} > 0$ such that $\forall p < \tilde{p}$:*

$$\lim_{n\to\infty} n\, g_w(n) \binom{n}{w} p^w (1-p)^{n-w} = 0 \quad \forall\ t(n)+1 \leq w \leq n\tilde{p}. \tag{1.80}$$

Notice that w is not taken fixed while taking the limit, because if the distance is strictly increasing, then, if we take w fixed, by definition of distance there always exists a finite n' for which $g_w(m) = 0\ \forall m \geq n'$, independently of the fact that a threshold probability exists or not.

Chapter 2

Entanglement Renormalization

Tensor networks (for a review see [19]) are an efficient way to describe some classes of quantum states. For example, for a system of n qubits the Hilbert space is $(\mathbb{C}^2)^{\otimes n}$, which has dimension 2^n, so that to specify a generic vector one needs 2^n coefficients: an exponential number in the system size. However, it is possible to find more compact representations for specific states. Even in the case in which an exact representation with few parameters is not possible, it may still be possible to find an approximate representation which is efficient, meaning that the tensor network does not represent precisely the state we wanted, but an approximately equal state whose properties are very close to the original ones (namely, average values w.r.t. observables of physical interest, e.g. the energy).

We can represent quantum states via tensors because of the following observation. Consider a qudit, i.e. a quantum system whose Hilbert space $\mathcal{H}$ is d-dimensional, and let $\{|i\rangle\}_{i=1}^{d}$ be a basis of $\mathcal{H}$ (in the following we will always consider qubits, but the argument holds for any finite d). Assume now that the system of interest is made up of n qudits. Then a state $|\psi\rangle \in \mathcal{H}^{\otimes n}$ can be written as

$$|\psi\rangle = \sum_{i_1,\dots,i_n} \psi_{i_1\dots i_n} |i_1 \dots i_n\rangle \tag{2.1}$$

where $\psi_{i_1\dots i_n}$ are some coefficients that we can view as the entries of a tensor with n indices, each of which can take d values. Then taking the scalar product $\langle\tilde{\psi}|\psi\rangle$ corresponds to tensor contraction:

$$\langle\tilde{\psi}|\psi\rangle = \sum_{i_1,\dots,i_n,\, j_1,\dots,j_n} \tilde{\psi}_{j_1\dots j_n}\psi_{i_1\dots i_n} \langle j_1 \dots j_n|i_1 \dots i_n\rangle \tag{2.2}$$

$$= \sum_{i_1,\dots,i_n} \tilde{\psi}_{i_1\dots i_n}\psi_{i_1\dots i_n}. \tag{2.3}$$

An example in which such a tensor has a simpler structure and can be represented by possibly much fewer parameters than d^n is the case of Matrix Product States (MPS) for 1-dimensional systems [20] (see figure 2.0.1): in this

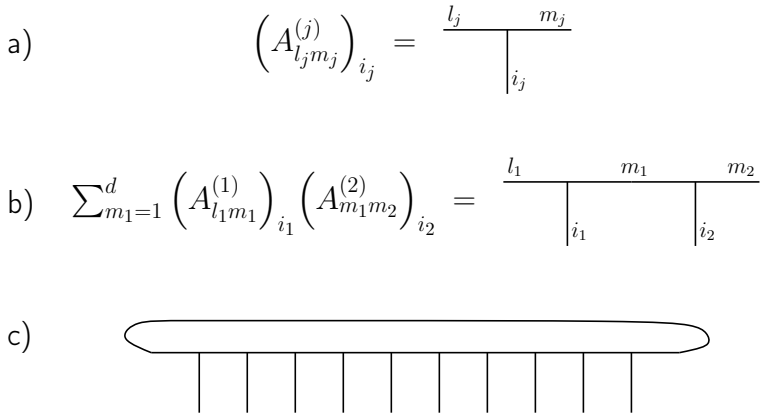

Figure 2.0.1: a) A 3 tensor is represented with three legs, each one corresponding to a different index (we will often omit such indices). b) Connecting two legs implicitly means that we are contracting the corresponding indices. c) An MPS is obtained by joining many smaller tensors. The first and last index also form a bond when we take periodic boundary conditions.

case, by definition, $\psi_{i_1 \dots i_n}$ can be written as a product of n matrices $A^{(j)}$ which carry an extra index i_j (hence they are really 3-tensors and not matrices), such that, assuming periodic boundary conditions,

$$\psi_{i_1 \dots i_n} = \mathrm{Tr}\,(A^{(1)}_{i_1} \dots A^{(n)}_{i_n}). \tag{2.4}$$

Without periodic boundary conditions the first and last matrix are respectively a row and a column vector. If all those matrices are $D \times D$, then the tensor is given overall by nD^2 independent coefficients, which is a linear number in the system size, instead of exponential. Moreover, if the system possesses translational invariance, then such a number reduces just to D^2 because all those matrices would be equal to each other.

The indices $i_1, \dots, i_n$ are called the *physical indices*, whereas the indices l_j, m_j of the matrices $A^{(j)} = (A^{(j)}_{l_j m_j})_{l_j m_j}$ are called the *virtual indices*. Hence an MPS can be decomposed in smaller tensors $A^{(j)}_{i_j}$ which we can represent with three legs (see figure 2.0.1), where one leg represents the physical index i_j and the other two represent the virtual indices m_j, l_j. Whenever we connect two legs, we say that there is a *bond* between the corresponding tensors. A bond represents a *contraction* of the indices corresponding to the two legs, i.e. we set the two indices to be equal and we sum over them.

MPS have proven to be very useful in simulating 1-dimensional non-critical systems, specifically their ground states [21]. However, they are not defined in higher dimensions and they fail to accurately represent 1-dimensional critical systems. There have been different approaches to overcome these obstacles by defining other types of tensor networks: one example are Projected Entangled

Pairs in higher dimensions (PEPS, [22]), whereas the Multi-scale Entanglement Renormalization Ansatz (MERA, [4, 5]) works quite well in simulating 1-dimensional critical systems and can be used also in higher dimensions. In this and in the following chapters we will focus only on MERA.

Here we summarize some properties of MERA which are useful especially for its application in condensed matter physics. They are discussed in detail in [5]:

1. efficient contractibility: expectation values of local observables can be computed efficiently (because reduced density matrices over a few sites can be computed efficiently);

2. versatility w.r.t. lattice structure, topology and number of dimensions;

3. built-in support to the area law for the entanglement entropy in any dimension and to the logarithmic scaling of the entanglement entropy in 1D; power law decay of correlation functions;

4. symmetries are easy to be included in the description.

2.1 The Multi-scale Entanglement Renormalization Ansatz

We will start by defining the structure of a MERA and subsequently discuss about its usage.

A MERA is a tensor network, sometimes called *holographic* because the tensors in a MERA do not reproduce the lattice geometry as it is the case of MPS and PEPS, but they are organised in *levels* or *layers*, and one can think of these levels as defining an "emergent" dimension or, as we will call it, a *renormalization scale*. These levels can be described as mapping a lattice at some scale to a new lattice at a larger one, as we will formalise in section 2.2. We index the levels of a MERA from 0 at the bottom to some s at the top. Here we will describe a *1D binary MERA* for simplicity (see figure 2.1.1), but the construction can be easily generalised. We will also consider periodic boundary conditions, but open boundary conditions can be described by a MERA as well.

The building blocks of a (binary) MERA are the following tensors (see figure 2.1.2 for some more general examples):

1. *Disentanglers*: two-body unitaries mapping a state over two sites to another state over those two. As we shall explain, such unitaries are called in this way because in condensed matter applications their function is to remove local entanglement, going from the bottom of the network to the top (whereas of course they introduce it going in the other direction).

2. *Isometries*: recall that an isometry $V\colon \mathcal{H}' \to \mathcal{H}$ is a map such that $V^\dagger V = \mathbb{1}_{\mathcal{H}'}$. In this case $\mathcal{H}$ is is the Hilbert space of two neighbouring

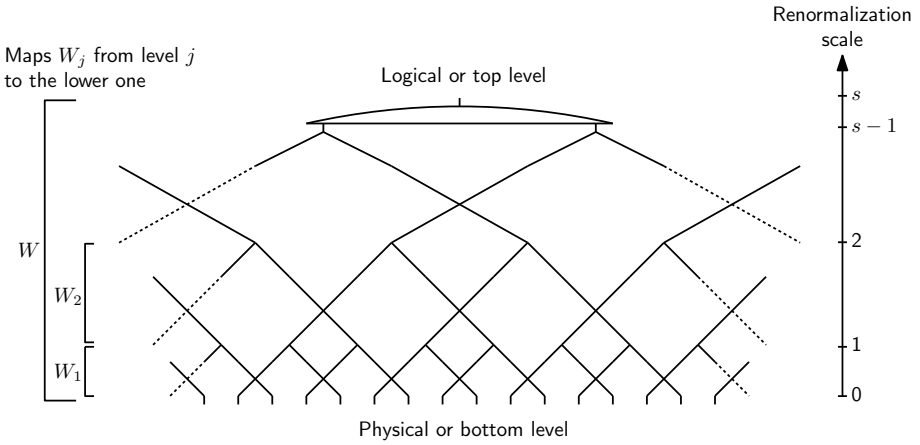

Figure 2.1.1: An example of a MERA with s levels, where here $s = 4$. More specifically, we show a 1D binary MERA with 2-to-2 body disentanglers (the crosses) and 1-to-2 body isometries (triangles). We also draw a 1-to-2 top tensor as an example. We assume periodic boundary conditions, so the dotted line of a disentangler on the right corresponds to the solid line of the corresponding disentangler on the left, and vice-versa.

sites at some level and $\mathcal{H}'$ is the Hilbert space of a single site one level higher. In the renormalization direction (from bottom to top) we are applying $V^\dagger$.

3. A *top tensor* which may vary depending on the network considered: in condensed matter physics it is usually an isometry T for which $T^\dagger$ is mapping the whole second to last level to a 1-dimensional Hilbert space, i.e. the MERA represents a single state, whereas in the quantum error correction perspective $T^\dagger$ maps to the Hilbert space of k sites at the top, whose state is not fixed, and the MERA then represents a subspace of states instead of just one. In section 2.2 and chapter 3 those k sites will be the logical qubits.
   Often the top tensor can be just a combination of disentanglers and isometries, with the same structure of the rest of the network: in such a case we will say that there is a trivial top tensor.

The network is then constructed using the above building blocks: starting from the bottom, there is a layer of disentanglers followed by one of isometries; we call *level* of the MERA the combination of these two layers. Then, similarly, we add layers of disentanglers and isometries till reaching one level before what we want to be the top of the network. Finally we add the top tensor, if there is a non-trivial one, otherwise we just add a final layer of disentanglers and isometries.

Notice that in principle all disentanglers can be different from each other

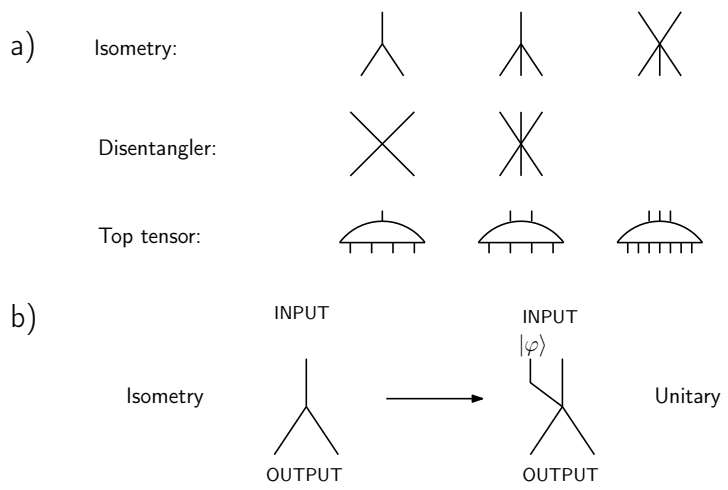

Figure 2.1.2: a) A few graphical examples of what can be taken as the geometric structure of isometries, disentanglers and top tensors. For example, the isometry can be taken as a 1-to-2 body map or as a 1-to-3 map. b) A 1-to-2 body isometry is equal to some 2-to-2 unitary with a fixed input state $|\varphi\rangle$.

and the same holds for the isometries, however in practice one often picks them all equal to each other, either to simplify the description of the network, or because there is a specific physical motivation (we define such a MERA as *scale-invariant* in definition 3.1.1).

It will be useful for the following to see that in this setting an isometry V can be viewed as a unitary with one input state which has been fixed (see figure 2.1.2): $V|\psi\rangle = U_V(|\psi\rangle \otimes |\varphi\rangle)$ for some unitary U_V. As a standard choice, for qubits we choose the state $|\varphi\rangle$ to be $|0\rangle$, i.e. the $+1$-eigenstate of Pauli Z.

Definition 2.1.1 (Unitary extension)**.** We call U_V the unitary extension of the isometry V. Given a MERA, we call its unitary extension the network with the same structure constructed using the same disentanglers and the unitary extensions of the isometries and of the top tensor (see also figure 2.1.3).

Renormalization map and causal cones

We will say that going from some level to one level higher is completing one *renormalization step*. One level of the MERA, which is made up of a layer of disentanglers and one of isometries, is just an isometry $W_j \colon \mathcal{H}_j \to \mathcal{H}_{j-1}$ overall, mapping the Hilbert space $\mathcal{H}_j$ of the full lattice at level j to the Hilbert space $\mathcal{H}_{j-1}$ one level lower. Hence, in one renormalization step we are applying $W_j^\dagger \colon \mathcal{H}_{j-1} \to \mathcal{H}_j$.

Definition 2.1.2 (Renormalization map)**.** For a MERA we define the *renormalization map* $(\mathcal{E}^*)_{j-1}^{j}$ from level $j-1$ to level j as $(\mathcal{E}^*)_{j-1}^{j} \colon O_{j-1} \mapsto W_j^\dagger O_{j-1} W_j$,

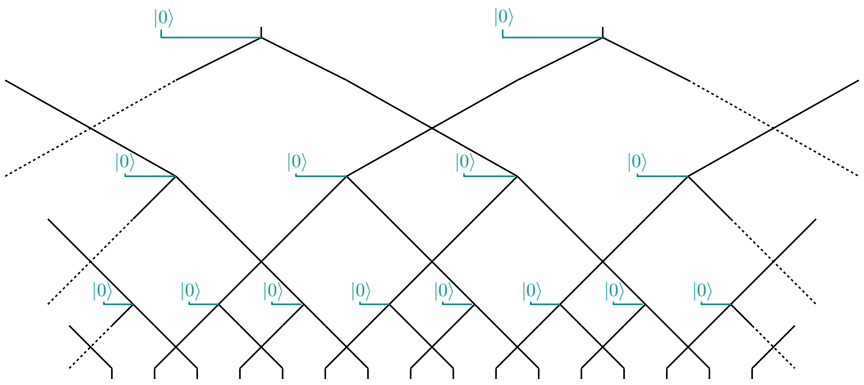

Figure 2.1.3: Unitary extension of the MERA in figure 2.1.1, neglecting the top tensor for simplicity of the representation (it can also be unitarily extended).

where O_{j-1} is an operator acting on $\mathcal{H}_{j-1}$ (the star will be clear in the following). We call $O'_j \coloneqq W_j^\dagger O_{j-1} W_j$ the *renormalized operator*. In particular we can renormalise density matrices ρ_{j-1}; then, a pure state $|\psi_{j-1}\rangle$ is mapped to $|\psi'_j\rangle \coloneqq W_j^\dagger |\psi_{j-1}\rangle$ and we call it the *renormalised state*.

We can draw tensor network diagrams representing the action of the renormalization map and describing pictorially the operations involved. We draw O'_j in figure 2.1.4. We can see that if an operator is supported on a limited number of sites, then the structure of the MERA allows to simplify a lot the computations that one has to do: whenever a disentangler is contracted with its adjoint, it gives the identity, and similarly for the isometries. The tensors which do not simplify are those which have one or more legs connected to the operator O, or those which are connected to other tensors with such property. It is convenient to define the *past causal cone* (see also figure 2.1.5).

Definition 2.1.3 (Past causal cone)**.** Let $G^{(j)} = \{g_1^{(j)}, g_2^{(j)}, \dots\}$ be a set of sites belonging to the lattice at level j. For each site $g_i^{(j)} \in G^{(j)}$ consider the bonds of the tensor network connecting it to other sites $g_{i,1}^{(j+1)}, g_{i,2}^{(j+1)}, \dots$ one level higher in the network, and define a set $\tilde{G}_i^{(j+1)} = \{g_{i,1}^{(j+1)}, g_{i,2}^{(j+1)}, \dots\}$. Define $G^{(j+1)} \coloneqq \bigcup_i \tilde{G}_i^{(j+1)}$ and repeat the procedure for $G^{(j+1)}$ till the top of the network at level s. The *past causal cone* $\mathfrak{C}_{G^{(j)}}$ of $G^{(j)}$ is defined as $\mathfrak{C}_{G^{(j)}} \coloneqq \bigcup_{k=j+1}^{s} G^{(k)}$.

We can see that, given an operator O_j supported on a set of sites $G^{(j)}$, the tensors that do not automatically simplify when we renormalise O_j, possibly up to the top of the network, are precisely those connected to the sites belonging to the past causal cone of $G^{(j)}$.

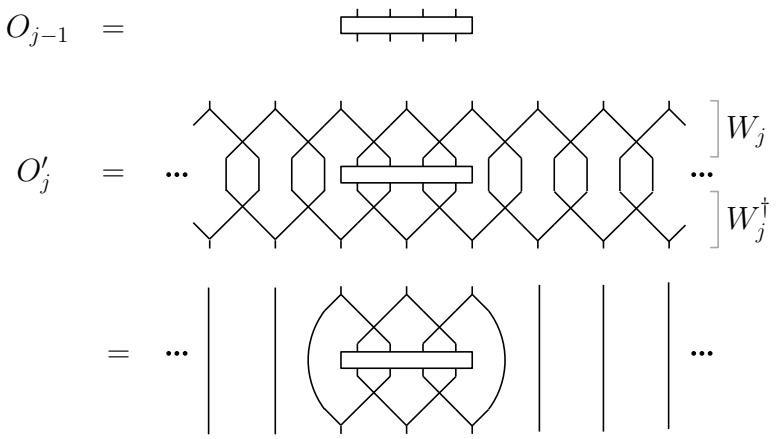

Figure 2.1.4: Given some operator O_{j-1} acting on level $j-1$, for example non-trivially supported on 4 qubits, we represent pictorially the computations needed to compute the renormalized operator $O'_j = W_j^\dagger O_j W_j$ at level j (vertical straight lines represent an identity). The upper layer in the second line is W_j and the lower one is $W_j^\dagger$. We observe that there are many immediate cancellations thanks to the fact that the building blocks of the network are unitaries or isometries.

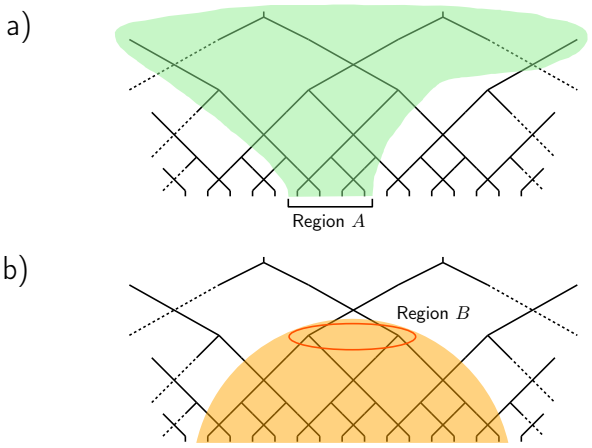

Figure 2.1.5: Examples a) of the past causal cone of region A and b) of the future causal cone of region B (the two qubits circled)

An important property of the past causal cone is that it shrinks exponentially in width, where the width is measured in terms of lattice spacing. This is related to the fact that the number of sites is exponentially decreasing level by level in the renormalization direction (because for example in a binary MERA two sites are mapped to one). However, because of the disentanglers, the past causal cone of a connected region A of $|A|$ sites does not really have size $|A|/2$ one level higher, but $|A|/2 + \mathcal{O}(1)$. Hence after s levels the width has become $|A|/2^s + \mathcal{O}(s)$. We will often ignore the correction $\mathcal{O}(s)$: if we want A to "collapse" to $\mathcal{O}(1)$ width, we do not really need just $\mathcal{O}(\log_2(|A|))$ renormalization steps but this number plus a few more. However, in the limit of large system sizes it is not so important.

In a similar way, we also define the

Definition 2.1.4 (Future causal cone)**.** Let $F^{(j)} = \{g_1^{(j)}, g_2^{(j)}, \dots\}$ be a set of sites belonging to the lattice at level j. For each site $g_i^{(j)} \in F^{(j)}$ consider the bonds of the tensor network connecting it to other sites $g_{i,1}^{(j-1)}, g_{i,2}^{(j-1)}, \dots$ one level lower in the network, and define a set $\tilde{F}_i^{(j-1)} = \{g_{i,1}^{(j-1)}, g_{i,2}^{(j-1)}, \dots\}$. Define $F^{(j-1)} \coloneqq \bigcup_i \tilde{F}_i^{(j-1)}$ and repeat the procedure for $F^{(j-1)}$ till the bottom of the network at level 0. The *future causal cone* $\mathfrak{F}_{F^{(j)}}$ of $F^{(j)}$ is defined as $\mathfrak{F}_{F^{(j)}} \coloneqq \bigcup_{k=0}^{j-1} F^{(k)}$.

Notice that, contrary to the past causal cone, the width of the future one *increases* exponentially going from the top to the bottom of the network.

2.2 MERA in condensed matter and quantum error correction

In the next paragraph we explain why MERA was developed and what the main ideas, making it a good approximation for certain classes of quantum many-body states, are. After that paragraph, we abandon this perspective for the remainder of this work and we explain how MERA can also represent a quantum error correcting code (we will further expand and formalise this at the beginning of chapter 3).

MERA as a multi-scale description of quantum many-body systems

Here we present MERA and the optimization of disentanglers and isometries as it has been done originally by Vidal [4]. The idea is that we have a quantum many-body system defined on a lattice and that we are given some state, generally the ground state of such a system. Suppose that we want to compute average values of observables, especially local observables. For many models one cannot get analytical solutions and so one has to exploit numerics. However, a pure state or more generally a density matrix of a many-body system is described by a number of coefficients whose number is exponential in the

system size, so that even just matrix multiplication is a task which requires a great computational power. Finding efficient representations for at least some states would be desirable, possibly at the cost of sacrificing a little bit of the precision with which one can compute the average values of interest for those states (but not too much).

In this picture MERA can be described as a *coarse-graining* map, or *renormalization group* transformation, mapping a state $|\psi\rangle$ on the physical lattice to a state $|\psi'\rangle$ on another one with fewer sites, representing the physical one looked from a larger scale. At the same time we are also going to renormalise operators from one scale to the other (the levels according to the terminology of the previous section). The coarse-graining transformation is required to have the property

$$\left|\langle\psi|O|\psi\rangle - \langle\psi'|O'|\psi'\rangle\right| \leq \epsilon, \tag{2.5}$$

where $|\psi\rangle', O'$ are the renormalised objects and ϵ should be "small" w.r.t. the precision with which we want to compute $\langle\psi|O|\psi\rangle$ for the observables O of interest. Moreover, we want that the approximation remains small under successive iteration of the renormalization procedure, till we get a lattice with so few sites that we can easily carry out computations with the coarse-grained state.

Consider a 1-dimensional physical lattice $\mathcal{L}$, with Hilbert space

$$\mathbb{V}_{\mathcal{L}} = \bigotimes_{s\in\mathcal{L}} \mathbb{V}_s \tag{2.6}$$

where s labels the sites in $\mathcal{L}$ and $\mathbb{V}_s$ is a finite dimensional Hilbert space. Let $|\psi\rangle \in \mathbb{V}_{\mathcal{L}}$. Group neighbouring sites in $\mathcal{L}$ in blocks of m sites (in the following we will always use $m = 2$, but the construction can be done with arbitrary m). Call $\mathcal{B}_{s'}$ one of such blocks, labelled by a block index s', and let $\mathbb{V}_{\mathcal{B}_{s'}} = \bigotimes_{s\in\mathcal{B}_{s'}} \mathbb{V}_s$.

We want to define a new lattice $\mathcal{L}'$ with

$$\mathbb{V}_{\mathcal{L}'} = \bigotimes_{s'\in\mathcal{L}'} \mathbb{V}'_{s'} \tag{2.7}$$

and an isometry

$$w\colon \mathbb{V}'_{s'} \to \mathbb{V}_{\mathcal{B}_{s'}}, \quad w^\dagger w = \mathbb{1}_{\mathbb{V}'_{s'}} \tag{2.8}$$

with image $w(\mathbb{V}'_{s'}) =: \mathbb{S}_{\mathcal{B}_{s'}} \subset \mathbb{V}_{\mathcal{B}_{s'}}$, where $\mathbb{V}'_{s'}$ is a Hilbert space with dimension $\dim \mathbb{V}'_{s'} \leq (\dim \mathbb{V}_s)^{|\mathcal{B}_{s'}|}$. The crucial point is actually the one of choosing the subspace $\mathbb{S}_{\mathcal{B}_{s'}}$ in an appropriate way. In order to get a good approximation, we need states in $\mathbb{S}_{\mathcal{B}_{s'}}$ to be "good" representatives of $|\psi\rangle$. But this is not enough: notice that the dimension of $\mathbb{S}_{\mathcal{B}_{s'}}$ can be as large as $(\dim \mathbb{V}_s)^{|\mathcal{B}_{s'}|}$: if this is the case, we can rewrite $|\psi\rangle$ at another scale, using much fewer sites, but *all* the information and so *all* the complexity of the computational problem would still be there, hidden in the dimension of the Hilbert space of those few sites. Therefore, another constraint which should be imposed on the renormalization

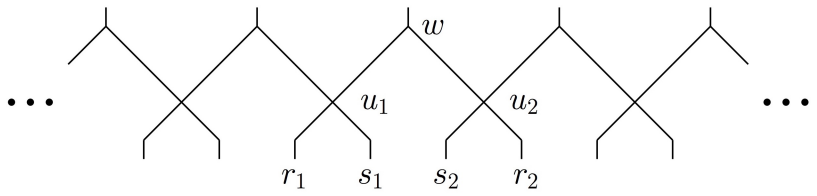

Figure 2.2.1: Instead of acting directly with the isometry w over the sites s_1 and s_2, the idea of Vidal was to insert the disentanglers u_1 and u_2 in between.

map is that the dimension of $\mathbb{S}_{\mathcal{B}_{s'}}$ does not grow "too much" or that it remains possibly constant.

Now there are two main ideas in the MERA construction. The first one comes from White's density matrix renormalization group (DMRG) [23, 24]: consider the reduced density matrix $\rho_{\mathcal{B}_{s'}} = \mathrm{Tr}_{\mathcal{L}\setminus\mathcal{B}_{s'}} |\psi\rangle \langle\psi|$ over a block and diagonalise it:

$$\rho_{\mathcal{B}_{s'}} = \sum_k p_k |\phi_k\rangle \langle\phi_k|_{\mathcal{B}_{s'}} . \tag{2.9}$$

Order the eigenvalues $\{p_k\}$ in decreasing order, $p_0 \geq p_1 \geq \ldots$, and consider the N highest ones such that

$$\sum_{k=0}^{N-1} p_k \geq 1 - \epsilon \tag{2.10}$$

for some truncation error ϵ fixed in advance. Then $\mathbb{S}_{\mathcal{B}_{s'}}$ is defined as the span of the N eigenvectors corresponding to such eigenvalues.

However, it was noticed that there were cases in which this approach was not enough to get a good approximation because of the accumulation of local entanglement under multiple renormalization steps [4], so the second main idea in MERA is to introduce disentanglers, as proposed by Vidal in the same paper. For simplicity we consider a transformation in 1D in which 2 sites are renormalised to 1 by an isometry $w\colon \mathbb{V}_{s_1} \otimes \mathbb{V}_{s_2} \to \mathbb{V}_{\mathcal{B}_{1'}}$. Let r_1, s_1, s_2, r_2 be four contiguous sites, as in figure 2.2.1, and let $\rho_{\{r_1,s_1,s_2,r_2\}} = \mathrm{Tr}_{\mathcal{L}\setminus\{r_1,s_1,s_2,r_2\}} |\psi\rangle \langle\psi|$. We require the disentanglers

$$u_1\colon \mathbb{V}_{r_1} \otimes \mathbb{V}_{s_1} \to \mathbb{V}_{r_1} \otimes \mathbb{V}_{s_1} \tag{2.11}$$

$$u_2\colon \mathbb{V}_{s_2} \otimes \mathbb{V}_{r_2} \to \mathbb{V}_{s_2} \otimes \mathbb{V}_{r_2} \tag{2.12}$$

to be such that the $\tilde{N}$ highest eigenvalues of

$$\tilde{\rho}_{\{s_1,s_2\}} = \mathrm{Tr}_{\{r_1,r_2\}}\Big((u_1 \otimes u_2)\rho_{\{r_1,s_1,s_2,r_2\}}(u_1^\dagger \otimes u_2^\dagger)\Big) \tag{2.13}$$

satisfy

$$\sum_{k=0}^{\tilde{N}-1} \tilde{p}_k \geq 1 - \epsilon \tag{2.14}$$

where $\tilde{N} < N$ and N is the same as in equation (2.10) with $\mathcal{B}_{s'} = \{s_1, s_2\}$.

We can see that N or $\tilde{N}$ are related to the *Schmidt rank* of $|\psi\rangle$, which can be taken as a measure of entanglement which can be easily computed numerically. Call $A = \{s_1, s_2\}$, $B = \{r_1, r_2\}$ and $C = \mathcal{L} \setminus AB$. Then $|\psi\rangle$ has a Schmidt decomposition

$$|\psi\rangle = \sum_k \sqrt{p_k}\, |\phi_k\rangle_A \otimes |\varphi_k\rangle_{BC} \tag{2.15}$$

for some vectors $\{|\varphi_k\rangle_{BC}\}$ and the number of non-zero terms in the sum is called the Schmidt rank. Note that if the Schmidt rank is 1, then there is no entanglement between A and BC, and that is the reason why the Schmidt rank can be taken as an entanglement measure (one could consider the entanglement entropy of ρ_A, but the condition above often seems to be more convenient in practice). Hence $\tilde{N} < N$ often corresponds to a less entangled state between A and BC.

MERA as a quantum error correcting code

Given a state $|\psi\rangle$ of some quantum system, it is not clear in general how to choose appropriate disentanglers (and isometries), so that many algorithms (for example [25]) have been developed to optimize their choice. In this work we are not going to take this point of view, but we will choose some disentanglers and isometries (or assume that they satisfy some properties) and then check what the quantum error correcting capabilities of the corresponding MERA are, or more precisely of the subspace of states it represents. We will consider a MERA with s levels and with qubits at each site: we call *physical qubits* the n ones at the bottom and *logical qubits* the k at the top.

Therefore, whereas in the previous paragraph we wanted to find disentanglers such that (2.5) was satisfied, here we are interested precisely in those states for which $\epsilon = 0$, i.e. those for which

$$\langle\psi|O|\psi\rangle = \langle\psi'|O'|\psi'\rangle \tag{2.16}$$

It is easy to characterize such states. As we have already said, the j-th level of the MERA is just an isometry $W_j \colon \mathcal{H}_j \to \mathcal{H}_{j-1}$ overall, mapping the Hilbert space $\mathcal{H}_j$ of the lattice at level j to the Hilbert space $\mathcal{H}_{j-1}$ one level lower. For a MERA with a total of s levels, call $W \colon \mathcal{H}_s \cong (\mathbb{C}^2)^{\otimes k} \to \mathcal{H}_0 \cong (\mathbb{C}^2)^{\otimes n}$ the map $W \coloneqq W_1 \dots W_s$. Then for a state $|\varphi\rangle \in \mathcal{H}_s$, we define $|\bar{\varphi}\rangle \coloneqq W\,|\varphi\rangle$. We can easily check that these are the states we are looking for:

$$\langle\bar{\varphi}'|O'|\bar{\varphi}'\rangle = \langle\bar{\varphi}|WW^\dagger OWW^\dagger|\bar{\varphi}\rangle \tag{2.17}$$

$$= \langle\varphi|\underbrace{W^\dagger W}_{\mathbb{1}_{\mathcal{H}_s}} W^\dagger OW \underbrace{W^\dagger W}_{\mathbb{1}_{\mathcal{H}_s}}|\varphi\rangle = \langle\bar{\varphi}|O|\bar{\varphi}\rangle\,. \tag{2.18}$$

If we now take a density matrix ρ supported on $\mathcal{H}_s$, it is mapped to a density matrix on $\mathcal{H}_0$ according to $\rho \mapsto W\rho W^\dagger$.

Definition 2.2.1 (Encoding map)**.** We call the map $\mathcal{E}\colon \mathcal{B}(\mathcal{H}_s) \to \mathcal{B}(\mathcal{H}_0)$, defined as

$$\mathcal{E}(\rho) = W\rho W^\dagger, \tag{2.19}$$

the *encoding map*.

Notice that $\mathcal{E}$ is an isometric quantum channel and hence it can be used as an encoder for a quantum error correcting code (which is its image) according to definition 1.1.2. Moreover, the renormalization map $\mathcal{E}^*$ in definition 2.1.2 is just the dual channel (definition A.0.2) of $\mathcal{E}$, and this justifies our choice of notation for $\mathcal{E}^*$.

It is interesting to consider the physical implementation of the isometries in the MERA, in order to realise that we can view MERA as a quantum circuit (see figure 2.1.3), mapping the logical states at the top (input of the circuit) to encoded physical states at the bottom (output). Indeed, as we have already said in section 2.1, an isometry V can be viewed as a unitary U_V acting on a bigger space, where one of the input states is fixed: $V(\cdot) = U_V(\,\cdot \otimes |0\rangle)$. Hence, considering the unitary extension of all the isometries in the MERA, we get a (unitary) quantum circuit of depth s and width n, which is what we called the *unitary extension* of the MERA (definition 2.1.3).

We stress that every isometric quantum channel can be seen as a quantum circuit and that its dual may be called a "renormalization map". The point is that for a MERA they have a peculiar structure, and that will allow us to prove some results in the following (picking a few more assumptions).

2.3 Examples: the Ising chain and the toric code

The Ising chain

Consider a 1D chain of $n = 2^s$ qubits (spin 1/2 particles), with periodic boundary conditions. The Hamiltonian

$$H = -\sum_{i=0}^{n-1} Z_i Z_{i+1}, \tag{2.20}$$

where Z_i is Pauli Z acting on the i-th qubit and where we set $Z_n \equiv Z_0$, has a 2-dimensional degenerate ground subspace spanned by the states

$$|000\ldots\rangle\,,\ |111\ldots\rangle\,. \tag{2.21}$$

We can see that this subspace admits a simple MERA representation, drawn in figure 2.3.1, where the disentanglers are simply the identity. In this case the representation is called a *tree tensor network*. The unitary extension of the isometries is a CNOT with the left qubit always chosen to be the control qubit and the right one the target qubit. The isometries are then given by such a unitary with the target qubit at the higher level fixed to $|0\rangle$. In the following, we refer to the unitary extension of the Ising MERA.

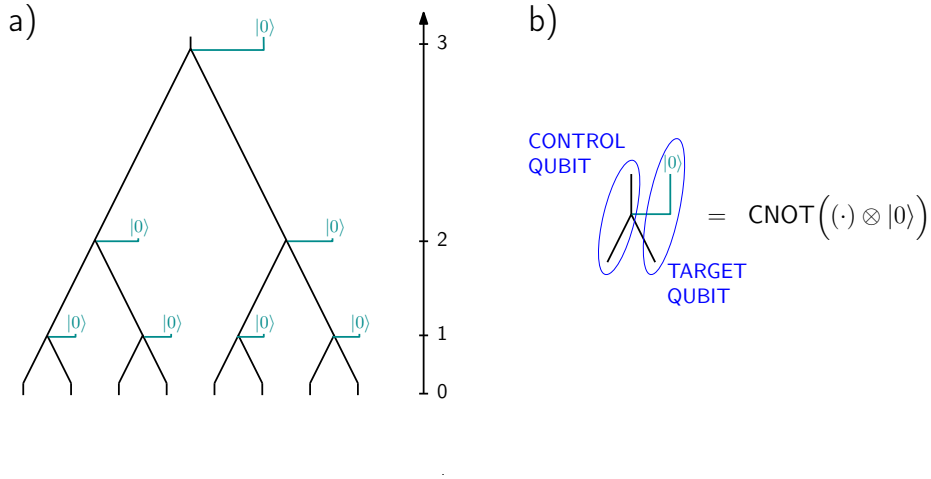

Figure 2.3.1: a) MERA representation (unitarily extended) of the Ising chain ground subspace. The state $|000\ldots\rangle$ (all spins up) is obtained by picking $|0\rangle$ as input state for the top free index, whereas $|111\ldots\rangle$ is obtained picking $|1\rangle$. b) The building block of the Ising MERA is a CNOT as shown here.

It will be useful to consider the action that conjugation by CNOT has on Pauli operators. Notice that, in order to specify its action on any Pauli operator, it is enough to specify it on a set of generators of the Pauli group (definition 1.5.1):

$$\mathbb{1} \otimes Z \mapsto Z \otimes Z \tag{2.22}$$
$$Z \otimes \mathbb{1} \mapsto Z \otimes \mathbb{1} \tag{2.23}$$
$$\mathbb{1} \otimes X \mapsto \mathbb{1} \otimes X \tag{2.24}$$
$$X \otimes \mathbb{1} \mapsto X \otimes X, \tag{2.25}$$

where the first qubit is the control qubit and the second one is the target qubit.

We can also rephrase the setting in terms of stabilizer codes (section 1.5). Indeed, the ground subspace of H is a stabilizer code, where the stabilizer generators are given by $\{Z_i Z_{i+1}\}_{i=0}^{n-2}$. Notice that we keep one term less than the ones in the Hamiltonian because, thanks to the periodic boundary conditions, we get the constraint

$$\prod_{i=0}^{n-1} Z_i Z_{i+1} = I^{\otimes n}, \tag{2.26}$$

which means that there are $n-1$ *independent* generators, so that the dimension of the code subspace is $2^{n-(n-1)} = 2$, as it should be.

Therefore, we have to show that states encoded by the above MERA are eigenstates with eigenvalue $+1$ of all the stabilizer generators. To do that,

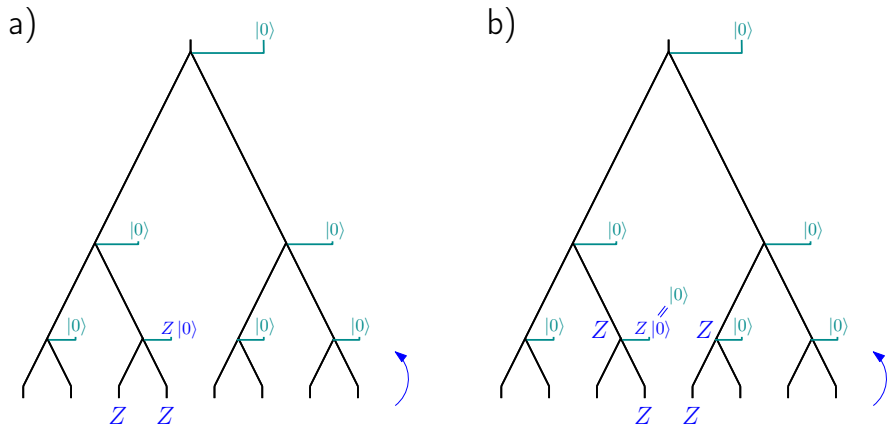

Figure 2.3.2: a) ZZ as shown is renormalized to a single Z on one of the decoupled qubits because CNOT $(Z \otimes Z)$ CNOT $= I \otimes Z$. b) Similarly, ZZ in the other position is renormalized to ZZZ, but $Z|0\rangle = |0\rangle$ and so we can say that it is effectively renormalized to ZIZ.

recall the so-called Heisenberg representation for quantum circuits [26]: consider a unitary U and an arbitrary operator N; for any state $|\psi\rangle$ we have the trivial identity:

$$UN|\psi\rangle = (UNU^\dagger)U|\psi\rangle \tag{2.27}$$

which means that to know the action of N on $|\psi\rangle$, followed by U, it is enough to know the action of $UNU^\dagger$ on $U|\psi\rangle$ (it can be easier!). In particular, $|\psi\rangle$ is an eigenvector of N if and only if $U|\psi\rangle$ is an eigenvector of $UNU^\dagger$. Here we take N as one of the generators and U as the unitary mapping the physical level to level 1, i.e. as the adjoint of the unitary extension of W_1. From the transformation shown in figure 2.3.2, we can see that half of the generators are mapped to a single qubit Z on one of the decoupled $|0\rangle$ states, which is their $+1$-eigenstate. Any of the remaining generators is similarly mapped to a single qubit Z at some higher level, hence this proves our statement.

The toric code

We can now understand the more complicated example of the toric code [2], whose MERA representation has been first given in [6] by Aguado and Vidal. In this case the network is 2-dimensional and each disentangler and isometry acts on a block of 16 qubits. There will be also a non-trivial top tensor.

Recall that the toric code is a stabilizer code defined in this way:

- Lattice: qubits are arranged on an $L \times L$ square lattice and they occupy the edges; moreover, we take periodic boundary conditions in all direc-

tions, which means that the lattice has overall the geometry of a torus and there are $2L^2$ qubits in total;

- Stabilizer generators: we divide them into two classes. The *star operators* are products of 4 Pauli X acting on the 4 qubits connected to a vertex of the lattice, whereas the *plaquette operators* are products of 4 Pauli Z acting on the 4 qubits laying on the edges of one of the small squares forming the lattice, called a plaquette. Thanks to the boundary conditions, the product of all star operators is the identity and the same holds for the plaquette operators, hence there are $2L^2 - 2$ independent generators, which means 2 encoded qubits.

It is easier to understand the construction of this example if we describe it in the encoding direction, i.e. if we iteratively add new levels at the bottom of the MERA network. We want each level to represent a valid toric code, satisfying all the required star and plaquette operator constraints. Moreover, logical operators (non-trivial cycles around the torus) should be mapped to logical operators.

Following figure 2.3.3, we start with a toric code with lattice spacing $2a$ and we add qubits in decoupled states $|0\rangle$ and $|+\rangle$ forming a new lattice with lattice spacing a (we consider $|+\rangle$ states for convenience but we may just take $|0\rangle$'s and add a Hadamard transformation to the isometries). Then we need to entangle those qubits with the old ones in an appropriate way; for that we use disentanglers and isometries constructed just from CNOT's as in figure 2.3.4. Notice that the $|0\rangle$ and $|+\rangle$ states are stabilized by Z and X respectively. The CNOT always maps Pauli operators to Pauli operators, as it is clear from (2.22), so those trivial stabilizers will be mapped to new stabilizers on the new lattice. We can compute them by repeated conjugation by CNOTs, getting table 2.3.1. Some X's are mapped to products of two star operators but it is not a problem because there is always another X which is mapped to one of the two factors. One can also check that the mapping of the old toric code stabilizers gives exactly those which together with the ones in table 2.3.1 form a complete set of star and plaquette operators for the new toric code. Moreover, it also holds that logical operators are mapped to logical ones [6].

Notice that such a construction works for a minimum of 16 qubits, i.e. we need to start from a toric code defined on a 4×4 lattice in order to encode it in larger and larger lattices. Therefore, we need to fix the 14 stabilizer generators at the top and restrict to states which are their common $+1$ eigenspace. Equivalently, in terms of MERA we can say that we need a top tensor given by a 16-body unitary which has 14 of its inputs fixed to some decoupled state that can always be taken to be $|0\rangle$. Because we have to remain with two qubits at the top, this tensor cannot be given by a combination of the disentanglers and isometries composing the rest of the network, but its exact form does not really matter and we can just talk about the stabilizers that are fixed at the top, whenever it is more convenient.

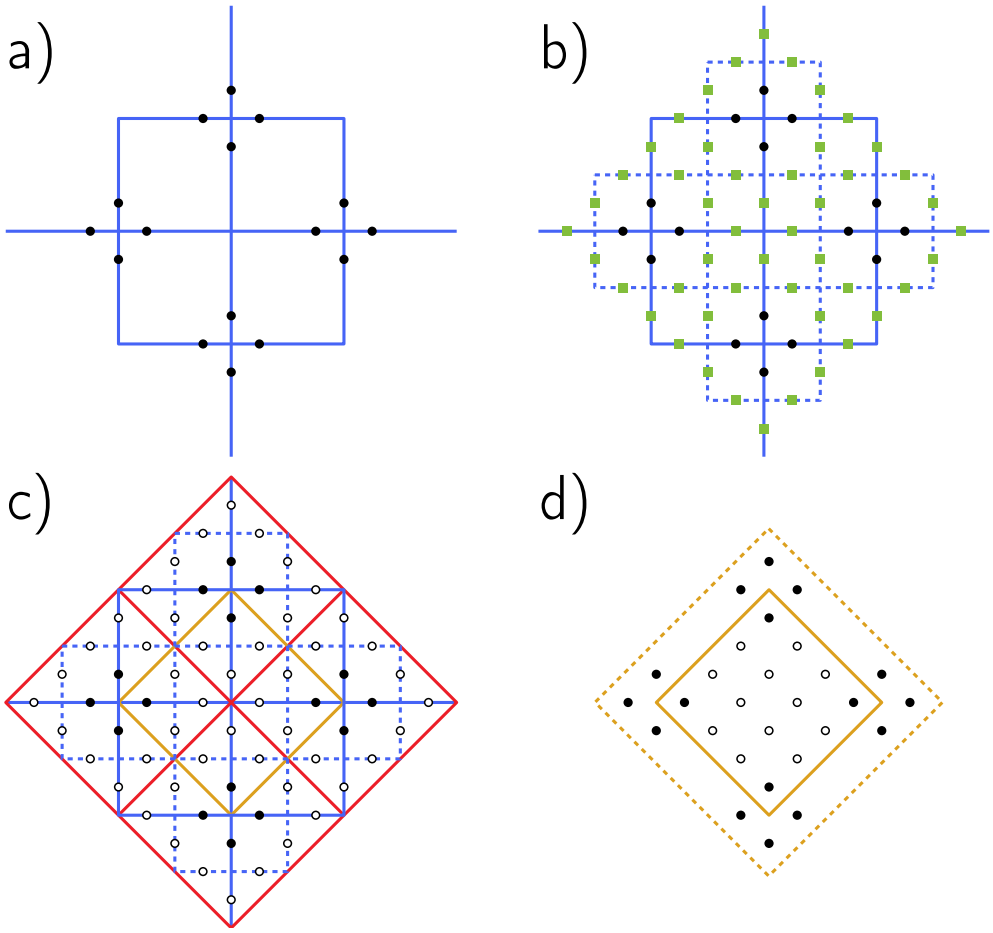

Figure 2.3.3: a) We represent just a part of a square lattice over which we have a toric code. The full-black qubits are not evenly spaced in the figure for convenience. b) New ancillas (in green) are inserted in state either $|0\rangle$ or $|+\rangle$, in such a way to form a new lattice with the lattice spacing halved (in [6] it is specified which ones are added in state $|0\rangle$ or $|+\rangle$). c) In the encoding direction, we apply first the four tilted red squares representing the isometries, and then we apply the disentanglers. We draw just one in the middle in orange but there are also others bordering with it. d) We choose to put the disentangler at a higher level as represented here. Remember that at the higher level the decoupled ancillas are removed and there are only the full-black qubits left.

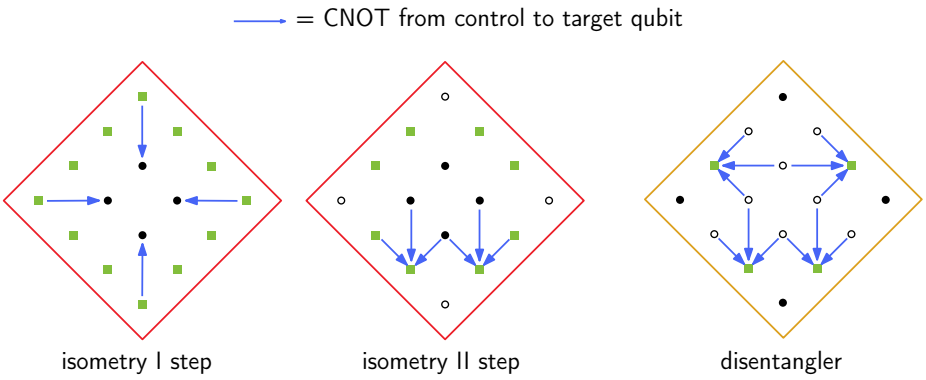

Figure 2.3.4: Isometry and disentangler for the toric code in the encoding direction. Notice that the two steps of the isometry do not commute. In the renormalization direction they are reversed and, as a last step, the ancillas are projected onto the "original" states $|0\rangle$ and $|+\rangle$.

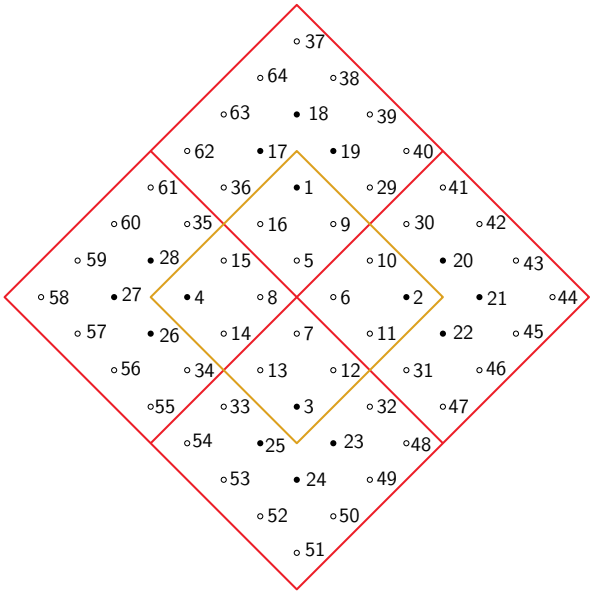

Figure 2.3.5: Numbering of the qubits we use throughout this work for the toric code.

Table 2.3.1: This table shows where the trivial stabilizers of the decoupled ancillas $|0\rangle$ and $|+\rangle$ are mapped, one level lower in the network. The numbering refers to figure 2.3.5.

Qubit number	Stabilized by	Mapped to	Notes
5	X	$X_1X_5X_9X_{16}$	Star op.
6	X	$X_6X_{10}X_2X_{31}X_{32}X_{12}$	Product of star op.
7	X	$X_7X_{12}X_3X_{13}$	Star op.
8	X	$X_4X_{15}X_8X_{13}X_{33}X_{34}$	Product of star op.
9	Z	$Z_1Z_{19}Z_9Z_{29}$	Plaquette op.
10	Z	$Z_{10}Z_5Z_9Z_6$	Plaquette op.
11	X	$X_{11}X_{12}X_{32}X_{31}$	Star op.
12	Z	$Z_{12}Z_7Z_6Z_{11}$	Plaquette op.
13	Z	$Z_{13}Z_{14}Z_8Z_7$	Plaquette op.
14	X	$X_{14}X_{13}X_{33}X_{34}$	Star op.
15	Z	$Z_{15}Z_{16}Z_5Z_8$	Plaquette op.
16	Z	$Z_1Z_{17}Z_{36}Z_{16}$	Plaquette op.

Remark 2.3.1. There is no need to fix the position of the disentanglers at a higher level relative to the position of the disentanglers one level lower to achieve the right mapping (except for the fact that the disentanglers should be centered on a vertex of the lattice). However, as a convention we decide to put the new disentangler centered on top of the old one (see figure 2.3.3d). We will make some statements that depend on this choice in their specific content, but not qualitatively.

Chapter 3

MERA Codes: Correctability of Erasure Noise

Here we present and discuss some of the results which have been derived by Kim and Kastoryano in [7]. They consider families of approximate quantum error correcting codes where the encoder is given by certain classes of MERA and, exploiting the local structure of the tensor network, they prove that erasure errors involving not too large regions are locally correctable in the sense of section 1.4. A lower bound on the code distance is then derived. We stress that here the notion of correctability refers to erasure errors, while the performance of the code against arbitrary errors cannot be probed with the following methods.

First we re-introduce and further formalise the quantum error correction (QEC) perspective on MERA. We saw in section 2.1 that in condensed matter physics MERA has been introduced as a way to efficiently represent a quantum many-body state (containing a not-too-high degree of entanglement) by repeated coarse-graining of the lattice structure, while preserving the relevant properties of the state of interest – specifically, average values of local observables. Given a state $|\psi\rangle$ on the physical Hilbert space and an operator O acting on it, and denoting by $|\psi'\rangle$ and O' the corresponding renormalized objects, the hope in condensed matter physics is that $\langle\psi|O|\psi\rangle \cong \langle\psi'|O'|\psi'\rangle$ if the MERA representing $|\psi\rangle$ was properly constructed and optimized.

In QEC we are interested precisely in those states for which equality holds. We will consider, consequently, not MERAs representing a single state, as it is often the case, but MERAs representing entire subspaces, leaving free k indices at the top. When we look at MERA as a quantum circuit, we use those indices as input and then the circuit outputs an encoded state on the physical indices at the bottom.

To be more concrete, referring to figure 2.1.1, we consider a system of size n where the physical degrees of freedom are taken to be qubits (it is enough that the dimension of the local Hilbert space is finite, but for simplicity we will consider qubits). We introduce a renormalization scale from 0 to

$s = \log_2(n/k)$, where we refer to 0 as the *physical level* or bottom of the network, whereas s is the *logical level* or top of the network. A full layer of disentanglers and isometries is just an isometry overall, and we will denote it by W_j, mapping level j to level $j-1$. Then the MERA when it is run from top to bottom is just $W \coloneqq W_1 \dots W_s$, mapping logical states in $\mathcal{H}_s$ to physical states in $\mathcal{H}_0$. If we consider density matrices, then such a mapping is given by $\mathcal{E}(\rho) \coloneqq W\rho W^\dagger$, which is an isometric quantum channel that we call the encoding map (definition 2.2.1).

Thus we can consider quantum error correcting codes (QECC) where the encoder $\mathcal{E}$ has the structure of a MERA. The code subspace $\mathcal{C}_s \subset \mathcal{H}_0$ is then the image of $\mathcal{H}_s$ under W, i.e. $\mathcal{C}_s = \{W|\psi_s\rangle = W_1 \dots W_s |\psi_s\rangle : |\psi_s\rangle \in \mathcal{H}_s\}$. Moreover, it follows then that the renormalization map $\mathcal{E}^*$, which is given by $\mathcal{E}^*(\cdot) = W^\dagger(\cdot)W$ (definition 2.1.2), is just the dual of the encoder and so it can be used to track, up to the logical level, the action of the operators (the "errors") acting on the physical system. In particular, notice that if $|\psi\rangle \in \mathcal{C}_s$, then it can be renormalised exactly, in the sense that there exists a $|\psi_s\rangle$ such that $|\psi\rangle = W|\psi_s\rangle$ and

$$\langle\psi|O|\psi\rangle = \langle\psi_s|W^\dagger O W|\psi_s\rangle = \langle\psi_s|\mathcal{E}^*(O)|\psi_s\rangle = \langle\psi_s|O^{(s)}|\psi_s\rangle \tag{3.1}$$

where we defined $\mathcal{E}^*(O) = O^{(s)}$. Equation (3.1) means that the average value of O is the same if we compute it on the physical level or using the renormalized quantities on the logical one. In general we use the notation $(\mathcal{E}^*)_{s_1}^{s_2}(O^{(s_1)}) = O^{(s_2)}$ to indicate the renormalization of an operator acting on level s_1 up to level $s_2 \geq s_1$.

Definition 3.0.1 (MERA codes)**.** A MERA code is a QECC where the encoder is given by an isometry W which has the structure of a MERA.

3.1 Elementary blocks and transfer operator(s)

Following [7], we consider a MERA with a uniform structure at all length scales, which will simplify the analysis and allow for an easy construction of MERAs of the same type but with a different number of levels, which corresponds to a family of QECCs for systems of larger and larger sizes.

Definition 3.1.1 (Scale-invariant MERA)**.** A MERA is *scale-invariant* if all the disentanglers at all length scales are equal to each other and if the same holds also for the isometries.

We point out that the fact that the tensor network is scale invariant does not mean that it describes a critical system: the toric code [2, 3] has a scale-invariant representation (section 2.3, [6]) but it corresponds to a system which is gapped. We may say, however, that a system represented by a scale-invariant MERA is a scale-invariant system.

As we noticed in section 2.1, the size of the past causal cone (definition 2.1.3) corresponding to some region of the physical lattice shrinks exponentially till it becomes of $\mathcal{O}(1)$ size and then it remains constant. Based on this observation we define:

Definition 3.1.2 (Elementary blocks)**.** Let B be a block of contiguous sites at some level in the MERA. Consider all the sites one level higher in the network which are connected to sites in B via the bonds of isometries and disentanglers. If they form a block B' of contiguous sites such that $|B| = |B'|$, then we say that B is an elementary block.

For an operator $O_{elem} \colon \mathcal{H}_{B_{el}} \to \mathcal{H}_{B_{el}}$ supported on an elementary block B_{el} we can write its extension on the full lattice as $\mathbb{1} \otimes \ldots \otimes \mathbb{1} \otimes O_{elem} \ldots \otimes \mathbb{1}$, i.e. we add as many identity tensor product factors as the size of the lattice minus the size of B_{el}. Call L_j the lattice at level j and let $|L_j|$ its size (number of qubits forming such a lattice). Then for every operator O_{elem} there is an operator $O'_{elem} \colon \mathcal{H}_{B'_{el}} \to \mathcal{H}_{B'_{el}}$ such that

$$(\mathcal{E}^*)_{s_1}^{s_1+1}(\mathbb{1} \otimes \ldots \otimes \mathbb{1} \otimes O_{elem} \ldots \otimes \mathbb{1}) = \mathbb{1} \otimes \ldots \otimes \mathbb{1} \otimes O'_{elem} \ldots \otimes \mathbb{1} \tag{3.2}$$

where $(\mathcal{E}^*)_{s_1}^{s_1+1} \colon \mathcal{B}((\mathbb{C}^2)^{\otimes |L_{s_1}|}) \to \mathcal{B}((\mathbb{C}^2)^{\otimes |L_{s_1+1}|})$ is the renormalization map from level s_1 to $s_1 + 1$. Remark: the number of identities on the r.h.s. and l.h.s. is different.

It is convenient to define what we will call

Definition 3.1.3 (Transfer operator)**.** Given an elementary block B_{el}, we define the transfer operator associated to that elementary block as the map ϕ such that

$$(\mathcal{E}^*)_{s_1}^{s_1+1}(\mathbb{1} \otimes \ldots \otimes \mathbb{1} \otimes O_{elem} \ldots \otimes \mathbb{1}) = \mathbb{1} \otimes \ldots \otimes \mathbb{1} \otimes \phi(O_{elem}) \ldots \otimes \mathbb{1} \tag{3.3}$$

$\forall O_{elem}$ supported on B_{el}. If we want to stress between which levels we are applying the renormalization procedure we write $\phi_{s_1}^{s_1+1}$ and in general we define $\phi_{s_1}^{s_2} \coloneqq \phi_{s_2-1}^{s_2} \circ \ldots \circ \phi_{s_1}^{s_1+1}$.

Remark 3.1.1. Notice that it is possible that there exist elementary blocks of different sizes and that, even if the size is the same, they can be of different "types" (see figure 3.1.1 and 3.1.2 for an example). This happens because in general we do not assume translational invariance. Hence, we cannot assume that there is only one transfer operator, but we assume in the following that this is the case to keep the discussion as simple as possible. We will come back in the following to the general issue. It is even possible that one type of elementary block is renormalized to another type.
Moreover, notice that there is not even translational invariance by a number of sites equal to the size of the support of a disentangler. We can realize it from figure 3.1.1: for example, not all physical qubits on the left leg of a disentangler at the bottom level are the "same", because they can have a

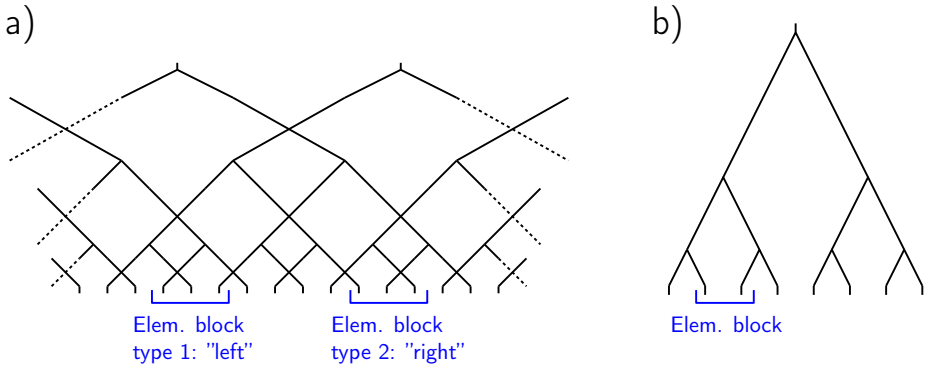

Figure 3.1.1: a) For a binary MERA we have two types of elementary blocks, b) for the Ising MERA just one.

different "stack" of disentanglers on top of them, i.e. a different "history" in terms of the encoding. However, depending on the properties and symmetries of disentanglers and isometries it is possible for some MERAs to represent a translationally-invariant system, as it is the case for the Ising chain or the toric code.

Properties and assumptions on the transfer operator

Here we want to make clear which the properties of ϕ are that follow from our definitions and which the extra assumptions are that we will make, following [7]. First recall that because $\mathcal{E}$ is a quantum channel it follows that $\mathcal{E}^*$ is

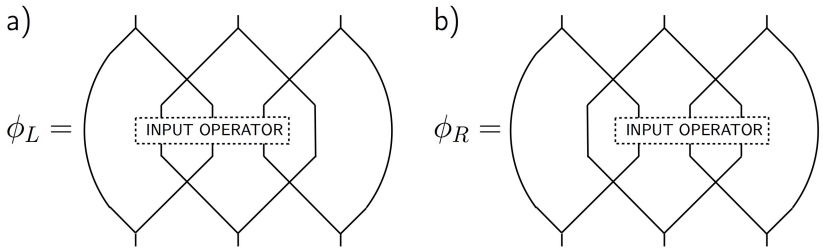

Figure 3.1.2: a) There are two renormalization maps associated with the two types of elementary block in figure 3.1.1a (we obtain these diagrams in a similar way to figure 2.1.4). We call them ϕ right and left. Remember that they pick an operator as input and give as output another operator supported on the same number of sites.

a completely positive and unital ($\mathcal{E}^*(\mathbb{1}) = \mathbb{1}$) linear map. It is clear from the definition that ϕ inherits the same properties from $\mathcal{E}^*$.

At this point it is important the fact that, by definition, ϕ has input and output spaces of the same dimension, which allows us to potentially diagonalize ϕ when we write it as a matrix (see appendix A.1 for a more detailed discussion). In particular we know that the spectral radius is 1, being ϕ positive and unital, and that all eigenvalues are real or come in complex conjugate pairs. We also know that eigenvalues of modulus 1 have trivial Jordan blocks. For all the other eigenvalues we have to assume that the same happens. This is the first assumption we make, but see appendix A.1 to check how it can be relaxed without changing much for the following. Then we can write

$$\phi(O) = \sum_{k=0}^{d_{el}^2-1} \lambda_k \operatorname{Tr}(R_k^\dagger O) L_k \tag{3.4}$$

where d_{el} is the dimension of the Hilbert space of an elementary block B_{el} and

1. $\{R_k^\dagger\}$ and $\{L_k\}$ are bi-orthonormal basis of right and left eigenvectors (w.r.t. the Hilbert-Schmidt inner product), i.e. $\langle R_l, L_k \rangle = \operatorname{Tr}(R_l^\dagger L_k) = \delta_{lk}$, $\phi(L_k) = \lambda_k L_k$ and $\phi^*(R_k^\dagger) = \lambda_k R_k^\dagger$. Notice that the two families are not separately orthonormal;

2. $L_0 = \mathbb{1}$ and $\lambda_0 = 1$ because ϕ is unital. Consequently, it follows from the bi-orthonormality condition that $\operatorname{Tr}(R_k) = \delta_{k,0}$;

3. $|\lambda_k| \leq 1 \quad \forall k$.

Furthermore, we make the assumption that there is only one eigenvector with eigenvalue 1 (namely the identity because ϕ is unital), i.e. that $|\lambda_k| < 1$ for $k \neq 0$. This is the crucial assumption. One thing which follows from this (see appendix A.1) is that R_0 is a density matrix and in particular that it is the unique stationary state of the quantum channel ϕ^*. Notice that this is not the same channel as $\mathcal{E}$, even when restricted on elementary blocks, because in general $\mathcal{E}$ maps to a region which is larger than the elementary block.

Definition 3.1.4 (K&K assumptions)**.** Given a MERA with transfer operator ϕ, by Kim & Kastoryano (K&K) assumptions we mean the following two conditions on ϕ:

1. ϕ takes the form of equation (3.4), i.e. it is diagonalizable;

2. $|\lambda_k| < 1$ for $k \neq 0$.

Remark 3.1.2. We stress that ϕ does not have to be trace-preserving and hence it is not necessarily a quantum channel given the above assumptions. However, this is not really important because we will not need the trace-preservation property anywhere. In any case, we can make precise when ϕ is trace-preserving and therefore a quantum channel:

Lemma 3.1.1. *Given all the above assumptions ϕ is trace-preserving if and only if $R_0 = \mathbb{1}/d_{el}$.*

Proof. Assume that ϕ is trace-preserving. Then we have that $\mathrm{Tr}(L_k) = \mathrm{Tr}(\phi(L_k)) = \lambda_k L_k$, which implies that $\mathrm{Tr}(L_k) = d_{el}\delta_{k,0}$. It follows that $\mathrm{Tr}(O) = \mathrm{Tr}(\phi(O)) = d_{el}\,\mathrm{Tr}(R_0^\dagger O) = d_{el}\,\langle R_0, O\rangle$, which holds $\forall O \in \mathcal{B}(\mathcal{H}_{el})$ if and only if $R_0 = \mathbb{1}/d_{el}$.
Conversely, assume that $R_0 = \mathbb{1}/d_{el}$. Then, from the bi-orthonormality condition we get $\mathrm{Tr}(L_k) = d_{el}\delta_{k,0}$. Trace preservation follows by taking the trace of (3.4). □

Definition 3.1.5 (Scaling dimension and RG-regular MERA)**.** Arranging the eigenvalues of ϕ in descending order in terms of their modulus, $1 = \lambda_0 \geq |\lambda_1| \geq |\lambda_2| \geq \ldots$, we define as *scaling dimension* of the MERA the parameter $\nu \coloneqq -\log_2(|\lambda_1|)$. We say that a MERA is *RG-regular* if ϕ satisfies the K&K assumptions. In particular in such a case we have that $\lambda_0 > \lambda_1$ and $\nu > 0$.

Definition 3.1.6 (Idealized K&K assumptions)**.** We say that the transfer operator ϕ satisfies the *idealized* K&K assumptions if it satisfies the K&K assumptions and $\lambda_k = 0$ for $k \neq 0$.

We will explain the choice of their name in section 3.5. Sometimes we will refer to them as the $\nu \to \infty$ assumptions because they correspond to the case in which "$\nu = +\infty$".

Useful norm inequalities

Here we will derive some inequalities which we will use in chapter 4. The first one is that $\forall O \in \mathcal{B}(\mathcal{H}_{el})$

$$\|\phi(O)\| = \|\mathcal{E}^*(\cdots \otimes \mathbb{1} \otimes O \otimes \mathbb{1} \otimes \ldots)\| \tag{3.5}$$

$$= \left\|W^\dagger(\cdots \otimes \mathbb{1} \otimes O \otimes \mathbb{1} \otimes \ldots)W\right\| \tag{3.6}$$

$$\leq \left\|W^\dagger\right\| \|O\| \|W\| = \|O\|\,. \tag{3.7}$$

Hence ϕ is norm non-increasing. Second, we have the following result:

Theorem 3.1.1. *Consider ϕ as in* (3.4) *and define*

$$\tilde{\phi}(O) \coloneqq \phi(O) - \mathrm{Tr}(R_0 O)\,\mathbb{1} = \sum_{k=1}^{d_{el}^2-1} \lambda_k\,\mathrm{Tr}(R_k^\dagger O)\,L_k, \tag{3.8}$$

where $O \in \mathcal{B}(\mathcal{H}_{el})$. Then

$$\left\|\tilde{\phi}(O)\right\| \leq c\,2^{-\nu}\,\|O\| \tag{3.9}$$

where ν is defined as above in definition (3.1.5) *and*

$$c = \sum_{k\neq 0} \|R_k\|_1 \|L_k\| \leq (d_{el}^2 - 1) \max_{k\neq 0} \|R_k\|_1 \|L_k\| \tag{3.10}$$

is a finite constant independent of O and of the eigenvalues of ϕ.
Considering $\phi_0^s = \phi \circ \ldots \circ \phi$ (s times), $\phi_0^s(O) = \sum_k \lambda_k^s \operatorname{Tr}(R_k^\dagger O) L_k$, the corresponding upper bound is

$$\left\| \tilde{\phi}_0^s(O) \right\| \leq c\, 2^{-\nu s} \left\| O \right\| . \tag{3.11}$$

Proof. Using the properties of a norm we have that

$$\left\| \tilde{\phi}(O) \right\| \leq \sum_{k \neq 0} |\lambda_k| \left| \operatorname{Tr}(R_k^\dagger O) \right| \left\| L_k \right\| \tag{3.12}$$

$$\leq 2^{-\nu} \sum_{k \neq 0} \left| \operatorname{Tr}(R_k^\dagger O) \right| \left\| L_k \right\| . \tag{3.13}$$

Moreover, we can use the matrix Hölder inequality to estimate

$$\left| \operatorname{Tr}(R_k^\dagger O) \right| = |\langle R_k, O \rangle| \leq \left\| R_k \right\|_1 \left\| O \right\| , \tag{3.14}$$

which proves the statement. □

3.2 Error correcting capabilities of MERA codes against erasures: the results of Kim & Kastoryano

Here we present the main results derived by Kim & Kastoryano (K&K) in [7]. The first one is that erased regions which are "sufficiently small" are approximately correctable, and the second one is a lower bound on the distance of the code, given a scale invariant RG-regular MERA (definition 3.1.5). We will present the main ideas of the proofs, whereas more details can be found in the aforementioned paper.

We consider erasure noise and so we will refer to the notion of local approximate quantum error correction (LAQEC), presented in section 1.4. We partition into ABC the system composed of n physical qubits at the bottom of the MERA, where A is the erased region, B is completely surrounding A, and C is the rest of the system. We also consider a purifying reference system R for code states. As a first step, in lemma 3.2.1 we assume A to be simply connected and C to be empty, which means that we prove the existence of a recovery map for erasure errors which is not really local but global. With theorem 3.2.2 we sharpen the statement by proving the existence of a local recovery map.

Otherwise stated, here we consider 1D MERA codes. However, the following results can be easily generalized to higher dimensions (only the distance is peculiar, as we shall see in the following). Moreover, we restrict to binary MERAs (figure 2.1.1), for which $n = k2^s$ and so $s = \log_2(n/k)$; in the case of m-nary MERAs we would simply have to change the basis of the logarithm: $s = \log_m(n/k)$. For a binary MERA, A collapses to elementary size under

renormalization after $r_A \gtrsim \log_2 |A|$ steps. Then there are $s - r_A$ renormalization steps over elementary blocks, where one can use the K&K assumptions (definiton 3.1.4) on ϕ to get:

Lemma 3.2.1. *Consider an RG-regular 1D MERA code (definition 3.1.5) and let ρ^{ABR} be a purified code state, where A is a simply connected region. Then for any $O_{AR} \in \mathcal{B}(\mathcal{H}_A \otimes \mathcal{H}_R)$ we have that*

$$\left|\mathrm{Tr}\Big((\rho^{AR} - \rho^A \otimes \rho^R) O_{AR}\Big)\right| \leq 2c\, d_{el}^2\, \|O_{AR}\|\, 2^{-\nu(s-\log_2|A|)}. \tag{3.15}$$

Sketch of the proof. Theorem 1.4.1 states that we can find an approximate recovery map if and only if there is approximate decoupling between A and CR (only R in this case), i.e. if there are almost no correlations between A and CR. Then the strategy is to prove bounds on such correlations. One can see that they depend on the eigenvalues of ϕ, except for the one corresponding to the identity. If all the eigenvalues are smaller than 1 in modulus, as we are supposing, it follows that the strenght of such correlations decays exponentially with the number of iterations of ϕ (the complete proof can be found in [7]). □

Remark 3.2.1. We can actually relax the condition on A and require A to be just connected and not simply connected. In the former case, we can just purposely erase the "holes" in A in order to make it simply connected.

Theorem 3.2.1 (Global correctability for simply connected regions)**.** *Consider an RG-regular 1D MERA code and let A be a simply connected region. Then there exists a recovery map $\mathcal{R}\colon \mathcal{B}(\mathcal{H}_B) \to \mathcal{B}(\mathcal{H}_{AB})$ such that*

$$\sup_{\rho^{ABR}} \mathfrak{B}(\mathcal{R}_B^{AB}(\rho^{BR}), \rho^{ABR}) \leq c'\, 2^{-\nu(s-\log_2|A|)/4} \tag{3.16}$$

where $c' = \frac{1}{3} c^{1/4} d_{el}^{1/2}$, $\mathfrak{B}$ is the Bures distance (definition 1.3.1) and the supremum is taken over purified code states ρ^{ABR}.

Proof. First recall from lemma B.0.1 that $\|\rho - \sigma\|_1 \geq 2\,(1 - \sqrt{F(\rho,\sigma)}) = 2\mathfrak{B}(\rho,\sigma)^2$ and recall that [10] the trace norm of an operator A can be characterized as $\|A\|_1 = \sup_{\|B\|\leq 1} |\mathrm{Tr}(AB)|$. Then from (3.15) we get that

$$\left\|\rho^{AR} - \rho^A \otimes \rho^R\right\|_1 \leq 2c\, d_{el}^2 2^{-\nu(s-\log_2|A|)}, \tag{3.17}$$

whereas using theorem 1.4.1 and equation (1.50) we can write:

$$\sup_{\rho^{ABR}} \left\|\rho^{AR} - \rho^A \otimes \rho^R\right\|_1 \geq 2 \sup_{\rho^{ABR}} \mathfrak{B}(\rho^{AR}, \rho^A \otimes \rho^R)^2 \tag{3.18}$$

$$\geq 2\left(3 \min_{\omega^A} \sup_{\rho ABR} \mathfrak{B}(\omega^A \otimes \rho^R, \rho^{AR})\right)^4 \tag{3.19}$$

$$= 2\left(3 \inf_{\mathcal{R}_B^{AB}} \sup_{\rho ABR} \mathfrak{B}(\mathcal{R}_B^{AB}(\rho^{BR}), \rho^{ABR})\right)^4. \tag{3.20}$$

Combining the two we get that:

$$\inf_{\mathcal{R}_B^{AB}} \sup_{\rho ABR} \mathfrak{B}(\mathcal{R}_B^{AB}(\rho^{BR}), \rho^{ABR}) \leq \frac{1}{3} c^{1/4} d_{el}^{1/2}\, 2^{-\nu(s-\log_2|A|)/4}. \tag{3.21}$$

□

By similar ways to those that one can use to derive lemma 3.2.1, one can prove:

Theorem 3.2.2 (Local correctability for simply connected regions [7])**.** *Consider an RG-regular 1D MERA code, with n physical qubits at the bottom of the network and k logical qubits at the top. Partition the physical qubits into regions A, B and C, where A is a simply connected region, B is a region comprising all the physical qubits which are at a geometric distance at most x away from A, and C is the rest of the physical system. Let R be a purifying reference system for code states. Moreover, assume that $|AB| < n/k$. Then there exist a recovery map $\mathcal{R}\colon \mathcal{B}(\mathcal{H}_B) \to \mathcal{B}(\mathcal{H}_{AB})$ such that*

$$\sup_{\rho^{ABCR}} \mathfrak{B}(\mathcal{R}_B^{AB}(\rho^{BCR}), \rho^{ABCR}) \leq c'' \left(\frac{|A|}{x}\right)^{\nu/4}. \tag{3.22}$$

where the supremum is taken over purified code states ρ^{ABCR}.

This is the formal statement regarding the first main result of K&K. The difference with theorem 3.2.1 is that here B is a region surrounding A but it does not necessarily have to include all the rest of the physical system. The reason why we assume $|AB| < n/k$ and consequently $|A| < n/3k$ is that to use the K&K assumptions we need region A to collapse to elementary size at least one level before the top (see the explicit proof in [7]).

It is clear that in the limit $\nu \to \infty$, i.e. if the idealized K&K conditions hold (definition 3.1.6), we can take x as small as $|A| + 1$ and still get *exact* correctability.

Corollary 3.2.1. *In the limit $\nu \to \infty$, any region A of size $|A| < n/3k$ is exactly correctable.*

Bounds on the distance

The extra tool that we need to prove lower bounds on the distance of a MERA code is the union lemma (see section 1.4). It states that if two independently correctable regions are far enough from each other, then they can be jointly corrected when both regions are erased (see lemma 1.4.1 for the precise statement). Such a tool will allow us to extend correctability from connected regions to disconnected ones, allowing us to correct erasures of regions whose size is bigger than the biggest size we may tolerate for a connected one, and allowing us to correct not just clusters of erasures, but also the case in which they are

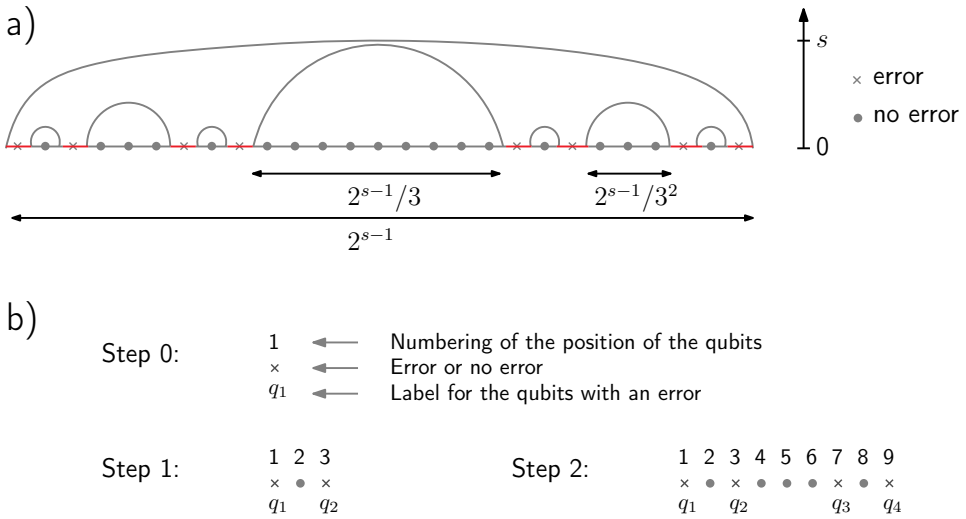

Figure 3.2.1: a) The distribution of errors involving the smallest number of sites for which we cannot prove correctability, using corollary 3.2.1 and/or the union lemma (lemma 1.4.1), takes a Cantor-like shape. b) Conventions used in the proof of theorem 3.2.3.

spread in different locations. The latter case is the one we will really need to prove lower bounds on the distance.

Intuitively, the smallest distribution of erasures for which we cannot prove correctability has to involve not clusters of erasures but single-qubit erasures spread with the lowest possible "density" in a region of size $n/3k$, however not as low as to allow us to use the union lemma in any place.

Theorem 3.2.3 (Lower bound on the distance of 1D MERA codes)**.** *Consider a 1D MERA code. If it satisfies the idealized K&K assumptions (definition 3.1.6), then the distance d of the code is lower bounded by:*

$$d \geq \frac{1}{2}\left(\frac{n}{k}\right)^{\alpha}, \quad \alpha = \log_3 2 \approx 0.63. \tag{3.23}$$

Proof. We will say that a (possibly disconnected) erased region is non-provably-correctable (*n-p-c*) if we cannot prove its correctability using corollary 3.2.1 and/or the union lemma (lemma 1.4.1).
In order to prove the statement we want to find the smallest n-p-c region. Then the distance is larger or equal to its size. We claim that such a region takes a Cantor-like shape, as it is shown in figure 3.2.1. The proof will be constructive, with reference to figure 3.2.1 for the notation.
Preliminary observation: consider an n-p-c cluster of erasures A (by cluster we mean a set of contiguous erasures). If we consider the case in which we subdivide this region into 3 parts, $A = A_L A_M A_R$, and remove the middle part A_M, i.e. we consider erased only the right and left parts A_L and A_R, we

can see that this second (disconnected) erased region $A' = A_L A_R$ is still n-p-c because there are not enough error free qubits around A_L or A_R to use the union lemma, whereas A' as a whole is n-p-c because A was n-p-c. Hence, given a n-p-c cluster A, we have found another one A' with $|A'| < |A|$. This implies that the minimal n-p-c region cannot contain any cluster of erasures, but only single-qubit erasures surrounded by at the least one non-erased qubit both on its left and on its right, otherwise it would not be minimal (two consecutive erased qubits may be ok, but in this way the "density" of erasures would increase and it would not correspond to the minimal n-p-c region).
Now we want to construct the minimal n-p-c region by adding erasures one by one till they are spread on a region of size (at least) $n/3k$ and in such a way that the union lemma cannot be used in any place.
Step 0: we start by adding a qubit q_1 that we consider to be the leftmost erasure (position 1) of the distribution we are going to construct (see figure 3.2.1).
Step 1: by the previous considerations, in order to achieve our goal the only place where we can put erasure q_2 is position 3, whereas position 2 is error-free.
Step 2: it is tempting to put erasure q_3 at position 5, however we can see that non-correctability of q_2 is already ensured by q_1, hence we need to ensure non-correctability for the whole region $\{1, 2, 3\}$, which requires us to put an erasure q_3 at position 7. But this is not enough because q_3 "alone" like this would be correctable, so we need to put also q_4 at position 9.
Step t: we need to add erasures $q_{2^{t-1}+1}, \dots, q_{2^t}$, where $q_{2^{t-1}+1}$ is at position $2 \cdot 3^{t-1} + 1$ and q_{2^t} at position 3^t and the other ones reproduce the pattern of erasures from position 1 to 3^{t-1}.
We iterate till $t = T$ such that $3^T \geq n/3k$, i.e $T \geq \log_3(n/k) - 1$. The total number of erasures that we need to add to be unable to prove correctability is 2^T. It follows that

$$d \geq 2^T \geq \frac{2^{\log_3(n/k)}}{2} = \frac{1}{2}\left(\frac{n}{k}\right)^{\log_3 2}. \tag{3.24}$$

□

Corollary 3.2.2. *Consider a MERA code describing a physical system in D spatial dimensions, defined on a hyper-cubic lattice with side of length $L = (n/k)^{1/D}$. Suppose that the MERA is isotropic in the different directions (there is no preferred one). Then, if it satisfies the idealized K&K assumptions, the distance is lower-bounded by*

$$d \geq \frac{1}{2}\left(\frac{n}{k}\right)^{0.63/D}. \tag{3.25}$$

Proof. If we deal with the different dimensions in the same way, this means that the worst-case distribution of erasures for which we cannot prove correctability is the same as in the 1D case, i.e. we just need erasures lying on a line with the pattern of figure 3.2.1, because then they do not renormalise

over an elementary block before reaching the top of the network (this is something which is needed to prove correctability in the proofs of theorem 3.2.2 and corollary 3.2.1, see [7]). □

Remark 3.2.2. In the K&K paper [7] the bound in corollary 3.2.2 appears as an upper bound but we believe that it is a typo.

3.3 Discussion

We want to analyse the lower bound in corollary 3.2.2 and its consequences for the codes satisfying the given hypothesis. In particular we want to compare it with the bounds that have been derived in a number of papers [18, 27, 28]. In [18] it was derived by Bravyi, Poulin and Terhal that local stabilizer codes (see definition 1.5.3) in dimension $D \geq 2$ are subject to the trade-off

$$d \leq c \left(\frac{n}{k}\right)^{(D-1)/2} \tag{3.26}$$

between the distance d, the number of physical qubits n and logical qubits k, where c is some numerical constant independent of the other parameters. Instead in [27] Bravyi and Terhal derived that for local stabilizer codes in any $D \geq 1$ it holds

$$d \leq c\, n^{\frac{D-1}{D}} \tag{3.27}$$

which means in particular that in $D = 1$ the distance is upper bounded by a constant.

Comparing these no-go results with the result of Kim and Kastoryano we see that they are incompatible in $D = 1$, implying that the codes satisfying the assumptions of corollary 3.2.2 must be non-local. We note that this is something allowed by the MERA structure, because the width of the future causal cone (definition 2.1.4) increases exponentially, hence it is possible that an operator supported on a few sites at some intermediate level is supported on a growing number of sites the more we push it to the bottom of the network. On the contrary, there is no contradiction in dimension $D = 2$ or higher, therefore in that case it may be possible to build a local code satisfying the bound (3.25). Indeed, we shall see that the toric code is an example of a local stabilizer code in $D = 2$ with a MERA representation, satisfying some *refined* version of the K&K assumptions (definition 3.4.1), whose distance is known to be $\sqrt{n}$, which is even a better asymptotic scaling than the one guaranteed by (3.25).

3.4 General lessons from the Ising chain and the toric code

Here we are going to explain why the Ising chain is a very bad quantum error correcting code whereas the toric code is a pretty good one, in terms of the eigenvectors of the renormalization map.

Ising chain. Consider first the Ising chain (section 2.3, figures 2.3.1 and 3.1.1b). In this section we are concerned with the renormalization of operators ("errors") acting on the physical level. Specifically, we will consider Pauli errors. Call $\mathcal{C} = \text{span}\{|0\rangle^{\otimes n}, |1\rangle^{\otimes n}\}$ the code subspace. Recalling the action (2.22) of conjugation by CNOT, we notice the following, denoting for example by XY an operator where X acts on the left leg of an isometry, or its unitary extension U, and where Y acts on the right one.

1. XI is renormalized by U to XX, which corresponds to the fact that the ancilla $|0\rangle$ *switches* to $X|0\rangle = |1\rangle$. This implies that XI maps the code subspace to an orthogonal subspace, because code states have all the ancillas set to $|0\rangle$ and, when one projects the current state onto any of them, at some point one gets $\langle 0|1\rangle = 0$ (all intermediate tensors simplify similarly to figure 2.1.4);

2. IX is renormalized to IX, so that also in this case the code subspace is mapped to an orthogonal subspace;

3. XX is renormalized to XI, which means that we do not know just from one level if $\mathcal{C}$ is mapped to itself (and so XX is a logical operator) or to something orthogonal to it (CNOT maps Pauli operators to Pauli operators, so either an ancilla switches or not, which means that either $\mathcal{C}$ is mapped to an orthogonal subspace or to itself). However, one level higher, we have a single qubit X acting on the left leg of a CNOT, which means that we are in the same case as item 1, i.e. there is an ancilla at the next level which is switching from $|0\rangle$ to $|1\rangle$.
   We can ask what an operator is which does not make any ancilla switch and which is not (proportional to) the identity on the logical qubit at the top, i.e. we can ask what is an example of a logical operator. From the above observations we can infer that $X^{\otimes n}$ is such an example, because at each level it is renormalized to a product of X's acting on every qubit of the new lattice, whereas no ancilla switches to $|1\rangle$. Hence, we get an X acting on the free index at the top.

4. ZI is renormalized to ZI;

5. IZ is renormalized to ZZ, which is equivalent to ZI on the code subspace because $Z|0\rangle = I|0\rangle$.
   Therefore, any single qubit Z is still a single qubit Z one level higher and so on, till Z reaches the top with no ancilla switching, hence it is another example of a logical operator. In particular, it has weight 1 and therefore the distance of $\mathcal{C}$ is 1 independently of the system size, which makes the Ising chain a bad code (unless there is some symmetry-protection mechanism assuring that there are only X errors).
   According to our definition of an elementary block (definition 3.1.2), in this case there is only one kind of elementary block and it is made up of two qubits as in figure 3.1.1b. It is easy to see that IZ, ZI, ZZ are all

Table 3.4.1: Regarding the toric code, in this table for each qubit in the support of a disentangler we list to which operator a single qubit X or Z is renormalized, plus the ancillas which are switching from $|0\rangle$ to $|1\rangle$ or from $|+\rangle$ to $|-\rangle$ (for the qubit numbering see figure 2.3.5). Recall that $X|0\rangle = |1\rangle$ and $Z|+\rangle = |-\rangle$.

Qubit number	X	Switching	Z	Switching
1	X_1	$\|1\rangle_9 \|1\rangle_{16}$	Z_1	$\|-\rangle_5$
2	X_2	$\|1\rangle_{31}$	Z_2	$\|-\rangle_6$
3	X_3		Z_3	$\|-\rangle_7$
4	X_4	$\|1\rangle_{34}$	Z_4	$\|-\rangle_8$
5	X_1	$\|1\rangle_{10} \|1\rangle_{15}$		$\|-\rangle_5$
6	X_2	$\|1\rangle_{10} \|1\rangle_{12}$		$\|-\rangle_6$
7	X_3	$\|1\rangle_{12} \|1\rangle_{13}$		$\|-\rangle_7$
8	X_4	$\|1\rangle_{13} \|1\rangle_{15}$		$\|-\rangle_8$
9		$\|1\rangle_9 \|1\rangle_{10}$	Z_1, Z_{19}	$\|-\rangle_5 \|-\rangle_{29} \|-\rangle_{40}$
10		$\|1\rangle_{10}$	Z_1, Z_{19}	$\|-\rangle_6 \|-\rangle_{29} \|-\rangle_{40}$
11		$\|1\rangle_{12} \|1\rangle_{31}$		$\|-\rangle_{11}$
12		$\|1\rangle_{12}$		$\|-\rangle_6 \|-\rangle_7 \|-\rangle_{11}$
13		$\|1\rangle_{13}$		$\|-\rangle_7 \|-\rangle_8 \|-\rangle_{14}$
14		$\|1\rangle_{13} \|1\rangle_{34}$		$\|-\rangle_{14}$
15		$\|1\rangle_{15}$	Z_1, Z_{17}	$\|-\rangle_8 \|-\rangle_{36} \|-\rangle_{62}$
16		$\|1\rangle_{15} \|1\rangle_{16}$	Z_1, Z_{17}	$\|-\rangle_5 \|-\rangle_{36} \|-\rangle_{62}$

eigenvectors of the transfer operator with eigenvalue 1. Therefore, the Ising chain does not satisfy the assumptions of corollary 3.2.2, as we can expect, otherwise it should have been a much better code.

Toric code. Something similar happens in the case of the toric code (section 2.3). Using table 3.4.1 we can derive to what any Pauli operator is renormalized, because we can renormalise independently each tensor product factor (the labelling of the position of the qubits refers to figure 2.3.5). First we can notice that any single-qubit Pauli operator, except X_3, causes some ancillas to switch and hence it is mapping the code subspace to an orthogonal one. Regarding X_3, one level higher it sits in the same position relative to the upper disentangler as the position of qubit 7 relative to the disentangler shown in figure 2.3.5. Hence, it causes an ancilla to switch at the next level and therefore it is also mapping the code subspace to an orthogonal one. However, there are multi-qubit errors for which the ancillas switch an even number of times, similarly to the XX case for the Ising chain, so that we need to look at the next levels to understand if they are logical operators or not.

It is interesting to consider the error $X_5X_6X_7X_8$. Notice that the 16 qubits in the support of a disentangler form an elementary block (definition 3.1.2) because they are connected to the 16 full-black qubits in figure 2.3.5. Denoting by ϕ the transfer operator (definition 3.1.3) associated to this elementary block, we get that

$$\phi(X_5X_6X_7X_8) = X_1X_2X_3X_4. \tag{3.28}$$

The numbering on the right does not really matter, but it is important that the position of qubits 1,2,3,4 at the higher level, relative to the new disentangler, is the same as the relative position of qubits 5,6,7,8 w.r.t. the old disentangler (see figures 2.3.3d and 2.3.5). Therefore, according to (3.28), $X_5X_6X_7X_8$ is an eigenvector with eigenvalue 1 and it follows from the structure of the network that it is propagated up to one level before the top of the network. It may seem then that the toric code should be no better then the Ising chain, however, at this point the top tensor of the toric code is important: this eigenvector is indeed renormalised to the identity. Equivalently, notice that it is a stabilizer of the toric code and hence by construction it acts as the identity on the code subspace.

From the previous observations we can learn the following:

1. for the toric code it is not difficult to realize that the ancillas which are switching are in one-to-one correspondence with error syndromes (definition 1.5.4). There are some ancillas which are mapped to one bit of the syndrome (values ± 1) and others which are mapped to the product of two bits (and there is always one ancilla mapped to one of the two factors). This is directly related to what we noticed in table 2.3.1, where we saw that the stabilizers of the ancillas are mapped to star or plaquette operators or to the product of two of them.
   We can realize that such a correspondence between ancillas and error syndromes depends only on the fact that the CNOT, building the MERA representation of the toric code, maps Pauli operators to Pauli operators. Therefore, the correspondence holds for any MERA whose disentanglers and isometries are Clifford operators (as we are going to define them in definition 3.5.1);

2. it is possible that we need to renormalise an error across a few levels and not just one to understand what its action on the code subspace is, for example X_3 for the toric code;

3. in the case in which there are more than one type of elementary block, it is possible that an eigenvector of one transfer operator with an eigenvalue of modulus 1 is an eigenvector with eigenvalue 0 of another one. When this happens we say that there is *synergy* between transfer operators. If the network is arranged in such a way that one type of elementary block is always mapped to the other one, then such an eigenvector for which there is synergy is not so harmful because it does not propagate till the

top without changing the state of all the ancillas, i.e. it is not a logical operator as we may naively expect;

4. if an eigenvector with an eigenvalue of modulus 1 is an eigenvector with eigenvalue 0 of the top tensor, we say that there is *synergy* between the transfer operator and the top tensor. For example, we saw that this happens for some stabilizers of the toric code;

5. eigenvectors with eigenvalue 1 are logical operators unless there is some kind of synergy.

We can infer then that the assumptions of theorem 3.2.3 can be relaxed a little bit while still implying the same lower bound on the distance, including the case in which there are more than one type of elementary block.

Definition 3.4.1 (Refined K&K assumptions)**.** Consider a MERA code and suppose that its MERA representation has M types of elementary blocks with corresponding transfer operators $\{\phi_i\}_{i=1}^{M}$. Suppose that

1. $\forall i = 1 \dots M$, ϕ_i is diagonalizable, with eigenvalues $\{\lambda_k^{(i)}\}_{k\geq 0}$;

2. $\forall i = 1 \dots M$, ϕ_i has $\bar{k}_i \geq 1$ eigenvalues such that $\left|\lambda_k^{(i)}\right| = 1$ for $k = 0 \dots \bar{k}_i - 1$ and $\lambda_k^{(i)} = 0$ for $k \geq \bar{k}_i$. Let $\lambda_0^{(i)} = 1$ be the eigenvalue whose corresponding eigenvector is the identity (which is always present, see appendix A.1);

3. there is synergy, as defined above, between different transfer operators and/or the top tensor of the given MERA, such that $\forall i = 1 \dots M$ all the eigenvectors corresponding to the eigenvalues $\{\lambda_k^{(i)}\}_{k=1}^{\bar{k}_i - 1}$ are *not* logical operators.

We call these three conditions the *refined* K&K assumptions.

It is clear that theorem 3.2.3 holds if we consider the refined K&K assumptions instead of the idealized ones.

3.5 A 1D MERA code with high threshold probability

In this section we present the results of a few numerical simulations that we have performed. Our aim is to construct interesting families of quantum error correcting codes based on MERA.

Structure of the network and general framework

We consider a 1D binary MERA with periodic boundary conditions, as in figure 3.5.1, and we fix the number of logical qubits to 1 (the number of qubits at

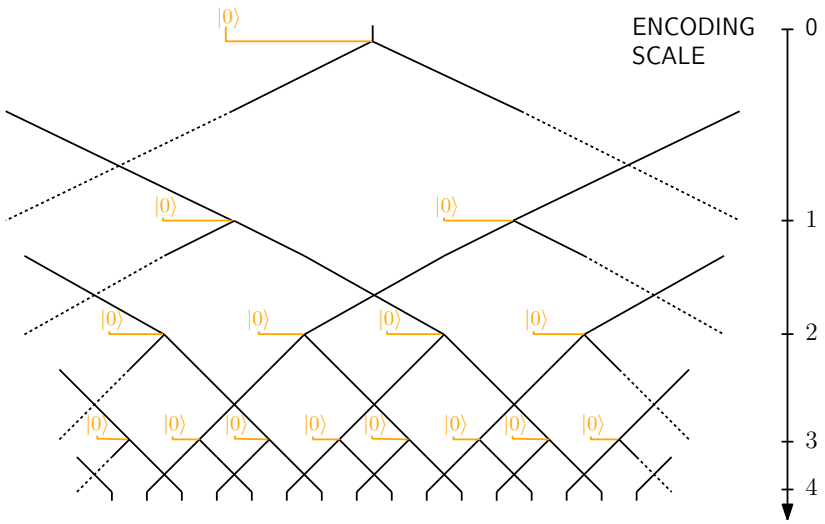

Figure 3.5.1: The network we consider throughout section 3.5. We assume periodic boundary conditions, so the dotted line of a disentangler on the right corresponds to the solid line of the corresponding disentangler on the left, and vice-versa. We use an encoding scale starting from 0 at the logical level. The number of qubits at level j is 2^j. Usually we will refer to the unitary extension of the MERA (definition 2.1.1) because the ancillas are in correspondence with error syndromes in the case of Clifford unitaries (stabilizer codes) and Pauli noise.

the top). In particular, we consider a scale-invariant network (definition 3.1.1), i.e. we pick all the isometries at all levels to be equal to some isometry $\tilde{V}$ and all the disentanglers equal to some unitary U. More specifically, we are concerned with the unitary extension of the MERA (definition 2.1.1), therefore we will refer in general to the unitary extension V of the isometry $\tilde{V}$, such that $\tilde{V}(\cdot) = V(|0\rangle \otimes (\cdot))$.

We want to analyse the code properties as a function of the number of levels that we add at the bottom. For that it will be convenient to use an *encoding* scale (see figure 3.5.1) oriented in the opposite direction w.r.t. the renormalization scale that we have considered so far.

We will focus on two cases:

1. General unitaries: we pick two random unitaries U and V and construct the network from them. We may choose them according to the Haar measure but it does not really matter how we pick them. In terms of their matrix representation, the isometry $\tilde{V}$ is equal to the first two columns of V in the standard computational basis (Z-basis for each qubit).

2. Clifford unitaries.

Definition 3.5.1 (Clifford group)**.** Consider the Pauli group $\mathcal{G}_n$ over n qubits (definition 1.5.1). An n-qubit unitary C is called a *Clifford unitary* or *Clifford*

group operator if

$$CPC^\dagger \in \mathcal{G}_n \qquad \forall P \in \mathcal{G}_n. \tag{3.29}$$

The set of all Clifford unitaries over n qubits is a group called the Clifford group over n qubits.

When we use Clifford unitaries we are defining a stabilizer code with the MERA (we call it a *stabilizer MERA code*): in the so-called Heisenberg representation of quantum circuits [26], the trivial stabilizers (single-qubit Z) over the ancillas are mapped on the physical level to other Pauli operators, by definition of Clifford unitary. Moreover, those Pauli operators are precisely the stabilizer generators whose $+1$ eigenspace is the code subspace defined by the MERA, as we have explained in section 3.4.

At level 1, we start with just one stabilizer over two qubits. At level 2, that stabilizer is mapped to a stabilizer over 4 qubits; besides that, two more stabilizers are introduced, coming from the two new ancillas. Continuing like this, the more levels of encoding we add, the more stabilizer generators are introduced, while the pre-existing ones can become more and more non-local because they are in general supported on a growing number of qubits.

It is convenient to represent Pauli operators and Clifford unitaries using binary vectors and binary matrices (vectors and matrices with entries equal to 0 or 1). Modulo phases, a Pauli operator $P \in \mathcal{G}_n$ can be written as a product of $2n$ independent generators, that conventionally are taken to be $\{X_1, X_2, \ldots, X_n, Z_1, Z_2, \ldots, Z_n\}$, where for example $X_1 = X \otimes I \otimes \cdots \otimes I$. Hence any $[P] \in \tilde{\mathcal{G}}_n \coloneqq \mathcal{G}_n/\{\pm 1, \pm i\}$ can be written as $[P] = \prod_{i=1}^n [X_i^{\alpha_i}][Z_i^{\beta_i}]$ where $\alpha_i, \beta_i \in \{0, 1\}$ and where we use square brackets for equivalence classes in $\tilde{\mathcal{G}}_n$. Let $\mathbb{F}_2$ be the field given by the set $\{0, 1\}$ together with modulo-2 addition and multiplication. The set of all binary vectors of length N forms a vector space over $\mathbb{F}_2$ that we call the binary vector space $\mathbb{F}_2^N$.

Conventionally, the group isomorphism $\sigma\colon \tilde{\mathcal{G}}_n \to \mathbb{F}_2^{2n}$ is defined as $[P] \mapsto (\alpha_1, \alpha_2, \ldots, \alpha_n, \beta_1, \beta_2, \ldots, \beta_n)$, which is denoted by $(\boldsymbol{\alpha}|\boldsymbol{\beta})$. However, we choose to use

$$\begin{aligned} \sigma'\colon \quad \tilde{\mathcal{G}}_n &\to \mathbb{F}_2^{2n} \\ [P] &\mapsto (\alpha_1, \beta_1, \ldots, \alpha_n, \beta_n) \end{aligned} \tag{3.30}$$

and we sometimes denote it by $[P] \mapsto (\alpha_1\beta_1|\ldots|\alpha_n\beta_n)$. For example, in our convention $[XIZY]$ over 4 qubits is mapped to $(1, 0, 0, 0, 0, 1, 1, 1)$ or equivalently $(10|00|01|11)$.

Similarly, Clifford unitaries are represented as binary (symplectic) matrices $C\colon \mathbb{F}_2^{2n} \to \mathbb{F}_2^{2n}$, where the condition (3.29) translates into the requirement that $CJC^T = J$, where for example in the case $n = 2$ in our convention

$$J = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{3.31}$$

whereas the standard one is obtained by swapping the two middle rows and middle columns.

We will consider Pauli noise because stabilizer codes are well-suited in general against this error model. In our framework, the advantage is that Pauli operators acting on the physical level are renormalized to other Pauli operators at higher levels. If the tensor product factor acting on the ancilla $|0\rangle$ is either I or Z, nothing happens, but if it is X or Y then the ancilla switches to $|1\rangle$ (modulo phases, which are irrelevant here because they are an overall constant which is the same for *all* code states).

As we noticed in section 3.4, the ancillas which are switching by the action of some Pauli operator P are in correspondence with the error syndrome associated to P, because if P is renormalized as I or Z over an ancilla, then this means that P commutes with the corresponding stabilizer, otherwise P anticommutes with it (in section 3.4 we analysed the toric code as an example, but we can now notice that it is a stabilizer MERA code, being constructed just with CNOTs, and that the argument here works in general in the same way).

Diagonalizability of the transfer operators

From the discussion of section 3.1 (see in particular figures 3.1.1 and 3.1.2) we know that in the case of a 1D binary MERA there are two types of elementary blocks, corresponding to the "left" transfer operator ϕ_L and to the "right" one ϕ_R. We would like to compare the results of Kim & Kastoryano (section 3.2) with some numerical simulations. Therefore, we are interested in finding codes satisfying the idealized K&K assumptions (section 3.1, definition 3.1.6).

First we checked whether for binary MERAs it is possible to pick two Clifford unitaries U and V such that ϕ_R and ϕ_L are diagonalizable and have only one eigenvector of eigenvalue 1, whereas all the others are 0. It turns out that this is not possible, as proved via exhaustive search over all the 720^2 possibilities of picking two Clifford unitaries (for 2 qubits there are 720 Clifford group elements modulo phases). It may be possible picking qubits or other types of MERAs (not a binary one), but we do not know that because the cardinality of the Clifford group grows even more than exponentially fast, so that exhaustive search becomes quite inefficient.

However, it is possible that for example ϕ_L^2 satisfies the idealized K&K assumptions. We can explain such a behavior considering the matrix representation of ϕ in the Pauli basis (we do not write a subscript when we do not need to distinguish between ϕ_R and ϕ_L). Here ϕ maps operators supported on 3 qubits to operators supported onto 3, so by "Pauli basis" we mean the set of Pauli operators on 3 qubits, normalized with a factor $1/\sqrt{8}$: in this way they form an orthonormal basis w.r.t. the Hilbert-Schmidt inner product, i.e. $\langle \tilde{P}_i, \tilde{P}_j \rangle = \delta_{ij}$ where $\tilde{P}_i = P_i/\sqrt{8}$ and P_i is a Pauli operator (there are 64 of them for 3 qubits). It is easy to realize that ϕ maps Pauli operators to Pauli

operators or to the null operator $\mathbf{0}$, being constructed from Clifford operators (but it is not a Clifford group element because of the second possibility).

As we explain in appendix A.1, the matrix representation of ϕ has entries in the Pauli basis given by

$$\hat{\phi}_{ij} = \mathrm{Tr}(\tilde{P}_i \, \phi(\tilde{P}_j)). \tag{3.32}$$

Because Pauli operators, except for the identity, always have trace equal to 0, it follows that

$$\left|\hat{\phi}_{ij}\right| = \begin{cases} 1 & \text{if} \quad \phi(P_j) \propto P_i \\ 0 & \text{else.} \end{cases} \tag{3.33}$$

where the proportionality constant can be ± 1 or $\pm i$ (using binary vectors phases are neglected by construction, here they cannot be discarded from the description). We notice that in a column there is always at most one entry with modulus 1 (because a Pauli operator is mapped at most to another one, or to $\mathbf{0}$), whereas there can be more in a row (because ϕ does not have to be injective). If the i-th element on the diagonal has modulus 1, then the corresponding Pauli operator P_i is an eigenvector with eigenvalue equal to that diagonal element.

In principle there can also be eigenvectors with eigenvalues different from 0, ± 1 and $\pm i$. Indeed, in all the many cases we analysed, we found that for Clifford operators all the roots of the characteristic polynomial of $\hat{\phi}$ always belong to the set

$$\{0, \pm 1, \pm i, e^{\pm i \frac{\pi}{3}}, e^{\pm i \frac{2\pi}{3}}\}, \tag{3.34}$$

but we do not have an analytical explanation about why these are the only possibilities. Anyway, the Jordan blocks often have a small dimension. In particular, it is not very rare to find examples where the only eigenvalue of modulus 1 is the one corresponding to the identity (which is always an eigenvalue and it is equal to 1, as explained in appendix A.1), where all the other roots of the characteristic polynomial are zero and the Jordan blocks of both ϕ_R and ϕ_L have dimension at most 2, so that ϕ_R^2 and ϕ_L^2 satisfy the idealized K&K assumptions (definition 3.1.6); moreover, for some of such codes also $\phi_L \circ \phi_R$ and $\phi_R \circ \phi_L$ satisfy the same assumptions (see the following for an example), which means that the transfer operators across *two* levels can satisfy the idealized K&K assumptions and therefore the hypothesis of theorem 3.2.3. This shows the reason why we called them "idealized": it seems that they cannot be satisfied as originally presented in [7] (at least surely not in the case of a binary MERA as studied here), but there are possibilities of getting something very similar to them, so that they are still interesting conditions to study.

In terms of the result of theorem 3.2.3, for those codes this implies that the lower bound on the distance is modified just in the pre-factor, which is lowered by a factor of 3 because we need operators to renormalise to an elementary block not just one but two levels before the top of the network. The asymptotic scaling should be unchanged, however we observe that in the case of just a few

levels of encoding there are some finite-size and discretization effects which are ignored in theorem 3.2.3:

- up to level 3 (numbering of levels as in figure 3.5.1) there are not enough qubits to talk about elementary blocks and so from level 2 to level 0 the renormalization does not involve ϕ and this may have an effect similar to a non-trivial top tensor;

- the asymptotic scaling is also affected by the fact that the number of qubits on which an operator is supported does not really halve at each renormalization step, but it involves a few more qubits.

However, for the following it is not so important what the precise Kim & Kastoryano lower bound on the distance should be. For different examples of MERA codes, we numerically computed the distance for a few levels of encoding and then we performed some Monte Carlo simulations to check whether there is a threshold probability (section 1.5), i.e. an *effective* linear distance (definition 1.5.9).

Before showing the results of our simulations, we briefly summarize what we got for general unitaries in terms of the eigenvalues of the transfer operators. First, ϕ is generally not diagonalizable. Second, we see from the simulations that the eigenvalues seem to take any value in the unit disk in the complex plane, with no restrictions as it is the case for Clifford unitaries. Hence it looks difficult to find a non-Clifford unitary satisfying the idealized K&K assumptions. Therefore, in the following we restrict only to Clifford operators, i.e. we analyse only MERA stabilizer codes.

Code distance

In the case of Clifford operators, we simulated many codes (many choices of Clifford disentanglers and isometries) and found lower bounds on their distance. For a fixed code, our computer program works as follows: it checks first if there exists a weight 1 Pauli operator P which is a logical operator; if there is one, then the distance is equal to 1 and the program stops, otherwise we repeat the process for weight 2 Pauli operators and so on. To check whether a Pauli operator P is logical we verify first whether it commutes with all the stabilizers and, second, whether it is contained in the stabilizer group (it is a linear algebra problem that can be solved via Gaussian elimination for binary vectors); if P satisfies both conditions, then it is a logical operator. If we want to check that the distance is lower-bounded by $t+1$, then we stop the program after having checked all Pauli operators with weight smaller or equal to t, unless the program has already stopped by itself because the distance was found to be smaller than $t+1$.

In general (not just for MERA codes) the problem of finding the distance of a stabilizer code has a complexity which is exponential in n. Instead, as

Table 3.5.1: Some examples of stabilizer codes with their distance after 2, 3 or 4 levels of encoding (respectively 4, 8 and 16 physical qubits). We list also the eigenvalues of the transfer operators, where the subscript in parenthesis is the algebraic multiplicity. For codes where there is only one eigenvector with eigenvalue 1, we also write the maximum dimension of a Jordan block.

	Distance			Eigenvalues		
Ex.	Level 2	Level 3	Level 4	ϕ_L	ϕ_R	Jordan
1	1	1	1	1	$1_{(2)}, -1_{(2)}$	
2	1	1	1	$1_{(4)}, e^{\pm 2\pi i/3}_{(2)}$	$1_{(4)}, e^{\pm 2\pi i/3}_{(2)}$	
3	2	2	2	$1_{(2)}$	1	
4	1	1	3	$1_{(2)}$	1	
5	2	2	3	1	1	3
6	2	3	4	1	1	3
7	1	2	3	1	1	2

we do with the above algorithm, in order to decide if the distance is lower-bounded by $t+1$, one has to check all operators of weight smaller or equal to t, whose number is

$$\sum_{w=1}^{t} 3^w \binom{n}{w}, \tag{3.35}$$

which is of order $\mathcal{O}(3^t n^t)$. The problem has then "just" polynomial complexity, but the degree of the polynomial gets bigger the higher lower-bound one wants to check, times an exponential factor, in such a way that it is very difficult to prove interesting lower bounds for even just 5 levels of encoding (32 qubits).

In table 3.5.1 we have listed a few examples of codes with their corresponding distance, the non-zero eigenvalues of ϕ_R and ϕ_L and the maximum dimension of a Jordan block (it is interesting only the case in which the transfer operators have only one eigenvalue of modulus 1). We make the following observations:

- Examples 1, 2 and 3 have a distance which remains constant, which can be ascribed to the fact that there are eigenvectors with eigenvalue 1 which are propagated from the bottom to the top without causing any ancilla to switch, i.e. they are logical operators.

- Example 4 is peculiar, because from what we have just said we would not expect the distance to increase (we checked that it is 5 at level 5). Indeed, this is an example where the first levels act as a non-trivial top tensor: the point is that ϕ needs at least 6 qubits to be defined, as it clear from figure 3.1.2, so it is possible that its non-identity eigenvector with eigenvalue 1 makes some ancilla switch at level 1 or 0 (numbering

as in figure 3.5.1), so that it is not a logical operator. It is something similar to what happens for the toric code, as we noticed in section 3.4.

- Examples 5, 6 and 7 satisfy the idealized K&K assumptions across 3 renormalization steps (ex. 5, 6) or just 2 (ex. 7), so these are the codes that we expect to perform better.

Threshold probability

As we have discussed in sections 1.1 and 1.5, the distance is not the best indicator of how well a code will perform. In particular, there can be a threshold probability even if the distance is sub-linear in the system size (see section 1.5). Then, we want to check whether there exists a stabilizer MERA code with a non-zero threshold probability w.r.t. Pauli noise and some recovery map, that we have to construct.

Our computer program works as follows:

- we randomly generate a Pauli error P, where each tensor product factor is independently an X, Y or Z with probability $p/3$, for some p, or an I with probability $1 - p$.

- we measure the error syndrome associated to P, i.e. we check if P commutes or anticommutes with the stabilizers of the ancilla qubits (once we map such stabilizers to the physical level using the encoding map).

  Remark 3.5.1. Numerically we do that in one of the two following ways, depending on convenience: either we directly verify whether P commutes or anti-commutes with those stabilizers, or we renormalise P with the unitary extension of the MERA and check what the state of the ancillas is.
  Experimentally one has two options: 1) measure the stabilizer on the bottom level, which in general means measuring operators supported on a large number of qubits, far from each other in the general case 2) decode the encoded state applying the inverse of the unitary encoding map and then measure the ancillas. The second option has the advantage that one has to perform just a single-qubit measurement, but the disadvantage is that one has to perform also the decoding, as we have said, and that would make error correction impossible if the noise channel acts on the decoded state (there is no reason why that should not happen in an experiment). Hence one has to stick with method 1.

- We run a recovery algorithm (definition 1.5.7) chosen among one of those that we will present in the following, which takes the error syndrome and outputs its best guess about what is a correction $\tilde{P}$, chosen among the Pauli operators compatible with the given syndrome (definition 1.5.4) such that $\tilde{P}P$ acts as the identity on the code subspace, i.e. such that $\tilde{P}P$ belongs to the stabilizer group. If it does not, then it is a logical operator and we say that the recovery algorithm *failed* for the error P.

- We repeat the procedure for many random errors and estimate the logical error rate as the fraction of times the recovery algorithm failed. We repeat the whole procedure also for different probabilities p and various levels of encoding.

Before presenting some results, we want to formalize the problem of finding a "good" recovery algorithm. The idea is that we have a linear map and we are given a few bits (the syndrome) of the output caused by some input (an error); from those bits we have to guess an input (a correction), compatible with those bits, such that our guess lays in the same coset of the true input. We can phrase the problem in purely classical terms:

Definition 3.5.2 (The recovery game)**.** Consider a linear map $\mathcal{M}_n \colon \mathbb{F}_2^{2n} \to \mathbb{F}_2^{2n}$ and a subspace $S_n \subset \mathbb{F}_2^{2n}$ such that $\dim S_n \leq n-1$, where $\mathbb{F}_2^{2n}$ is the binary vector space over $\mathbb{F}_2$. Let P_n be a probability distribution over the 2^{2n} binary vectors in $\mathbb{F}_2^{2n}$. Suppose that Alice and Bob know what $S_n, \mathcal{M}_n$ and P_n are.
Alice chooses a vector $v_1 \in \mathbb{F}_2^{2n}$ according to P_n and reveals $\dim S_n$ bits of $v_2 = \mathcal{M}_n(v_1)$ to Bob. According to some strategy, Bob chooses (computes) a vector v_3 such that $\mathcal{M}_n(v_3)$ coincides with the revealed bits. If $v_3 \oplus v_1 \notin S_n$, we count one fail for Bob ($\oplus$ denote the bitwise sum modulo 2).
They repeat the procedure N times. Call $f_{n,N}$ the fraction of times Bob fails. The bits revealed by Alice are always at the same position for every trial. Bob wins the recovery game w.r.t. sequences of subspaces $\{S_n\}_{n\geq 2}$, maps $\{\mathcal{M}_n\}_{n\geq 2}$ and probability distributions $\{P_n\}_{n\geq 2}$ if he can find a strategy such that

$$\lim_n \lim_N f_{n,N} = 0. \tag{3.36}$$

Definition 3.5.3 (The efficient-recovery game)**.** We say that Bob wins the efficient-recovery game if he can find a winning strategy for the recovery game and if he can compute v_3 in polynomial time, using the same strategy.

In our case, the subspace S_n is the code subspace defined by a 1D binary MERA (as in figure 3.5.1) with $\log_2 n$ levels, the map $\mathcal{M}_n$ is the unitary extension of the renormalization map and the probability distribution P_n is the one corresponding to i.i.d. Pauli noise (throughout this section we implicitly use the isomorphism (3.30) to map Pauli operators to binary vectors). The vector v_1 is an error and the revealed bits are the error syndrome (for 1 logical qubits they are always $n-1$). We denote the syndrome by $\boldsymbol{s} \in \mathbb{F}_2^{n-1}$ to conform the notation to the one we used in definition 1.5.4. The strategy is a recovery algorithm (definition 1.5.7). Moreover, being able to win the recovery game is equivalent to the existence of a threshold probability.

As we discussed in section 1.5, there are two commonly used strategies:

- the maximum-likelihood recovery algorithm: Bob chooses the most likely vector v_3 compatible with the given syndrome $\boldsymbol{s}$, that is, he looks for the maximum of the probability distribution $P_n(v|v \in C_{\boldsymbol{s}})$, where we call $C_{\boldsymbol{s}}$

the set of vectors $v \in \mathbb{F}_2^{2n}$ corresponding to Pauli operators compatible with $\boldsymbol{s}$ (definition 1.5.4);

- the smallest-weight recovery algorithm: Bob chooses v_3 to be the smallest-weight vector in $C_{\boldsymbol{s}}$ (i.e. the corresponding Pauli operator is the smallest-weight Pauli operator compatible with the given syndrome). If the smallest-weight vector is not unique, then Bob chooses one according to some criterion or he can just pick a random one.

The problem is that both types of recovery algorithm take an exponential time in n to compute v_3, hence, if one is able to win the recovery game using one of them, then one is not able to win also the efficient-recovery game using the same one, but it may be possible using another one: depending on what $\mathcal{M}_n$ is, the most common way of tackling the problem is to find a recovery algorithm which does not look for the most likely or the smallest-weight vector, but which looks for a "very" likely or "sufficiently small" weight vector, where "very" or "sufficiently" have to be ultimately quantified by the fact that one is able to win the efficient-recovery game or not.

For stabilizer MERA codes we checked first whether we can win the recovery game with the smallest-weight recovery. We present the results of the simulations we have performed with the code listed as example 7 in table 3.5.1, which looked the most promising one because it satisfies the idealized Kim & Kastoryano assumptions across just 2 levels. As a reference, example 7 is constructed from the following Clifford element as disentangler:

$$I \otimes Z \mapsto Z \otimes I \tag{3.37}$$
$$Z \otimes I \mapsto Z \otimes X \tag{3.38}$$
$$I \otimes X \mapsto X \otimes Y \tag{3.39}$$
$$X \otimes I \mapsto Z \otimes Z, \tag{3.40}$$

where for example $I \otimes Z$ is mapped to $Z \otimes I$ under conjugation, and from the following one as isometry:

$$I \otimes Z \mapsto Z \otimes X \tag{3.41}$$
$$Z \otimes I \mapsto Y \otimes Y \tag{3.42}$$
$$I \otimes X \mapsto I \otimes Y \tag{3.43}$$
$$X \otimes I \mapsto X \otimes Y. \tag{3.44}$$

We have some evidence for the existence of a threshold probability, whose value is around 19% (see figure 3.5.2). Unfortunately simulating the system for 32 qubits or more is daunting and then figure 3.5.2 does not give very strong evidence of this fact. However, we considered other codes with the same properties of example 7 and they also seem to show a threshold probability around the same value, which makes us think that the value of such

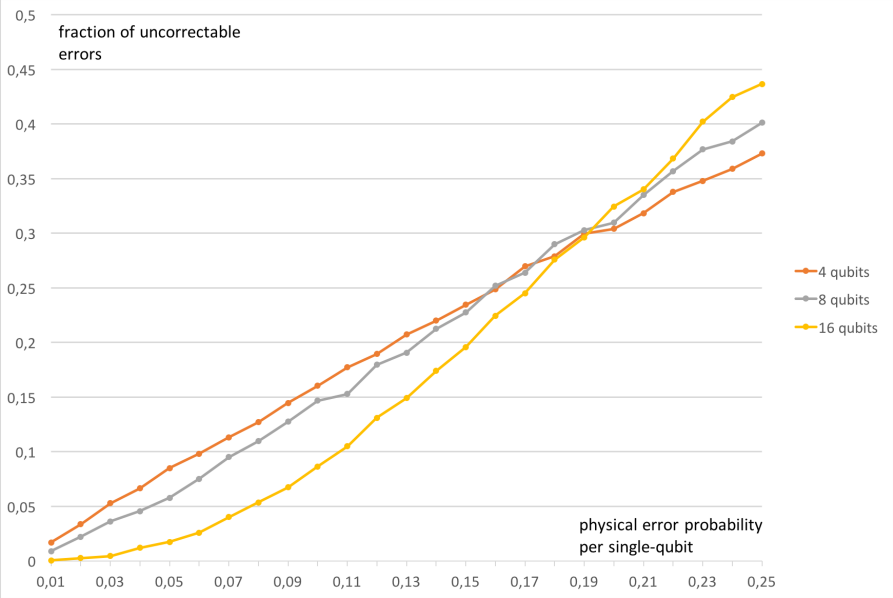

Figure 3.5.2: We show the results of the simulations we performed with example 7 in table 3.5.1. On the x-axis there is the physical error probability p for single-qubit i.i.d. Pauli noise (definitions 1.1.10 and 1.1.9), whereas on the y-axis we report our estimate for the logical error rate, evaluated with Monte Carlo methods as the fraction of times the smallest-weight recovery algorithm failed in correcting a Pauli error. We see that the lines cross around 19% and that below this value the logical error rate seems to decrease with the system size, whereas above it seems to increase, suggesting that $\bar{p} \simeq 0.19$ may be a threshold probability for this MERA stabilizer code.

a threshold probability should come just from geometric considerations (for codes satisfying the idealized K&K assumptions, of course).

A value of 19% would be enormous for a threshold probability, especially for a 1D system: for example for the toric code in 2D it is 10.3% w.r.t. Pauli noise using a minimal-matching decoder (see for example [29] for a review). The price we pay here is the non-locality of the stabilizer generators of our MERA code. We do not really know how to show explicitly that there is no choice for the stabilizer generators such that they are all local, however this must be the case because otherwise it would contradict the results of Bravyi, Poulin and Terhal [18, 27].

Candidate recovery algorithms and their properties

We then tried to construct a recovery algorithm (definition 1.5.7) able to win the efficient-recovery game (definition 3.5.3), i.e. we tried to construct a classical algorithm which takes as input an error syndrome $\boldsymbol{s}$ and outputs in polynomial time a Pauli operator compatible with $\boldsymbol{s}$ (definition 1.5.4), in such a way that there still exists a threshold probability (definition 1.5.8), even if it is lower than the one achievable with the smallest-weight recovery algorithm (whose corresponding recovery map is given in definition 1.5.6). If

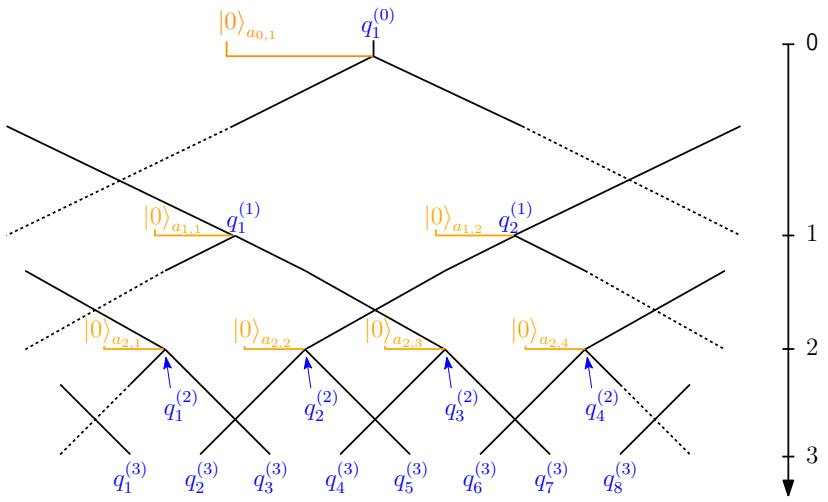

Figure 3.5.3: The i_j-th ancilla at level j is denoted by a_{j,i_j}, $i_j = 1 \dots 2^j$. Qubits al level j are denoted by $q_{i_j}^{(j)}$, $i_j = 1 \dots 2^j$. An operator acting on level j is denoted with a superscript j, e.g. $O^{(j)}$, meaning that it acts on $\bigotimes_{i_j=1}^{2^j}(\mathcal{H}_{a_{j,i_j}} \otimes \mathcal{H}_{q_{i_j}^{(j)}})$, where here $\mathcal{H} = \mathbb{C}^2$ because we consider qubits.

a recovery algorithm does not manage to win the efficient-recovery game, we say that such an algorithm "failed". In this section we examine several natural recovery algorithms for stabilizer MERA codes, that we call *type I*, *type IIa* and *type IIb* (they failed).

We introduce first some notation, exemplified in figure 3.5.3. In the encoding direction, at each step j we introduce 2^j new ancillas in the state $|0\rangle$. We label by a_{j,i_j} the i_j-th ancilla introduced at level j, and we denote its initial state as $|0\rangle_{a_{j,i_j}}$, where $i_j = 1 \dots 2^j$. Moreover, qubits al level j are denoted by $q_{i_j}^{(j)}$, $i_j = 1 \dots 2^j$. An operator acting on level j is denoted with a superscript j, e.g. $O^{(j)}$, meaning that it acts on

$$\bigotimes_{i_j=1}^{2^j}\left(\mathcal{H}_{a_{j,i_j}} \otimes \mathcal{H}_{q_{i_j}^{(j)}}\right), \tag{3.45}$$

where here $\mathcal{H} = \mathbb{C}^2$ because we consider qubits. We denote an operator at level j acting non-trivially only over an ancilla a_{j,i_j} as $O_{a_{j,i_j}}^{(j)}$. Operators at level j can be mapped at lower levels by the encoding map $\mathcal{E}$ (definition 2.2.1). We call $\tilde{\mathcal{E}}_{j_1}^{j_2}$ the encoding map from level j_1 to level $j_2 \geq j_1$ constructed using the unitary extension (definition 2.1.1) of the MERA instead of the MERA itself. Then we define

$$O_{a_{j,i_j}}^{(j_2)} := \tilde{\mathcal{E}}_j^{j_2}\left(O_{a_{j,i_j}}^{(j)}\right) \tag{3.46}$$

(see figure 3.5.4 for an example) for $j_2 \geq j$. One of the most relevant cases

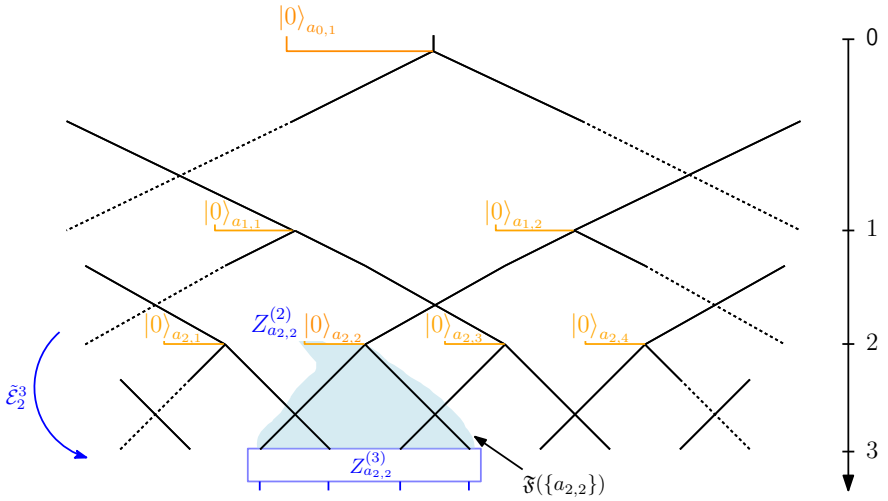

Figure 3.5.4: $Z^{(2)}_{a_{2,2}}$ is a single-qubit Pauli Z acting on ancilla $a_{2,2}$ at level 2. The encoding map $\tilde{\mathcal{E}}^3_2$ maps $Z^{(2)}_{a_{2,2}}$ to $Z^{(3)}_{a_{2,2}}$, which is supported at most on the qubits in the future causal cone $\mathfrak{F}(\{a_{2,2}\})$ of $a_{2,2}$ (not necessarily supported non-trivially on each of them, but at least on one). In particular, this means that $Z^{(3)}_{a_{2,2}}$ in general has non-identity tensor product factors over more than one qubit, whereas $Z^{(2)}_{a_{2,2}}$ only over a single one.

is when O is Pauli Z, because $Z^{(j)}_{a_{j,i_j}}$ is the trivial stabilizer of $|0\rangle_{a_{j,i_j}}$, and therefore $Z^{(j_2)}_{a_{j,i_j}}$ is the corresponding stabilizer at level j_2 (i.e. it is a stabilizer of the stabilizer code represented by the MERA when it has a total of j_2 levels). Notice that $Z^{(j_2)}_{a_{j,i_j}}$ for $j_2 > j$ is in general a multi-qubit Pauli operator, whereas $Z^{(j)}_{a_{j,i_j}}$ is a single-qubit one (see also figure 3.5.4).

In the following, consider a MERA with N total encoding levels. Recall that, when we say that an ancilla *switches*, we mean that its state changes from $|0\rangle$ to $|1\rangle$ modulo phases (there are no other possibilities for stabilizer MERA codes).

Type I. Notice that, given a syndrome $\boldsymbol{s}$ (definition 1.5.4) caused by an error $P^{(N)}_{\boldsymbol{s}}$, it is easy to find a Pauli operator compatible with it: write $\boldsymbol{s}$ as

$$\boldsymbol{s} = (s_{0,1}, s_{1,1}, s_{1,2}, \dots, s_{j,i_j} \dots, s_{N-1,2^{N-1}}), \tag{3.47}$$

i.e. we order the syndrome bits starting from one bit corresponding to the first ancilla at the top, then two bits corresponding to the two ancillas introduced at level 1 and so on. If $s_{j,i_j} = -1$, it means that the ancilla a_{j,i_j} has switched from $|0\rangle_{a_{j,i_j}}$ to $|1\rangle_{a_{j,i_j}} = X^{(j)}_{a_{j,i_j}} |0\rangle_{a_{j,i_j}}$. It follows that a way to make this ancilla switch back to $|0\rangle_{a_{j,i_j}}$ is to act on the bottom level with $X^{(N)}_{a_{j,i_j}}$, because this operator is renormalized to $X^{(j)}_{a_{j,i_j}}$ by definition and it cancels the other one

($X^2 = I$). Notice that all the other ancillas do not change if we apply $X^{(N)}_{a_{j,i_j}}$, hence the Pauli operator

$$\bar{P}^{(N)}_{\boldsymbol{s}} \coloneqq \prod_{j,i_j\,:\,s_{j,i_j}=-1} X^{(N)}_{a_{j,i_j}} \tag{3.48}$$

is compatible with $\boldsymbol{s}$. Recall that, because we are considering only Clifford operators, the encoding and renormalization of Pauli operators can all be represented in terms of binary vectors and binary symplectic matrices through the isomorphism (3.30). Therefore, the computation of $\bar{P}^{(N)}_{\boldsymbol{s}}$ takes at most $\mathcal{O}(n^4 \log n)$ time, where $n = 2^N$ is the system size ($n-1$ ancillas, for each one at most $\log_2 n$ multiplications of matrices with dimension smaller than $n \times n$).

It is intuitively understandable (and we checked it numerically) why (3.48) is not a good choice to try to correct the error $P^{(N)}_{\boldsymbol{s}}$ that caused the syndrome $\boldsymbol{s}$: because the width of the future causal cone of an ancilla increases exponentially (it is immediate to extend the definition 2.1.4 of future causal cone to the unitary extension of the MERA), we have that in general $X^{(N)}_{a_{j,i_j}}$ can be supported on a large number of qubits (in the same way as Z in figure 3.5.4), but this means that very likely $\bar{P}^{(N)}_{\boldsymbol{s}} P^{(N)}_{\boldsymbol{s}}$ has a high weight, which makes it a good candidate to be a logical error.

Notice that if we multiply $\bar{P}^{(N)}_{\boldsymbol{s}}$ by any $Z^{(N)}_{a_{j,i_j}}$, $j = 0 \dots N-1$ and $i_j = 1 \dots 2^j$, then we obtain another Pauli operator which is compatible with the given syndrome. Indeed, all Pauli operators with this property are obtained multiplying $\bar{P}^{(N)}_{\boldsymbol{s}}$ by any number of elements in the set $\{Z^{(N)}_{a_{j,i_j}}\}$, so that in total we get 2^{n-1} such Pauli operators. It would be desirable to have an efficient algorithm which reduces the weight of $\bar{P}^{(N)}_{\boldsymbol{s}}$ till we get one of the smallest weight Pauli operators among those 2^{n-1} ones. However, at best of our knowledge, such an algorithm is not available, so that in principle one has to check all the possibilities, but they are exponentially many.

We verified the performance of a randomized heuristic algorithm: for M times we pick a random number of elements in the set $\{Z^{(N)}_{a_{j,i_j}}\}$ and multiply $\bar{P}^{(N)}_{\boldsymbol{s}}$ by each of those selected elements; if the new weight is smaller we save such choice, otherwise we discard it and try another one. However, we saw from our simulations that this randomized algorithm has a very poor performance unless M is quite large, but then this slows down the computation a lot, also because we repeat the procedure M times even if e.g. after the first one we have already found the smallest-weight error. In particular, it slows it down so much that it is preferable in general to use simply the smallest-weight recovery algorithm.

Type IIa. The main idea of type IIa and IIb recovery algorithms is to exploit the local structure of MERA to construct a Pauli operator $\bar{P}^{(N)}_{\boldsymbol{s}}$ with a (hopefully) small weigh, compatible with a given syndrome $\boldsymbol{s}$ caused by an error $P^{(N)}_{\boldsymbol{s}}$. We call $\bar{P}^{(N)}_{\boldsymbol{s}}$ a "correction". We want to proceed level by level of

the MERA, in the sense that we want to fix the ancillas at the bottom level first, then the ones one level higher and so on (see also figure 3.5.5).

An important observation is that if an ancilla a_{j,i_j} switches, then it means that the error $P_{\boldsymbol{s}}^{(N)}$ *must* have at least one tensor-product factor which is an X, Y or Z acting on one of the qubits belonging to the intersection of level N and of the future causal cone $\mathfrak{F}(\{a_{j,i_j}\})$ (definition 2.1.4) of such an ancilla. Hence we want to construct $\bar{P}_{\boldsymbol{s}}^{(N)}$ by putting at least one X, Y or Z in the future causal cone of every ancilla switching, looking for the right small-weight combination which makes such an ancillas actually switch back to $|0\rangle$, but which at the same time does not make switch the ancillas that were already fine. Notice that, on the contrary, if an ancilla does not switch, it does not mean that there is no non-identity tensor-product factor of $P_{\boldsymbol{s}}^{(N)}$ in its future causal cone.

The correction $\bar{P}_{\boldsymbol{s}}^{(N)}$ is constructed as

$$\bar{P}_{\boldsymbol{s}}^{(N)} = \prod_{j=1}^{N} \tilde{\mathcal{E}}_j^N(\tilde{P}_{\boldsymbol{s}}^{(j)}) \tag{3.49}$$

where $\tilde{P}_{\boldsymbol{s}}^{(j)}$ at level j is computed as we are going to explain. We start computing $\tilde{P}_{\boldsymbol{s}}^{(N)}$, then $\tilde{P}_{\boldsymbol{s}}^{(N-1)}$ and so on, because the construction of $\tilde{P}_{\boldsymbol{s}}^{(j)}$ depends on all the previous $\{\tilde{P}_{\boldsymbol{s}}^{(i)}\}_{i=j+1}^{N}$. See figure 3.5.6 exemplifying a few steps of the computation.

Algorithm to compute $\tilde{P}_{\boldsymbol{s}}^{(j)}$:

1. If $j = N$, let $P_{\text{temp}}^{(N)} = P_{\boldsymbol{s}}^{(N)}$, otherwise $P_{\text{temp}}^{(N)} = P_{\boldsymbol{s}}^{(N)} \prod_{i=j+1}^{N} \tilde{\mathcal{E}}_i^N(\tilde{P}_{\boldsymbol{s}}^{(i)})$.

2. Compute the following two sets (i.e. "compute the partial syndrome of $P_{\text{temp}}^{(N)}$ at level $j-1$"):

$$\begin{aligned} A_{|0\rangle}^{(j-1)} &= \{a_{j-1,i_{j-1}} : [P_{\text{temp}}^{(N)}, Z_{j-1,i_{j-1}}^{(N)}] = \mathbf{0}, i_{j-1} = 1 \dots 2^{j-1}\} \\ A_{|1\rangle}^{(j-1)} &= \{a_{j-1,i_{j-1}} : \{P_{\text{temp}}^{(N)}, Z_{j-1,i_{j-1}}^{(N)}\} = \mathbf{0}, i_{j-1} = 1 \dots 2^{j-1}\}. \end{aligned} \tag{3.50}$$

3. We label by $q_1^{(j)}, \dots, q_{2^j}^{(j)}$ the qubits at level j and we define $L^{(j)} = \{q_i^{(j)}, i = 1 \dots 2^j\}$. For every $a \in A_{|1\rangle}^{(j-1)}$ we call

$$F_a^{(j)} = \mathfrak{F}(\{a\}) \cap L^{(j)} \tag{3.51}$$

$$C_a^{(j-1)} = \mathfrak{C}(F_a^{(j)}) \cap A_{|0\rangle}^{(j-1)}, \tag{3.52}$$

where $\mathfrak{F}$ and $\mathfrak{C}$ are respectively the future and past causal cones (definitions 2.1.4 and 2.1.3). We also define

$$F^{(j)} = \{F_a, a \in A_{|1\rangle}^{(j-1)}\} \tag{3.53}$$

$$C^{(j-1)} = \bigcup_{a \in A_{|1\rangle}^{(j-1)}} C_a^{(j-1)}. \tag{3.54}$$

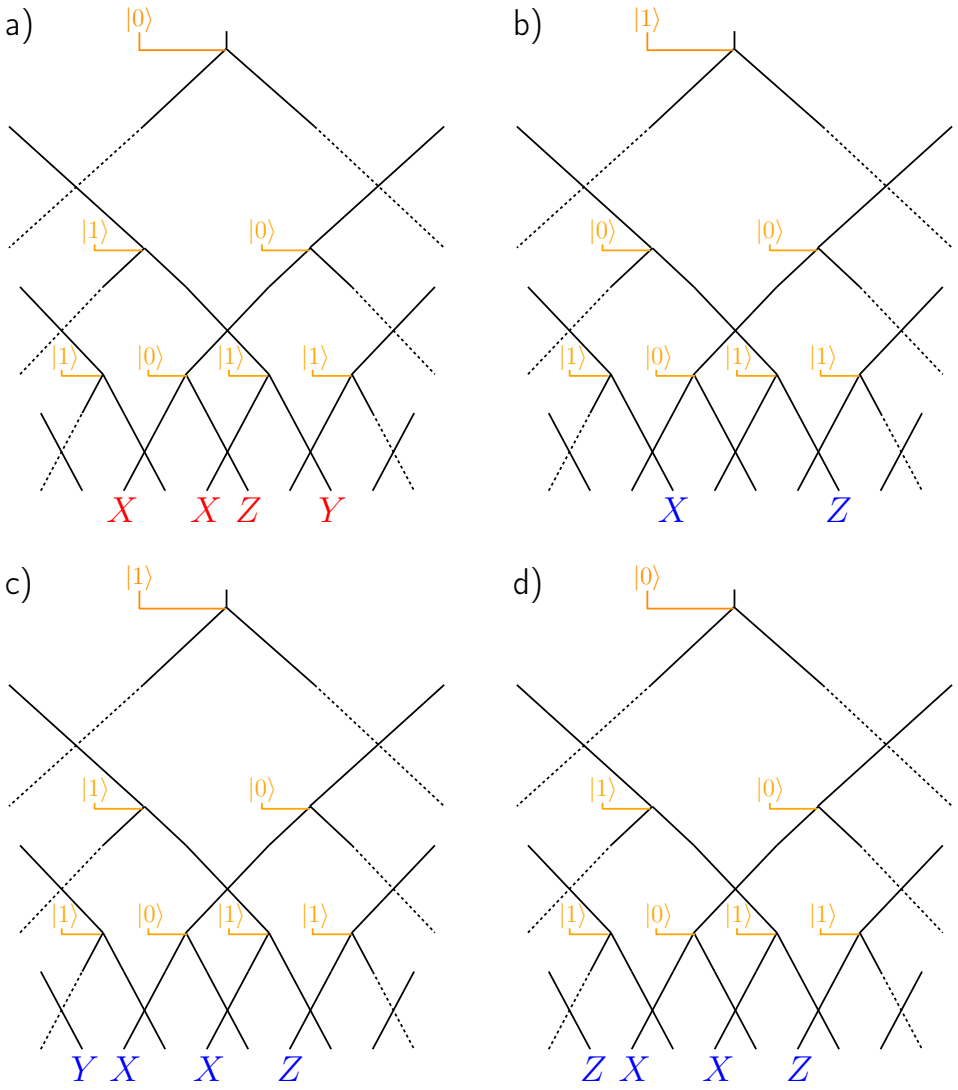

Figure 3.5.5: Here we show an example which is not specific to any code but it is just illustrative of how the recovery algorithm type IIa works. Let an error $P_{\boldsymbol{s}}^{(3)}$ be $P_{\boldsymbol{s}}^{(3)} = IXIXZIYI$ as shown (in red) in a). The error syndrome $\boldsymbol{s}$ of $P_{\boldsymbol{s}}^{(3)}$ is also shown in a), in terms of which ancillas switch or not. Figures b), c) and d) show the construction of a correction $\bar{P}_{\boldsymbol{s}}^{(3)}$ (ideally, the smallest-weight one) which has the same syndrome as $P_{\boldsymbol{s}}^{(3)}$. At each of those steps we update the correction (in blue) in such a way that at step b) its syndrome coincides with $\boldsymbol{s}$ at the lower level, at step c) at the two lowest levels and at step d) at all three levels. From d) we see that in this example $\bar{P}_{\boldsymbol{s}}^{(3)} = ZXIXIZII$ and so $P_{\boldsymbol{s}}^{(3)}\bar{P}_{\boldsymbol{s}}^{(3)} = ZIIIZZYI$. Moreover, by construction $P_{\boldsymbol{s}}^{(3)}\bar{P}_{\boldsymbol{s}}^{(3)}$ has the trivial syndrome (all ancillas in the state $|0\rangle$), because each ancilla either switch twice or none. We say that the recovery algorithm did not manage to correct $P_{\boldsymbol{s}}^{(3)}$ if $P_{\boldsymbol{s}}^{(3)}\bar{P}_{\boldsymbol{s}}^{(3)}$ does not belong to the stabilizer group of the given stabilizer MERA code.

4. We then look for the smallest-weight Pauli error $P^{(j)}$ at level j satisfying the following conditions:

   (a) $\forall F \in F^{(j)}$, $F \cap S_{P^{(j)}} \neq \emptyset$, where $S_{P^{(j)}}$ is the set of faulty locations of $P^{(j)}$, as defined with equation (1.5), i.e. $S_{P^{(j)}} \subseteq L^{(j)}$ is the set of qubits over which $P^{(j)}$ acts with a non-identity tensor product factor;

   (b) for all ancillas $a \in A^{(j-1)}_{|1\rangle}$, $\{Z^{(j)}_a, P^{(j)}\} = \mathbf{0}$;

   (c) for all ancillas $c \in C^{(j-1)}$, $[Z^{(j)}_c, P^{(j)}] = \mathbf{0}$.
   Remark: the ancillas with no X, Y or Z in their future causal cone can *not* switch, so they are not included here because it is geometrically guaranteed that they satisfy this condition.

   Such a task can be accomplished in more or less elaborate ways: the easiest one is just to consider all Pauli operators one by one, starting from the smallest weight ones, until we find the first one which satisfies all three conditions. Otherwise one can directly generate Pauli operators compatible with the first condition and then check just the last two.

5. We set $\tilde{P}^{(j)}_{\boldsymbol{s}}$ equal to the Pauli operator found at the previous step.

We observe that this algorithm is not really efficient, because for example to compute $\tilde{P}^{(N)}_{\boldsymbol{s}}$ in the worst-case there are $2^{N-1} = n/2$ ancillas switching and then one has to search a correction among all the possible 4^n Pauli operators. However, an advantage w.r.t. a recovery algorithm which considers the full error syndrome is that at each level there are fewer constraints that a correction has to satisfy, so that in general one has to check much fewer Pauli operators than 4^n in order to find one which is compatible with the above three conditions (and which has a small-weight). However, we saw from our simulations that the performance of this recovery algorithm is quite compromised and there is no threshold probability (or it is smaller than 10^{-7}).

One of the main problems of the type IIa recovery algorithm, in making a good guess to correct an error, can be understood in the following way: because we are considering i.i.d. Pauli noise (definition 1.1.9) and the corresponding probability distribution induced over Pauli operators, it is justified the strategy of picking $\tilde{P}^{(N)}_{\boldsymbol{s}}$ as the smallest-weight Pauli operator fixing the ancillas at the bottom level, however, at higher levels the probability distribution is not of i.i.d. kind anymore, but it depends on the specific Clifford unitaries which form the given MERA. Therefore, the higher we go the more likely it is that we choose to update the correction in a way that is not very likely to work.

Type IIb. Heuristically, we tried to solve that issue with the type IIb recovery algorithm, which we may also call the "minimal-updates" recovery algorithm. We start computing $\tilde{P}^{(N)}_{\boldsymbol{s}}$ using the same algorithm as the previous one, then we renormalise it by one level to $P^{(N-1)}_{\text{temp}} \equiv (\mathcal{E}^*)^{N-1}_N(\tilde{P}^{(N)}_{\boldsymbol{s}})$ where $\mathcal{E}^*$ is the renormalization map (definition 2.1.2).

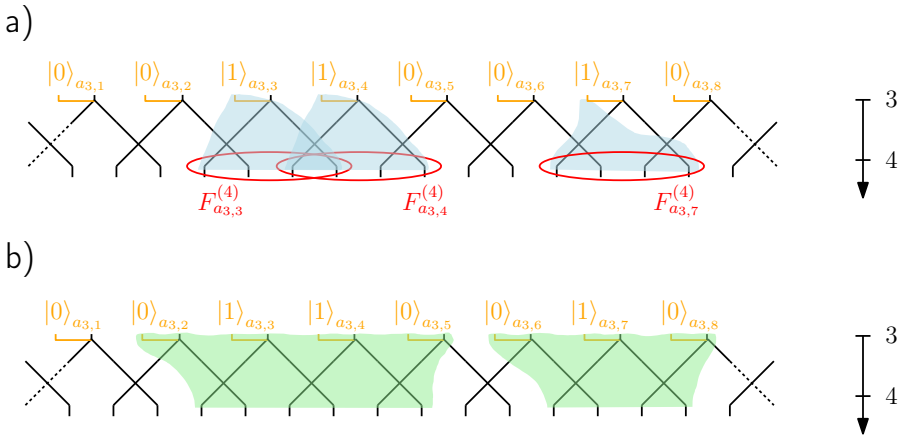

Figure 3.5.6: Here we pick $N = 4$ and explain graphically a few steps in the computation of $\tilde{P}^{(4)}_{\boldsymbol{s}}$, referring to the algorithm explained starting from equation (3.49). Because only the ancillas at level $N-1 = 3$ are involved, we do not draw the rest of the network. In this example assume that the states of the ancillas are found as shown. a) We have that $A^{(3)}_{|1\rangle} = \{a_{3,3}, a_{3,4}, a_{3,7}\}$ and $A^{(3)}_{|0\rangle} = \{a_{3,1}, a_{3,2}, a_{3,5}, a_{3,6}, a_{3,8}\}$, where these two sets are defined in (3.50). Then, for each $a \in A^{(3)}_{|1\rangle}$, $F^{(4)}_a$ includes the qubits we have circled in the first figure, where the $F^{(4)}_a$ are defined in (3.51). b) We consider the past causal cones of those qubits and we can see that $C^{(3)} = \{a_{3,2}, a_{3,5}, a_{3,6}, a_{3,8}\}$, where such a set is defined in (3.54).

We also compute the partial error syndrome of $P_{\text{temp}}^{(N-1)}$ at level $N-2$, similarly to step 2 in the previous algorithm. Now we look for the smallest-weight Pauli operator $\hat{P}^{(N-1)}$ such that $P^{(N-1)} \equiv \hat{P}^{(N-1)} P_{\text{temp}}^{(N-1)}$ satisfies the three conditions above. We proceed like this till the top, where we check if $(\mathcal{E}^*)_1^0(\tilde{P}_s^{(1)})$ is a logical operator or not, i.e. if it is an X, Y, Z or an I over the logical qubit al level 0. One may also compute the actual physical correction that one should apply at level N, but numerically that is not relevant to estimate the logical error rate and the threshold.

We call it the minimal-updates recovery algorithm because in this case we are renormalising to a target level the corrections computed at lower levels, and then trying to modify them as little as possible. In this way, we expected to find a correction more consistent with the probability distribution over Pauli operators at higher levels, but in the end the improvement in terms of performance of the recovery algorithm was negligible, if any.

We examined also other similar recovery algorithms, in which we proceeded for example 2-levels by 2-levels, instead of level by level. The performance improves approaching the one of the smallest-weight recovery algorithm, which uses the full syndrome information, but the computational cost increases as well. We believe that the problem with this approach is that, if we want to proceed k-levels by k-levels, where k is fixed, then for increasing number N of levels the ratio k/N gets smaller, so that this kind of recovery algorithm becomes less and less relatively powerful. It follows that for recovery algorithms of this kind we do not expect a threshold probability to exist.

Chapter 4

MERA Codes: Correctability of Arbitrary Noise

In this chapter we want to extend the results that have been derived by Kim & Kastoryano (K&K) [7] about the possibility of constructing a recovery map for MERA codes in the case of erasure errors (see chapter 3). Indeed, we want to consider arbitrary errors, in the sense that we do not want to assume that the location of the errors is known. The motivation comes from the fact that such an information is not generally available in a realistic situation, except in a few cases, namely when photon loss is experimentally the main source of errors.

Hence, we want to investigate under which circumstances we can still guarantee the existence of a "good" recovery map, even when the (precious) information about the location of errors is unavailable to us. By dropping such an assumption we can no longer use the theorems about local approximate quantum error correction (see section 1.4 for a summary of [17]) and so, as a tool, we will rely on the approximate Knill-Laflamme conditions derived by Bény & Oreshkov (see section 1.3 for a summary of [1]).

Remark 4.0.1. It is well-known (theorem 1.1.1, see chapter 7 of [8] for a discussion) that a code able to correct all erasures of $d-1$ qubits can also correct all errors on $(d-1)/2$ qubits, even without knowing their location, hence, as long as a noise channel has Kraus operators with weight smaller than the distance d of the code, then the "difference" between erasure and arbitrary noise is just a factor of 2, but, whenever the channel is a less restricted one, then we are entering the realm of approximate quantum error correction (section 1.3) and the previous connection breaks down, so that one has to rely on more elaborate techniques, like the ones mentioned above.

4.1 Tools

Consider a general noise channel $\mathcal{N}(\rho) = \sum_{i=1}^{N} E_i \rho E_i^\dagger$ and a (code) subspace $\mathcal{C}$ with projector Π. Referring to theorem 1.3.1 and corollary 1.3.1, we have

that $\mathcal{C}$ is an approximate quantum error correcting code w.r.t. $\mathcal{N}$, with accuracy parameter ϵ (definition 1.3.3), if and only if we can write

$$\Pi E_i^\dagger E_j \Pi = \lambda_{ij}\Pi + \Pi B_{ij}\Pi \tag{4.1}$$

where $\{\lambda_{ij}\}$ are the components of a density matrix $\lambda = (\lambda_{ij})_{ij}$ and the operators $\{B_{ij}\}$ are sufficiently "small". Note that such an expression above is not unique: first, we may consider different Kraus representations of $\mathcal{N}$, second, the separation on the r.h.s. into one part proportional to Π and the rest can be done in different ways. We call the $\{B_{ij}\}$ the (non-unique) *corrections* to the Knill-Laflamme conditions. By corollary 1.3.1, if the following inequality holds,

$$\max_{ij} \|\Pi B_{ij}\Pi\| \leq \frac{2}{N^2}\,\epsilon^2, \tag{4.2}$$

then $\mathcal{C}$ is an ϵ-QECC w.r.t. $\mathcal{N}$ (definition 1.3.3), where N is the number of Kraus operators in the chosen representation of $\mathcal{N}$. Notice that the $\{B_{ij}\}$ depend on such choice and on λ through equation (4.1).

Considering a system made up of n qubits, with Hilbert space $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$, we would ultimately like to consider i.i.d. noise (definition 1.1.5), i.e. $\mathcal{N} = \mathcal{N}_1^{\otimes n}$ where $\mathcal{N}_1$ is a single-qubit channel (qudits would work as well). In that case, *naively* N grows exponentially in the system size and this means that to prove ϵ-correctability one would need to find bounds on the corrections which are exponentially small in n. It will turn out that it is not easy to get that kind of bounds (the best bounds of this kind to our knowledge were found by Lee, Brell and Flammia in a paper [16] regarding the honeycomb model, in which the authors find bounds which are exponentially small in the *square root* of the system size, in the case in which the support of the errors is restricted to not-too-big simply connected region). Moreover, we will show that this is the reason why we will have to restrict just to *constant-support* Kraus operators. By constant-support we mean that the number of non-identity tensor product factors of any Kraus operator in a given representation is upper-bounded by a constant independent of the system size n, so that their number can increase at most polynomially, instead of exponentially.

Actually, the number N of Kraus operators describing $\mathcal{N}$ depends on the chosen representation and it can be taken as small as the dimension (definition A.0.3) $\dim \mathcal{N}|_{\mathcal{C}} \equiv \tilde{N}$ of the noise channel restricted to act on states supported on the code subspace (see appendix A), i.e. $\tilde{N}$ is the number of linearly independent Kraus operators of $\mathcal{N}|_{\mathcal{C}}$. Trivially we have that $\tilde{N} \leq (\dim \mathcal{C})^2$, where $\dim \mathcal{C}$ is the dimension of the code subspace. In the remainder of this chapter, otherwise stated we will consider a 1D binary MERA as in figure 2.1.1, for which we fix the number k of logical qubits, i.e. we fix the dimension of the code subspace, and for which we use different levels of encoding, corresponding to a number n of physical qubits which is growing with the depth of the MERA encoder. For fixed k, then, the upper bound $\tilde{N} \leq 2^{2k}$ is independent of the system size n and it seems that we can avoid the problem of finding bounds exponentially small in the system size in (4.2) by working with this

other Kraus representation for $\mathcal{N}|_{\mathcal{C}}$. This can be done, however it may be even harder to find bounds in the new representation.

We can give an intuitive explanation of why this is the case: consider for example $\mathcal{N}$ to correspond to Pauli noise, and let $\{E_i|_{\mathcal{C}}\}_{i=1}^{N}$ and $\{\tilde{E}_i|_{\mathcal{C}}\}_{i=1}^{\tilde{N}}$ be two Kraus representation of $\mathcal{N}|_{\mathcal{C}}$; in particular, let the former be the one in which each $\tilde{E}_i|_{\mathcal{C}}$ is proportional to a Pauli operator P_i (definition 1.1.7) and let the latter be a minimal representation. In the first case, by choosing Pauli operators we have a clear characterization of the weight of an Kraus operator and of the qubits on which it is supported. This is what will be helpful in the proofs in the next section, whereas it is not clear how to proceed in the second case.

One may also think that we may get a bound independent of the system size on $\|\Pi\tilde{B}_{ij}\Pi\|$, where $\Pi\tilde{B}_{ij}\Pi \equiv \Pi\tilde{E}_i^\dagger\tilde{E}_j\Pi - \tilde{\lambda}_{ij}\Pi$ for some $\tilde{\lambda}$, in the case in which we manage to get a bound on $\|\Pi B_{ij}\Pi\|$ which is not exponentially small in the system size, i.e. one may think that getting some not very useful bound in one representation may imply a good enough bound in another one. However, we can bring some evidence showing that this is not the case in general. Recall from appendix A that two representations of the same quantum channel are related by a unitary, or an isometry if one representation has more Kraus operators than the other one. In this case $\{\tilde{E}_i|_{\mathcal{C}}\}$ is the smaller set, hence we have that $\tilde{E}_i|_{\mathcal{C}} = \sum_j (V^\dagger)_{ij} E_j|_{\mathcal{C}}$ where V is an isometry and the indices i and j take respectively $\tilde{N}$ and N different values. So we have the transformation $\Pi\tilde{E}_i^\dagger\tilde{E}_j\Pi = \sum_{kl} V_{ki}V_{jl}^\dagger\,\Pi E_k^\dagger E_l\Pi$ and similarly

$$\left\|\Pi\tilde{B}_{ij}\Pi\right\| = \sum_{kl}\left\|V_{ki}V_{jl}^\dagger\,\Pi B_{kl}\Pi\right\|. \tag{4.3}$$

There is a caveat here, in the sense that $\tilde{B}_{ij}$ does not necessarily have to transform in this way, because in different representations the separation of $\Pi\tilde{E}_i^\dagger\tilde{E}_j\Pi$ into a density matrix and corrections as in (4.1) does not have to be the same. However, we do not know what is the best choice in general and it seems reasonable to use the above equation. Taking maximizations we get

$$\begin{aligned}
\max_{ij}\left\|\Pi\tilde{B}_{ij}\Pi\right\| &= \max_{ij}\sum_{kl}\left\|V_{ki}V_{jl}^\dagger\,\Pi B_{kl}\Pi\right\| && (4.4)\\
&\leq \max_{ij}\sum_{kl}|V_{ki}|\left|V_{jl}^\dagger\right|\,\|\Pi B_{kl}\Pi\| && (4.5)\\
&\leq N^2\Big(\max_{ki}|V_{ki}|\Big)\Big(\max_{jl}\left|V_{jl}^\dagger\right|\Big)\max_{kl}\|\Pi B_{kl}\Pi\| && (4.6)\\
&\leq N^2\max_{kl}\|\Pi B_{kl}\Pi\|. && (4.7)
\end{aligned}$$

Therefore, at least according to this inequality, if $\max_{kl}\|\Pi B_{kl}\Pi\|$ is sub-exponentially small in n, then we cannot prove that $\max_{ij}\|\Pi\tilde{B}_{ij}\Pi\|$ is upper-bounded by a constant independent of the system size (but it could still be possible to do that by some other means).

4.2 Correctability of constant-support arbitrary errors

In this section we want to show that a MERA code, satisfying some assumptions that we specify in the following, is an approximate QECC w.r.t. noise channels with constant-support Kraus operators, as defined in the previous section.

First, consider the errors to have weight equal to 1, i.e. consider a noise channel whose Kraus operators $\{E_i\}$ can all be expressed as single-qubit operators (it is a very unrealistic situation and very far from i.i.d. noise, clearly). To simplify the notation, assume that for each qubit there exists only one Kraus operator non-trivially supported onto it: in this way we can label Kraus operators and the position on the lattice with the same index, e.g. E_i is the Kraus operator supported on the i-th qubit (the results we will present generalize easily to more than one Kraus operators per qubit).

Following Kim & Kastoryano [7], we assume that there is only one type of elementary block (definition 3.1.2), and so only one transfer operator ϕ, such that ϕ is diagonalizable and

$$\phi(O) = \sum_{k\geq 0} \lambda_k \operatorname{Tr}(R_k^\dagger O)\, L_k \tag{4.8}$$

where $\lambda_0 = 1$, $|\lambda_k| < 1$ for $k \neq 0$, $L_0 = \mathbb{1}$, R_0 is a density matrix corresponding to the unique stationary state of the channel and $\operatorname{Tr}(R_l^\dagger L_k) = \delta_{kl}$ (for a careful discussion see section 3.1 and appendix A.1). To simplify the analysis, here we ignore again the fact that we already know that in general there are more than one transfer operator and that ϕ is not diagonalizable. However, the result can be generalized using what we mention in section A.1 or using more than one set of eigenvalues in the following, but that would just make the notation more cumbersome and it does not change the conclusion qualitatively.

Recall from the previous section that we want to find expressions for $\Pi E_i^\dagger E_j \Pi$ in the form of (4.1), i.e. we want to separate (a little bit heuristically) the action of $E_i^\dagger E_j$ into some part proportional to the identity on the code subspace and some "corrections". The code subspace projector Π on the image of an isometric encoder $\mathcal{E}$ is given by $WW^\dagger$, where $\mathcal{E}(\rho) = W\rho W^\dagger$ and W is an isometry, in our case the MERA itself. Hence

$$\Pi E_i^\dagger E_j \Pi = W\Big(W^\dagger E_i^\dagger E_j W\Big)W^\dagger = \Pi W \underbrace{\Big(W^\dagger E_i^\dagger E_j W\Big)}_{(\mathcal{E}^*)_0^s(E_i^\dagger E_i)} W^\dagger \Pi. \tag{4.9}$$

We can see that in parenthesis it appears the renormalization map $\mathcal{E}^*$ (definition 2.1.2) from the physical level 0 to the top of the network at level s (for a 1D binary MERA we have $n = k2^s$ where s is the depth of the network, see figure 2.1.1). Using the notation introduced at the beginning of chapter 3, we denote by $(\mathcal{E}^*)_{s_1}^{s_2}$ the renormalization map from level s_1 to level $s_2 \geq s_1$. Then we

can use the assumptions on ϕ whenever $E_i^\dagger E_j, E_i^\dagger, E_j, (\mathcal{E}^*)_0^{\tilde{s}}(E_i^\dagger E_j), (\mathcal{E}^*)_0^{\tilde{s}}(E_i^\dagger)$ and/or $(\mathcal{E}^*)_0^{\tilde{s}}(E_j)$ are supported on an elementary block, for any $\tilde{s} = 1 \dots s-1$.

We consider three cases in which we can choose a pair of errors E_i and E_j. The separation into such three cases is quite arbitrary and in the following we will explain the reason of this choice and when it is not really useful to make this distinction. Case 1 is peculiar but it can actually be included in case 2, and we will do that to simplify some formulas.

Case 1: $i = j$. In this case $E_i^\dagger E_i$ is supported on an elementary block, supposing that each qubit is included in some elementary block (which is the case for example for a binary MERA). It follows that

$$\Pi E_i^\dagger E_i \Pi = \Pi W\Big((\mathcal{E}^*)_0^s(E_i^\dagger E_i)\Big)W^\dagger \Pi \tag{4.10}$$

$$= \Pi W\Big(\phi_0^s(E_i^\dagger E_i)\Big)W^\dagger \Pi \tag{4.11}$$

$$= \operatorname{Tr}(R_0 E_i^\dagger E_i)\Pi + \Pi\Big(\sum_{k\neq 0} \lambda_k^s \operatorname{Tr}(R_k^\dagger E_i^\dagger E_i) W L_k W^\dagger\Big)\Pi. \tag{4.12}$$

Case 2: i "close" to j. Recall that, for our choice of noise channel, two errors E_i and E_j are separated on the physical level by a distance $|i-j|$ in terms of lattice spacing. Such a distance more or less halves at each renormalization step, till the past causal cones of qubit i and qubit j intersect at a scale $r_{ij} \sim \log_2(|i-j|)$. Then the two errors are not separate anymore but they are "scrambled" and renormalized together by the renormalization map.

Definition 4.2.1 (Close and far errors)**.** We say that E_i and E_j are *close* if $r_{ij} \leq s/2$ and *far* otherwise (see also figure 4.2.1).

When two errors are close we can write:

$$\Pi E_i^\dagger E_j \Pi = \operatorname{Tr}(R_0 (E_i^\dagger E_j)^{s/2})\Pi + \Pi\Big(\sum_{k\neq 0} \lambda_k^{s/2} \operatorname{Tr}(R_k^\dagger (E_i^\dagger E_j)^{s/2}) W L_k W^\dagger\Big)\Pi \tag{4.13}$$

where we use the notation $(E_i^\dagger E_j)^{s/2} = (\mathcal{E}^*)_0^{s/2}(E_i^\dagger E_j)$.

i "far" from j. When two elementary blocks are separated by a distance large enough (not necessarily when they are far according to our definition), then the operators supported onto them can be renormalized independently across a few levels, before their past causal cones (definition 2.1.3) intersect (see figure 4.2.1). In the case where E_i is far from E_j, we have that $(\mathcal{E}^*)_0^{r_{ij}}(E_i^\dagger E_j) = \phi_0^{r_{ij}}(E_i^\dagger)\phi_0^{r_{ij}}(E_j)$, where for far errors $r_{ij} > s/2$. Then

$$\begin{aligned}\Pi E_i^\dagger E_j \Pi = \operatorname{Tr}(R_0 E_i^\dagger)\operatorname{Tr}(R_0 E_j)\Pi &+ \Pi\Big(\operatorname{Tr}(R_0 E_i^\dagger)\sum_{k\neq 0}\lambda_k^s \operatorname{Tr}(R_k^\dagger E_j) W L_k W^\dagger\Big)\Pi \\ &+ \Pi\Big(\operatorname{Tr}(R_0 E_j)\sum_{k\neq 0}\lambda_k^s \operatorname{Tr}(R_k^\dagger E_i^\dagger) W L_k W^\dagger\Big)\Pi \\ &+ \Pi \sum_{k,l\neq 0} (\lambda_k\lambda_l)^{r_{ij}} \operatorname{Tr}(R_k^\dagger E_i^\dagger)\operatorname{Tr}(R_l^\dagger E_j) W((\mathcal{E}^*)_{r_{ij}}^s(L_k \otimes L_l))W^\dagger \Pi.\end{aligned} \tag{4.14}$$

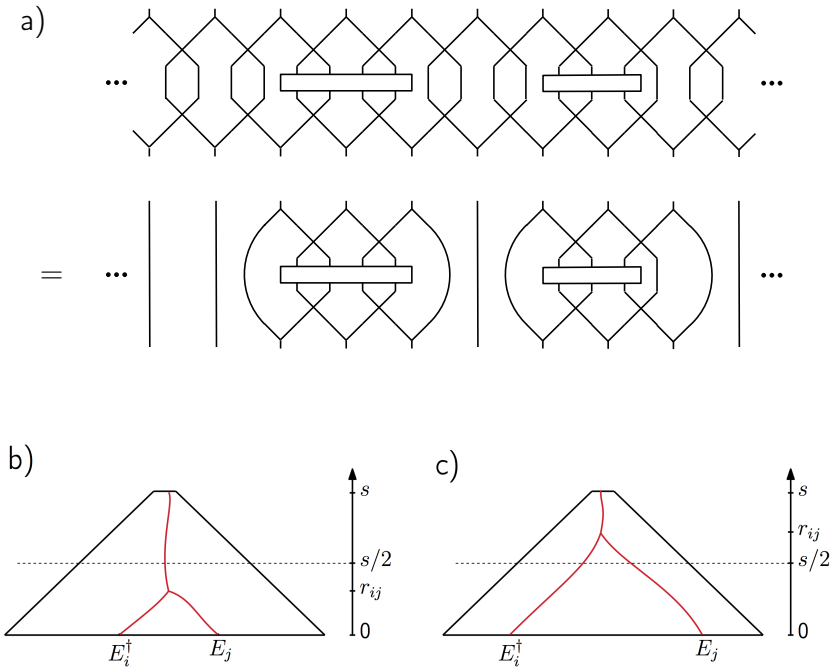

Figure 4.2.1: a) If two operators have non-trivial supports which are far enough (not only when their support is contained in an elementary block), or if a single operator has a support which can be divided into far-enough different clusters, then they can be renormalized independently (graphical computations similar to figure 2.1.4). b) The past causal cones of close errors $E_i^\dagger, E_j$ intersect before half of the network at a scale $r_{ij} \leq s/2$. c) In the case of far errors they intersect after half of the network.

To summarize, we can define

$$\begin{aligned}\lambda_{ij} \coloneqq{}& \delta(r_{ij} \leq s/2)\, \mathrm{Tr}(R_0 (E_i^\dagger E_j)^{s/2}) \\ &+ \delta(r_{ij} > s/2)\, \mathrm{Tr}(R_0 E_i^\dagger)\, \mathrm{Tr}(R_0 E_j)\end{aligned} \tag{4.15}$$

$$\begin{aligned}\Pi B_{ij} \Pi \coloneqq{}& \delta(r_{ij} \leq s/2)\, \Pi\Big(\sum_{k\neq 0} \lambda_k^{s/2}\, \mathrm{Tr}(R_k^\dagger (E_i^\dagger E_j)^{s/2}) W L_k W^\dagger\Big)\Pi \\ &+ \delta(r_{ij} > s/2)\, \Bigg(\Pi\Big(\mathrm{Tr}(R_0 E_i^\dagger) \sum_{k\neq 0} \lambda_k^s\, \mathrm{Tr}(R_k^\dagger E_j) W L_k W^\dagger\Big)\Pi \\ &\qquad + \Pi\Big(\mathrm{Tr}(R_0 E_j) \sum_{k\neq 0} \lambda_k^s\, \mathrm{Tr}(R_k^\dagger E_i^\dagger) W L_k W^\dagger\Big)\Pi \\ &+ \Pi \sum_{k,l\neq 0} (\lambda_k \lambda_l)^{r_{ij}}\, \mathrm{Tr}(R_k^\dagger E_i^\dagger)\, \mathrm{Tr}(R_l^\dagger E_j) W((\mathcal{E}^*)^s_{r_{ij}}(L_k \otimes L_l)) W^\dagger \Pi\Bigg).\end{aligned} \tag{4.16}$$

These are our "candidates" for the density matrix and the corrections in the approximate Knill-Laflamme conditions (4.1).

Definition 4.2.2 (Density-matrix assumption)**.** We call the *density-matrix assumption* the hypothesis for which the matrix $\lambda = (\lambda_{ij})_{ij}$ with entries given by (4.15) is a positive matrix and therefore a density matrix (it is clear that the trace is 1 because $\sum_i \mathrm{Tr}(R_0 (E_i^\dagger E_i)^{s/2}) = \mathrm{Tr}(R_0 (\sum_i E_i^\dagger E_i)^{s/2}) = \mathrm{Tr}(R_0) = 1$, using the fact that $\sum_i E_i^\dagger E_i = \mathbb{1}$).

A sketch of the proof that such assumption holds in general would be the following. Define a matrix $\tilde{\lambda}^{(\psi)}$ with entries $\tilde{\lambda}^{(\psi)}_{ij} = \langle\psi| E_i^\dagger E_j |\psi\rangle$, where $|\psi\rangle$ is a code state. We have that $\tilde{\lambda}^{(\psi)}$ is positive because $\forall\, |\alpha\rangle = (\alpha_1, \ldots, \alpha_n) \in \mathbb{C}^n$

$$\langle\alpha|\tilde{\lambda}^{(\psi)}|\alpha\rangle = \sum_{ij} \bar{\alpha}_i \alpha_j\, \langle\psi| E_i^\dagger E_j |\psi\rangle = \| E\,|\psi\rangle \|^2 \geq 0 \tag{4.17}$$

where $E \coloneqq \sum_j \alpha_j E_j$. Moreover

$$\mathrm{Tr}(\tilde{\lambda}^{(\psi)}) = \sum_i \langle\psi| E_i^\dagger E_i |\psi\rangle = \langle\psi|\psi\rangle = 1. \tag{4.18}$$

Therefore $\tilde{\lambda}^{(\psi)}$ is a density matrix $\forall\, |\psi\rangle$. Then for all $k \neq 0$ we would like to take the "limit" $\lambda_k \to 0$ in equation (4.16), while keeping R_0 fixed (as in the previous equations, R_0 is the stationary state of the channel ϕ^*). It would follow from taking this limit that $\tilde{\lambda}^{(\psi)} \to \lambda$ for all $|\psi\rangle$ and so λ would also be a density matrix. However, we are not really sure if we can take this limit, which corresponds to the idealized K&K assumptions (definition 3.1.4) to be satisfied. In particular, we already know that they cannot really be satisfied by any 1D binary MERA (see section 3.5).

Theorem 4.2.1. *Consider a 1D binary MERA code $\mathcal{C}_s$ satisfying the Kim & Kastoryano assumptions (definition 3.1.4) and the density-matrix assumption (definition 4.2.2), with s levels of encoding, k logical qubits and $n = k2^s$ physical qubits. Suppose that the noise channel is $\mathcal{N}(\rho) = \sum_{i=1}^{n} E_i \rho E_i^\dagger$ where E_i is non-trivially supported only on qubit number i in the lattice or is proportional to the identity. Moreover, suppose that $\nu > 4$, where ν is the scaling dimension of the MERA (definition 3.1.5). Then $\forall \epsilon > 0$ there exists an s sufficiently large such that $\mathcal{C}_s$ is an approximate quantum correcting code with approximation parameter ϵ (definition 1.3.3).*

Proof. We need to find estimates on the corrections in (4.16). Similarly to how we proved theorem 3.1.1, for close errors we have that:

$$\|\Pi B_{ij} \Pi\| \leq 2^{-\nu s/2} \left\| E_i^\dagger E_j \right\| \sum_{k \neq 0} \|L_k\| \, \|R_k\|_1 \tag{4.19}$$

$$\leq 2^{-\nu s/2} \sum_{k \neq 0} \|L_k\| \, \|R_k\|_1 \tag{4.20}$$

$$\equiv C \, 2^{-\nu s/2}. \tag{4.21}$$

On the other hand, for far errors we have:

$$\|\Pi B_{ij} \Pi\| \leq 2C \, 2^{-\nu s} + C^2 \, 2^{-2\nu r_{ij}} \tag{4.22}$$

$$\leq (2C + C^2) \, 2^{-\nu s}. \tag{4.23}$$

Overall these results imply that

$$\max_{ij} \|\Pi B_{ij} \Pi\| \leq \max\{C \, 2^{-\nu s/2}, \, (2C + C^2) \, 2^{-\nu s}\}. \tag{4.24}$$

Suppose that the first term is greater. Recall that $s = \log_2(n/k)$ and so $2^{-\nu s/2} = (n/k)^{-\nu/2}$. It follows that

$$n^2 \max_{ij} \|\Pi B_{ij} \Pi\| \leq \mathcal{O}(n^{2-\frac{\nu}{2}}) \tag{4.25}$$

and therefore if $\nu > 4$ the r.h.s. is smaller than any fixed $\epsilon > 0$ for n sufficiently large (which means s sufficiently large). The theorem follows from corollary 1.3.1 and theorem 1.3.1 (here $N = n$).
Similarly, if the second term in (4.24) is greater, then we get $n^2 \max_{ij} \|\Pi B_{ij} \Pi\| \leq \mathcal{O}(n^{2-\nu})$, which for $\nu > 4$ can also be made arbitrarily small for a sufficiently large n. □

The above theorem can be generalized in many ways:

- First it holds for any kind of MERA (simply e.g. for a m-nary MERA r_{ij} is given by $\log_m |i - j|$ instead of $\log_2 |i - j|$) and in higher dimensions. In the latter case there is a dimensionality factor appearing in the system size: for example for a D-dimensional binary MERA $n = k2^{Ds}$ and this has the effect that we need to require $\nu > 4D$.

- There can be more than one single-qubit error per qubit. This simply at most multiplies the l.h.s. of (4.25) by a constant factor (there are 4 linearly independent errors on a qubit).

- We can consider errors supported on elementary blocks instead of a single qubit, so that the size of their support is upper-bounded by a constant, which is the size of the biggest elementary block. Then their number is polynomial in the system size. If the degree of the polynomial is m, then one needs $\nu > 4m$.

- We can also consider errors whose support has size upper-bounded by a constant M but where we do not require them to be supported within an elementary block. Then their number is still polynomial in n and to prove bounds on the corrections we need to generalise the technique we used above: we group the non-identity tensor product factors of $E_i^\dagger E_j$ into clusters of errors that we can renormalise independently; whenever the past causal cones of two clusters intersect, we renormalise them together and, whenever a cluster is supported on an elementary block, we use the assumptions on ϕ. If $n \gg M$, then for each choice of $E_i^\dagger E_j$ every cluster renormalizes at least over $s - \mathcal{O}(1)$ elementary blocks and so the norm of the corrections can be bounded by $\mathcal{O}(2^{-\nu(s-\mathcal{O}(1))})$.

4.3 Exact quantum error correction for arbitrary errors

The problem of generalising even more theorem 4.2.1 arises when we want to consider errors supported on an arbitrary number of sites, because then their number becomes exponentially large but we can get bounds on the corrections which are at most polynomially small in the system size (remember that $s = \log_2(n/k)$). Similarly to the way taken by Kim & Kastoryano, the only escape to this problem is to restrict to codes satisfying the idealized K&K assumptions (definition 3.1.4), which correspond to taking the "limit" $\nu \to +\infty$. Then it is not useful to make the distinction between close and far errors. In terms of the stabilizer MERA codes we studied in chapter 3, w.r.t. Pauli noise, it is only relevant whether $E_i^\dagger E_j$ renormalizes to some non-trivial Pauli operator on the logical qubit at the top without making any ancilla to switch (see section 3.5), because in that case it is a logical error. Then the minimal size of the support of such an error is the distance d of the code.

If some ancilla switches, this means that $\Pi E_i^\dagger E_j \Pi = \mathbf{0}$ (the null operator), whereas, if no ancilla switches but $\Pi E_i^\dagger E_j \Pi$ acts as (a multiple of) the identity on the top qubit, then it means that $\Pi E_i^\dagger E_j \Pi \propto \Pi$. To summarize, if we consider a noise channel whose Kraus operators $\{E_i\}$ all have weight smaller or equal to $(d-1)/2$, it follows that $E_i^\dagger E_j$ has weight at most $d-1$ and hence it is not a logical operator. In particular, the previous observations imply that in this case $\mathcal{N}$ satisfies the Knill-Laflamme conditions (section 1.2) on

the code subspace given by the MERA. Therefore, there exists a recovery map (the Knill-Laflamme recovery map, definition 1.2.1) such that $\mathcal{R}(\mathcal{N}(\rho)) = \rho$ for any code state ρ, i.e. a recovery map able to correct arbitrary errors supported on at most $(d-1)/2$ sites or any linear combination of them.

In chapter 3 we discussed lower bounds on d and we have provided evidence of the fact that there exists stabilizer MERA codes with a threshold probability (definition 1.5.8), that is, codes for which there exists a recovery map able to correct a number of errors much higher than just those with weight smaller than $(d-1)/2$ (specifically, a linear number of errors in the limit of large system sizes). We have also argued that it it seems difficult to find non-stabilizer codes satisfying the idealized K&K assumptions (definition 3.1.6), but if any exists, then similar bounds would hold.

Conclusions and Outlook

In this work we studied MERA codes, i.e. quantum error correcting codes whose corresponding encoder is given by a quantum circuit with the structure of a MERA. In particular, in section 3.5 we examined 1D stabilizer binary-MERA codes and we analysed their properties:

- in general they are non-local codes (according to definition 1.5.3), which on the one side makes it unclear how to realise them in practice, but on the other side allows them to have a better distance than local codes, which are subject to the bound $d \leq \mathcal{O}(1)$ [18, 27]. More specifically, the bound for generic MERA codes derived by Kim & Kastoryano [7] is

$$d \geq \frac{1}{2}\Big(\frac{n}{k}\Big)^{0.63}, \tag{4.26}$$

  in the case in which the code satisfies the "idealized K&K assumptions" (definition 3.1.6), where n and k are respectively the number of physical and logical qubits (see theorem 3.2.3). We have argued in section 3.5 that the scaling and the pre-factor should be modified because of some discretization and finite-size effects which are ignored in theorem 3.2.3, and because there is no code satisfying precisely the assumptions 3.1.6, as we have verified numerically by exhaustive search (at least for 1D binary MERAs).

- For some examples of stabilizer MERA codes (see table 3.5.1) we have found not only that their distance seems to increase with the system size (computing the distance is exponentially hard, so that we could run our simulations only using a few qubits), but also that they seem to show a high threshold probability (definition 1.5.8), around 19% (see figure 3.5.2), w.r.t. Pauli noise and the smallest-weight recovery algorithm (definitions 1.1.9 and 1.5.6). Finding the smallest-weight error compatible with a given syndrome is also exponentially hard in general, therefore we examined many ways of constructing an efficient recovery algorithm w.r.t. which there still exists a threshold probability, but none of our trials showed such properties (see the last paragraph of section 3.5).
  These examples of interesting codes are those which "almost" satisfy the idealized K&K assumptions, in the sense that they satisfy them across two and not just one renormalization steps (a broader discussion is in

section 3.5). Therefore, we can say that it was a good intuition to pick those assumptions in order to find an interesting code.

In chapter 4 we proved bounds on the error-correcting capabilities of generic MERA codes against arbitrary noise channels, assuming that such codes satisfy the (non-idealized) K&K assumptions (definition 3.1.4) and the density-matrix assumption (definition 4.2.2). In particular, we generalized the results of Kim & Kastoryano (section 3.2), which hold only in the case of erasure noise, to more general channels: in theorem 4.2.1 we showed that noise channels, whose Kraus operators are non-trivially supported on a number of physical qubits which is upper-bounded by a constant independent of the system size, are correctable in the sense of approximate quantum error correction (definition 1.3.3) if the system size is sufficiently large.

That is still quite restrictive in terms of the class of channels we allow, and our problem in general falls into the current topic of research which tries to prove how good a given code is in correcting against a generic noise channel, even (and especially) in the case in which the given code is a stabilizer code known to perform pretty well against Pauli noise.

To summarize, we think that MERA codes form an interesting class, among which it is possible to find examples that seem to show a high threshold probability – which is a quite remarkable fact – at the price of sacrificing locality. We leave to future work to confirm the existence of such a threshold probability, using more powerful numerical or analytical tools. Another open issue is whether for stabilizer MERA codes there exists an efficient recovery algorithm admitting a threshold probability. Moreover, we have not considered at all the issue of fault tolerance; in particular, we do not know by how much the threshold would change in the presence of measurement errors.

Appendix A

Quantum Channels

Here we review some basic facts about quantum channels. Proofs and a more detailed discussion can be found in [13].

Definition A.0.1 (Quantum channel)**.** A quantum channel $\mathcal{T}\colon \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H}')$ is a map with the following properties:

1. linear;
2. completely positive, i.e. $\forall n \in \mathbb{N}$ we have $\mathcal{T} \otimes \mathrm{id}_n \geq 0$, where id_n is the identity operator on $\mathcal{M}_n(\mathbb{C})$;
3. trace-preserving, i.e. $\mathrm{Tr}(\mathcal{T}(\rho)) = \mathrm{Tr}(\rho)\ \forall \rho \in \mathcal{B}(\mathcal{H})$.

The above properties assure that $\mathcal{T}$ maps any density matrix to another density matrix. Note that we need complete positivity and not just positivity because we reasonably want a state to be mapped to a state also when we trivially include in our description any other system playing the role of a spectator. We also define

Definition A.0.2 (Dual of a quantum channel)**.** Given a quantum channel $\mathcal{T}\colon \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H}')$, we define a map $\mathcal{T}^*\colon \mathcal{B}(\mathcal{H}') \to \mathcal{B}(\mathcal{H})$ as the map such that

$$\mathrm{Tr}(O\,\mathcal{T}(\rho)) = \mathrm{Tr}(\mathcal{T}^*(O)\rho) \quad \forall \rho \in \mathcal{B}(\mathcal{H}), \forall O \in \mathcal{B}(\mathcal{H}') \tag{A.1}$$

and we call $\mathcal{T}^*$ the *dual* of the quantum channel $\mathcal{T}$.

The simplest examples of quantum channels are given by:

- $\mathcal{T}(\rho) = \rho \otimes \tilde{\rho}$ for some fixed state $\tilde{\rho}$, which w.l.o.g. can be taken to be pure because we can always consider the purification of $\tilde{\rho}$;
- $\mathcal{T}(\rho) = \mathrm{Tr}_A(\rho)$ for $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$;
- $\mathcal{T}(\rho) = U\rho U^\dagger$ for some unitary U.

The following result assures us that there is not much more than that.

Theorem A.0.1 (Stinespring Dilation)**.** *For every quantum channel $\mathcal{T}$ there exists an auxiliary system or* environment *with Hilbert space $\mathcal{H}_E$, a unitary U on $\mathcal{H} \otimes \mathcal{H}_E$ and a vector $|0\rangle \in \mathcal{H}_E$ such that*

$$\mathcal{T}(\rho) = \mathrm{Tr}_E\Big(U(\rho \otimes |0\rangle \langle 0|)U^\dagger\Big) \tag{A.2}$$

We can also define an isometry V as $V|\psi\rangle \coloneqq U(|\psi\rangle \otimes |0\rangle)$, $\forall |\psi\rangle \in \mathcal{H}$, and rewrite the above expression as

$$\mathcal{T}(\rho) = \mathrm{Tr}_E(V\rho V^\dagger). \tag{A.3}$$

Picking a basis $\{|i\rangle\}_{i=1}^N$ of $\mathcal{H}_E$ and defining

$$E_i = \langle i| V \tag{A.4}$$

we get the Kraus or operator-sum representation:

Theorem A.0.2 (Kraus Representation)**.** *Every quantum channel $\mathcal{T}$ can be written as*

$$\mathcal{T}(\rho) = \sum_i E_i \rho E_i^\dagger \tag{A.5}$$

where

$$\sum_i E_i^\dagger E_i = \sum_i V^\dagger |i\rangle \langle i| V = \mathbb{1}|_{\mathcal{H}}. \tag{A.6}$$

Rewritten in terms of the unitary $U_{\mathcal{T}}$ appearing in the Stinespring dilation we have:

$$U_{\mathcal{T}}\colon\ |\psi\rangle \otimes |0\rangle \mapsto \sum_i E_i |\psi\rangle \otimes |i\rangle\,. \tag{A.7}$$

It is immediately clear that the Kraus representation is not unique. If we change basis from $\{|i\rangle\}$ to $\{|\tilde{j}\rangle\}$, where $|i\rangle = \sum_j \Omega_{ij} |\tilde{j}\rangle$ and $\Omega = (\Omega_{ij})_{ij}$ is a unitary on $\mathcal{H}_E$, we see that

$$\begin{aligned} U_{\mathcal{T}}(|\psi\rangle \otimes |0\rangle) &= \sum_i E_i |\psi\rangle \otimes |i\rangle && \text{(A.8)} \\ &= \sum_{ij} E_i |\psi\rangle \otimes \Omega_{ij} |\tilde{j}\rangle && \text{(A.9)} \\ &= \sum_{ij} \Omega_{ij} E_i |\psi\rangle \otimes |\tilde{j}\rangle && \text{(A.10)} \end{aligned}$$

Defining $\tilde{E}_j \coloneqq \sum_i \Omega_{ij} E_i$, we find that

$$\tilde{E}_j^\dagger = \sum_i \overline{\Omega_{ij}} E_i^\dagger = \sum_i (\Omega^\dagger)_{ji} E_i^\dagger \tag{A.11}$$

$$\sum_j \tilde{E}_j^\dagger \tilde{E}_j = \sum_{ijk} E_i^\dagger \underbrace{(\Omega^\dagger)_{ji}\Omega_{kj}}_{\delta_{ik}} E_k = \sum_i E_i^\dagger E_i = \mathbb{1}, \tag{A.12}$$

hence the $\{\tilde{E}_j\}$ are another valid set of Kraus operators for $\mathcal{T}$.

We have a larger freedom actually. We can also think that $\mathcal{H}_E$ has dimension $N+M$, that $\{|i\rangle\}_{i=1}^N$ is not a basis but a subset of a basis and that the Kraus operators corresponding to $i = N+1, \ldots, N+M$ are all zero. Alternatively, we can think of isometrically mapping $\mathcal{H}_E$ of dimension N into some $\mathcal{H}_{\tilde{E}}$ of dimension $N+M$. We call Λ the isometry (corresponding to $\Omega^\dagger$ above, but that was a unitary) such that:

$$|\tilde{j}\rangle = \sum_i \Lambda_{ji} |i\rangle \tag{A.13}$$

$$E_i = \sum_j \Lambda_{ji} \tilde{E}_j \tag{A.14}$$

$$\tilde{E}_j = \sum_i (\Lambda^\dagger)_{ij} E_i \tag{A.15}$$

where $i = 1, \ldots, N$, $j = 1, \ldots, N+M$. Similarly as before, the set $\{\tilde{E}_j\}$ is another valid set of Kraus operators for $\mathcal{T}$. Conveniently, we can always extend an isometry to a unitary by enlarging the dimension of $\mathcal{H}_E$ up to $N+M$, as we have already mentioned. Then we have the following lemma, for which the proof of sufficiency has already been given, whereas the proof of necessity can be found in [13]:

Lemma A.0.1. *Two set of Kraus operators $\{E_i\}, \{\tilde{E}_j\}$ represent the same quantum channel if and only if there exist a unitary Λ' such that*

$$E_i = \sum_j \Lambda'_{ji} \tilde{E}_j \tag{A.16}$$

where the smaller set is padded with zeros.

We have seen that we can always use a larger set of Kraus operators to describe a channel. What about a smaller set?

Theorem A.0.3 (Minimal number of Kraus operators)**.** *Given a channel $\mathcal{T}$ with Kraus representation given by $\{E_i\}_{i=1}^N$, the minimal number of Kraus operators describing the same channel in any other representation is given by the number of linearly independent operators in the set $\{E_i\}_{i=1}^N$.*

Proof. Assume first, by contradiction, that another Kraus representation of $\mathcal{T}$ is given by $\{\tilde{E}_j\}_{j=1}^{\tilde{N}}$, where all the Kraus operators $\tilde{E}_j$ are linearly independent, and that $N < \tilde{N}$, meaning that there are fewer non-zero Kraus operators in the given set $\{E_i\}_{i=1}^N$ than in the set $\{\tilde{E}_j\}_{j=1}^{\tilde{N}}$. Then we add $\tilde{N} - N$ zero operators to the former. But then by lemma A.0.1 there exists a unitary $\Lambda' = (\Lambda'_{ji})_{ji}$ such that $\mathbf{0} = \sum_j \Lambda'_{ji} \tilde{E}_j$ for all $i > N$, which is impossible because the $\{\tilde{E}_j\}$ are linearly independent.
Assume now that the given $\{E_j\}$ are linearly dependent. Then we can find a linear combination with non-zero coefficients $\{\Lambda'_{j1}\}$ such that $E'_1 := \sum_j \Lambda'_{j1} E_j = \mathbf{0}$. We can also renormalise these coefficients in such a way that $\sum_j |\Lambda'_{j1}| = 1$ and extend the vector with entries $\{\Lambda'_{j1}\}$ to an orthonormal basis in order

to get a unitary Λ'. If another one among the $\{E'_i \coloneqq \sum_j \Lambda'_{ji} E_j\}$ is accidentally $\mathbf{0}$ we exclude it from the set. If the remaining operators are still linearly dependent we iterate the procedure, otherwise we have found a minimal representation. □

Note that there are infinitely many equivalent minimal representations because we can still reshuffle the operators using any unitary, which corresponds to changing basis in the "minimal" environment.

Definition A.0.3 (Dimension of a channel)**.** We call dimension of the channel $\mathcal{T}\colon \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H}')$ the minimal number of Kraus operators needed to represent $\mathcal{T}$, and we denote it by $\dim \mathcal{T}$.

From the above discussion we have the trivial bound $\dim \mathcal{T} \leq \dim \mathcal{H} \dim \mathcal{H}'$.

We introduce also the concept of complementary channel, which is obtained by tracing out the system instead of the environment in the Stinespring dilation:

Definition A.0.4 (Complementary channel)**.** Consider a quantum channel $\mathcal{T}\colon \mathcal{B}(\mathcal{H}_A) \to \mathcal{B}(\mathcal{H}_B)$, $\mathcal{T}(\rho_A) = \mathrm{Tr}_E(V\rho_A V^\dagger)$ where $V\colon \mathcal{H}_A \to \mathcal{H}_B \otimes \mathcal{H}_E$ is an isometry. We define the *complementary channel* $\widehat{\mathcal{T}}\colon \mathcal{B}(\mathcal{H}_A) \to \mathcal{B}(\mathcal{H}_E)$ of $\mathcal{T}$ as

$$\widehat{\mathcal{T}}(\rho_A) \coloneqq \mathrm{Tr}_B(V\rho_A V^\dagger) \tag{A.17}$$

Notice that, chosen a basis $\{|i\rangle\}$ of $\mathcal{H}_E$, a Kraus representation of $\mathcal{T}$ is given by the Kraus operators $\{E_i\}$, $E_i \coloneqq \langle i| V$. Then the complementary channel can be expressed w.r.t. the basis $\{|i\rangle\}$ as

$$\widehat{\mathcal{T}}(\rho_A) = \sum_{ij} \mathrm{Tr}_B(E_i \rho_A E_j^\dagger)\, |i\rangle \langle j| . \tag{A.18}$$

Of course the last expression is not unique but it depends on the chosen basis for the environment. Anyway, similarly to the fact that $\mathcal{T}(\rho_A) =: \rho_B$ is the state of the system B after the channel $\mathcal{T}$ was applied, $\widehat{\mathcal{T}}(\rho_A) =: \rho_E$ is the final state of the environment.

A.1 Quantum channels as matrices

Matrices form a vector space $\mathcal{M}_{d,d'}(\mathbb{C})$, which is isomorphic to $\mathbb{C}^{dd'}$. We can make it a Hilbert space defining a scalar product. Here we consider the *Hilbert-Schmidt inner product*:

$$\langle A, B\rangle \coloneqq \mathrm{Tr}(A^\dagger B). \tag{A.19}$$

Notice that we use a comma as notation for this scalar product. Any set of matrices orthonormal w.r.t. this scalar product forms an orthonormal basis (ONB) of $\mathcal{M}_{d,d'}(\mathbb{C})$. Given an ONB $\{F_\alpha\}_{\alpha=1}^{dd'}$ of $\mathcal{M}_{d,d'}(\mathbb{C})$ and an ONB $\{|F_\alpha\rangle\}_{\alpha=1}^{dd'}$ of $\mathbb{C}^{dd'}$ w.r.t. the standard scalar product $\langle\cdot|\cdot\rangle$ (we use a bar for this one), we can define the isomorphism $\sigma\colon \mathcal{M}_{d,d'}(\mathbb{C}) \to \mathbb{C}^{dd'}$, mentioned above, as $\sigma(F_\alpha) = |F_\alpha\rangle\ \forall \alpha$.

Consider now square matrices $\mathcal{M}_d(\mathbb{C})$. The space of linear maps $T\colon \mathcal{M}_d(\mathbb{C}) \to \mathcal{M}_{d'}(\mathbb{C})$ is isomorphic to $\mathcal{M}_{d^2,d'^2}(\mathbb{C})$. The standard way to get the matrix $\hat{T}$ corresponding to a linear map T is to write down the action of T on a basis $\{G_\beta\}_{\beta=1}^{d^2}$ of $\mathcal{M}_d(\mathbb{C})$ in terms of a basis $\{F_\alpha\}_{\alpha=1}^{d'^2}$ of $\mathcal{M}_{d'}(\mathbb{C})$, i.e. the matrix elements $\hat{T}_{\alpha\beta} = \langle F_\alpha|\hat{T}|G_\beta\rangle$ of $\hat{T}$ are given by

$$\hat{T}_{\alpha\beta} = \langle F_\alpha, T(G_\beta)\rangle \tag{A.20}$$

$$= \mathrm{Tr}(F_\alpha^\dagger T(G_\beta)). \tag{A.21}$$

Then composition of maps corresponds to matrix multiplication.

If T is *hermiticity-preserving*, i.e. $T(X^\dagger) = T(X)^\dagger$, then for the dual map T^* we have that $\widehat{T^*} = \hat{T}^\dagger$, in general, and $\widehat{T^*} = \hat{T}^T$ picking hermitian ONBs (the basis elements are hermitian matrices), which can always be done.

From now on consider $d = d'$, i.e. take T mapping between the same input and output spaces (or, more generally, between input and output spaces of the same dimension). In this way there is the possibility for T to be diagonalizable (it would be more appropriate to say *non-defective* in the second case). In general this is not true because $\hat{T}$ does not necessarily have to be normal even if it is hermiticity-preserving. However, if this latter hypothesis holds, which is true for positive maps and so for quantum channels in particular, then we have that T and its dual have the same spectrum because $\widehat{T^*} = \hat{T}^T$ picking hermitian ONBs, and it is clear that a matrix and its transpose have the same spectrum.

Notice that in this framework we have the simple connection between the (left) "eigenvectors" of T and the usual notion of (left) eigenvectors of a matrix, which is given by

$$T(X) = \lambda X \iff \hat{T}\,|X\rangle = \lambda\,|X\rangle\,. \tag{A.22}$$

Similarly the right eigenvectors Y are such that

$$T^*(Y) = \mu\, Y \iff \langle Y|\,\hat{T} = \bar{\mu}\,\langle Y|\,. \tag{A.23}$$

Moreover we have that $T(X) = \lambda X \iff T(X)^\dagger = \bar{\lambda}X^\dagger = T(X^\dagger)$ using hermiticity-preservation, which implies that the eigenvalues are always real or come in conjugate pairs, and in the former case there exists at least a hermitian eigenvector X (using linearity we get that $X + X^\dagger$ is an eigenvector with the same eigenvalue of X).

Calling $\rho(T)$ the spectral radius of T, for a positive T we have that $\rho(T) \leq ||T(\mathbb{1})||_\infty$ (see [13] for a proof) and if T is also unital or trace preserving we have $\rho(T) = 1$, because if it is unital then $\mathbb{1}$ is obviously an eigenvector with eigenvalue 1, and if it is trace-preserving then the dual is unital but they have the same spectrum. Hence, quantum channels in particular have spectral radius 1 and $\lambda = 1$ is always an eigenvalue. Note that it may have multiplicity greater than one and there may be complex eigenvalues λ with

$|\lambda| = 1$. However, they always have algebraic multiplicity equal to geometric multiplicity, i.e. trivial Jordan blocks (*peripheral spectrum theorem*, see [13] for a discussion).

Diagonalizability

If all the eigenvalues of $\hat{T}$ have trivial Jordan blocks, then we can write it as

$$\hat{T} = \sum_{k \geq 0} \lambda_k \left|L_k\right\rangle \left\langle R_k\right| \tag{A.24}$$

where now we denote by $\{R_k\}$ and $\{L_k\}$ the right and left eigenvectors respectively, which satisfy

$$\langle R_k | L_j \rangle = \delta_{kj}. \tag{A.25}$$

This is called the *bi-orthonormality condition*. It can also be rewritten as $\mathrm{Tr}(R_k^\dagger L_j) = \delta_{kj}$. Notice that it does not mean that the two sets are orthonormal among themselves (w.r.t. the Hilbert Schmidt inner product).

Then for every operator A:

$$\hat{T}\left|A\right\rangle = \sum_k \lambda_k \left|L_k\right\rangle \left\langle R_k | A\right\rangle = \sum_k \lambda_k \left\langle R_k, A\right\rangle \left|L_k\right\rangle = \sum_k \lambda_k \,\mathrm{Tr}(R_k^\dagger A)\left|L_k\right\rangle \tag{A.26}$$

which means that

$$T(A) = \sum_k \lambda_k \,\mathrm{Tr}(R_k^\dagger A) L_k. \tag{A.27}$$

If T is unital, then $\mathbb{1}$ is a left eigenvector (say that $L_0 = \mathbb{1}$) and it follows that $\mathrm{Tr}(R_k^\dagger) = \delta_{k,0} \implies \mathrm{Tr}(R_k) = \delta_{k,0}$. If instead T is trace-preserving, then $\mathrm{Tr}(L_k) = \mathrm{Tr}(T(L_k)) = \lambda_k \,\mathrm{Tr}(L_k)$, which means that $\mathrm{Tr}(L_k) = 0$ for every k such that $\lambda_k \neq 1$.

Stationary state

Consider now T to be a (completely) positive unital map so that T^* is a (C)PTP map, i.e. a quantum channel (here we do not really need *complete* positivity). Suppose that T is diagonalizable and in particular that it has only one eigenvalue of modulus 1, i.e. the one corresponding to the identity, $T(\mathbb{1}) = \mathbb{1}$. Then

$$T^*(A) = \mathrm{Tr}(A)\, R_0 + \sum_{k \neq 0} \bar{\lambda}_k \,\mathrm{Tr}(L_k^\dagger A)\, R_k \tag{A.28}$$

where $|\lambda_k| < 1$ for $k \neq 0$. Applying multiple times T^* to A we get:

$$\lim_n (T^*)^n(A) = \lim_n \Big(\mathrm{Tr}(A)\, R_0 + \sum_{k \neq 0} \bar{\lambda}_k^n \,\mathrm{Tr}(L_k^\dagger A)\, R_k\Big) \tag{A.29}$$

$$= \mathrm{Tr}(A)\, R_0. \tag{A.30}$$

If A is a density matrix it follows from positivity and trace-preservation that also R_0 is a density matrix, to which we refer as the (unique) *stationary state* of the channel, such that $T^*(R_0) = R_0$. In particular any state A converges to R_0 for $n \to \infty$ iterations of the channel T^*.

Non-diagonalizable channels

We can ask what happens if we have a channel T^* which is not diagonalizable but the characteristic polynomial of $\widehat{T^*}$ has only one root of modulus 1: is it still true, as in (A.30), that there is only one stationary state R_0 such that $\lim_n (T^*)^n(\rho) = R_0$ for any density matrix ρ? We will prove that this actually the case (see [30] for more details).

Recall that any matrix can be put into Jordan form through some invertible transformation $\hat{S}$. Hence, we can write

$$\widehat{T^*} = \hat{S}\hat{J}\hat{S}^{-1} \tag{A.31}$$

where

$$\hat{J} = \oplus_k \hat{J}_{d_k}(\lambda_k) \tag{A.32}$$

is the Jordan form and $\hat{J}_{d_k}(\lambda_k)$ is the Jordan block of dimension d_k associated with the eigenvalue λ_k (there may be in general more than one block associated with an eigenvalue if it is degenerate). Recall also that

$$\hat{J}_{d_k}(\lambda_k) = \lambda_k \mathbb{1}_{d_k} + \hat{N}_{d_k} \tag{A.33}$$

where $\hat{N}_{d_k}$ is either the 1×1 null matrix $\mathbf{0}_1$, for $d_k = 1$, or it is the matrix with matrix elements 1 right above the diagonal and 0 otherwise, for $d_k \geq 2$, hence it is nilpotent for any $p \geq d_k$, i.e. $(\hat{N}_{d_k})^p = \mathbf{0}_{d_k}$. It follows that

$$(\widehat{T^*})^m = \hat{S}\hat{J}^m\hat{S}^{-1} = \hat{S}\Big(\oplus_k [\hat{J}_{d_k}(\lambda_k)]^m\Big)\hat{S}^{-1} \tag{A.34}$$

$$[\hat{J}_{d_k}(\lambda_k)]^m := \begin{cases} (\hat{N}_{d_k})^m, & \lambda_k = 0 \\ \sum_{q=0}^{d_k-1} \binom{m}{q} \lambda_k^{m-q} (\hat{N}_{d_k})^q, & \lambda_k \neq 0 \end{cases} \tag{A.35}$$

which implies that for $|\lambda_k| < 1$

$$\lim_m [\hat{J}_{d_k}(\lambda_k)]^m = \mathbf{0}_{d_k}. \tag{A.36}$$

Therefore

$$\lim_m (\widehat{T^*})^m |\rho\rangle = \hat{S}\Big(\hat{J}_1(1) \oplus_{k\neq 0} \mathbf{0}_{d_k}\Big)\hat{S}^{-1} |\rho\rangle = |R_0\rangle\,. \tag{A.37}$$

Appendix B

Distance Measures

Here we review basic definitions and results about the trace distance, the fidelity and the entanglement fidelity.

Definition B.0.1 (Trace distance)**.** Given two density matrices ρ and σ, their *trace distance* is given by $\|\rho - \sigma\|_1$, which corresponds to the sum of the eigenvalues of $|\rho - \sigma|$.

Definition B.0.2 (Fidelity)**.** Given two density matrices ρ and σ, their *fidelity* is given by:

$$F(\rho, \sigma) = \left(\mathrm{Tr}\sqrt{\sqrt{\sigma}\rho\sqrt{\sigma}}\right)^2. \tag{B.1}$$

Lemma B.0.1. *The fidelity has the following properties (see chapter 2 of [8] for a discussion):*

1. $0 \leq F \leq 1$*;*
2. *symmetry:* $F(\rho, \sigma) = F(\sigma, \rho)$*;*
3. *for* $\sigma = |\psi\rangle\langle\psi|$ *pure:* $F(\rho, |\psi\rangle\langle\psi|) = \langle\psi|\rho|\psi\rangle \equiv F(\rho, |\psi\rangle)$*;*
4. *if also* $\rho = |\varphi\rangle\langle\varphi|$ *is pure, then* F *reduces to the overlap:* $F(|\varphi\rangle\langle\varphi|, |\psi\rangle) = |\langle\varphi|\psi\rangle|^2$*;*
5. *for every quantum channel* $\mathcal{T}$*:* $F(\mathcal{T}(\rho), \mathcal{T}(\sigma)) \geq F(\rho, \sigma)$*, i.e.* $\mathcal{T}$ *makes the states less distinguishable;*
6. $F(\rho_{AB}, \sigma_{AB}) \leq F(\rho_A, \sigma_A)$*, i.e. tracing out subsystems makes states less distinguishable;*
7. *Fuchs-van der Graaf inequality:*

$$1 - \sqrt{F(\rho, \sigma)} \leq \frac{1}{2}\|\rho - \sigma\|_1 \leq \sqrt{1 - F(\rho, \sigma)}. \tag{B.2}$$

The entanglement fidelity that we are going to define is a measure of closeness between density matrices which takes into account how well the entanglement between the system A and a purifying reference system R is preserved

under the action of a channel and not only how well the reduced state is preserved.

Definition B.0.3 (Schumacher's entanglement fidelity [31])**.** Given a quantum channel $\mathcal{T}\colon \mathcal{B}(\mathcal{H}_A) \to \mathcal{B}(\mathcal{H}_{A'})$, for a state ρ_A with purification $|\psi\rangle_{AR}$ we define *Schumacher's entanglement fidelity* as

$$F_\rho(\mathcal{T}) \coloneqq F\Big(\mathcal{T} \otimes \mathrm{id}(|\psi\rangle\langle\psi|), \mathrm{id} \otimes \mathrm{id}(|\psi\rangle\langle\psi|)\Big) \tag{B.3}$$

$$= F\Big(\mathcal{T} \otimes \mathrm{id}(|\psi\rangle\langle\psi|), |\psi\rangle\Big) = \langle\psi|\mathcal{T} \otimes \mathrm{id}(|\psi\rangle\langle\psi|)|\psi\rangle\,. \tag{B.4}$$

Lemma B.0.2 ([31])**.** *Schumacher's entanglement fidelity is independent of the chosen purification, which can be seen from the equivalent expression given by*

$$F_\rho(\mathcal{T}) = \sum_i |\mathrm{Tr}(\rho E_i)|^2 \tag{B.5}$$

where the $\{E_i\}$ are the Kraus operators of $\mathcal{T}$ in a given Kraus representation. Moreover, the above expression does not depend on which Kraus representation of $\mathcal{T}$ is chosen.

It is useful to extend the definition of Schumacher's entanglement fidelity to the case in which the second channel in (B.3) is a more general one:

Definition B.0.4 ((Generalized) Entanglement fidelity)**.** Given quantum channels $\mathcal{T}_1, \mathcal{T}_2$ acting on $\mathcal{B}(\mathcal{H}_\mathcal{A})$, for a state ρ_A with purification $|\psi\rangle_{AR}$ we define the *entanglement fidelity* as

$$F_\rho(\mathcal{T}_1, \mathcal{T}_2) \coloneqq F\Big(\mathcal{T}_1 \otimes \mathrm{id}(|\psi\rangle\langle\psi|), \mathcal{T}_2 \otimes \mathrm{id}(|\psi\rangle\langle\psi|)\Big). \tag{B.6}$$

Lemma B.0.3. *The entanglement fidelity is independent of the chosen purification. Moreover, w.l.o.g., picking $\mathcal{T}_1, \mathcal{T}_2$ represented by the same number of Kraus operators, we then get*

$$F_\rho(\mathcal{T}_1, \mathcal{T}_2) = \max_U \left|\sum_{kl} \mathrm{Tr}(F_l^\dagger E_k \rho)\,\langle l|U|k\rangle\right|^2 \tag{B.7}$$

where U is a unitary and $\{E_k\}, \{F_l\}$ are the Kraus operators of $\mathcal{T}_1, \mathcal{T}_2$ respectively. If the number of Kraus operators do not match, we can always change the representation for the channel with fewer Kraus operators, or we can pick U as an isometry mapping the smaller environment to the bigger environment (which can anyway be extended to a unitary).

Proof. By Uhlmann's theorem [32] the fidelity between two states can be expressed in terms of the maximum overlap between their purifications. Given $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$ and a purification $|\psi\rangle = \sum_i \sqrt{p_i}\,|\psi_i\rangle \otimes |\alpha_i\rangle$ where $\{|\alpha_i\rangle\}$ is a basis of a purifying reference system, any other purification is given by

$$|\psi(V)\rangle \coloneqq (\mathbb{1} \otimes V)\,|\psi\rangle = \sum_i \sqrt{p_i}\,|\psi_i\rangle \otimes V\,|\alpha_i\rangle \tag{B.8}$$

where V is a unitary. We have to prove that $F_\rho(\mathcal{T}_1, \mathcal{T}_2)$ is independent of V. We have that $\mathcal{T}_1(|\psi\rangle\langle\psi|) = \sum_{ij} \sqrt{p_i p_j} E_k |\psi_i\rangle\langle\psi_j| E_k^\dagger \otimes V |\alpha_i\rangle\langle\alpha_j| V^\dagger$ and similarly for $\mathcal{T}_2$. Purifications of these two states are given by

$$|\varphi_1\rangle = \sum_{ik} \sqrt{p_i} E_k |\psi_i\rangle \otimes V |\alpha_i\rangle \otimes |k\rangle \tag{B.9}$$

$$|\varphi_2(U)\rangle = \sum_{ik} \sqrt{p_i} E_k |\psi_i\rangle \otimes V |\alpha_i\rangle \otimes U |k\rangle \tag{B.10}$$

where U is a unitary (w.l.o.g. we can keep one of the two purifications fixed). By Uhlmann's theorem

$$F_\rho(\mathcal{T}_1, \mathcal{T}_2) = \max_U |\langle\varphi_1|\varphi_2(U)\rangle|^2 \tag{B.11}$$

$$= \max_U \left| \sum_{ijkl} \sqrt{p_i p_j} \langle\psi_j|F_l^\dagger E_k|\psi_i\rangle \langle\alpha_j|V^\dagger V|\alpha_i\rangle \langle l|U|k\rangle \right|^2 \tag{B.12}$$

$$= \max_U \left| \sum_{ikl} p_i \langle\psi_i|F_l^\dagger E_k|\psi_i\rangle \langle l|U|k\rangle \right|^2 \tag{B.13}$$

$$= \max_U \left| \sum_{kl} \mathrm{Tr}(F_l^\dagger E_k \rho) \langle l|U|k\rangle \right|^2 \tag{B.14}$$

where we have used that $V^\dagger V = \mathbb{1}$ and the orthonormality of the $\{|\alpha_i\rangle\}$. □

Notice that when $\mathcal{T}_2 = \mathrm{id}$, then

$$F_\rho(\mathcal{T}_1, \mathrm{id}) = F_\rho(\mathcal{T}_1) = \max_U \left| \sum_k \mathrm{Tr}(E_k \rho) \langle 0|U|k\rangle \right|^2 . \tag{B.15}$$

Using the Cauchy-Schwartz inequality, we see that $F_\rho(\mathcal{T}_1) \leq \sum_i |\mathrm{Tr}(\rho E_i)|^2$ and that this upper bound can be achieved picking U such that $\langle 0|U|k\rangle = \overline{\mathrm{Tr}(E_k\rho)}/\sqrt{\sum_i |\mathrm{Tr}(\rho E_i)|^2}$, so that we recover equation (B.5).

We can also give general bounds on (B.7). Defining a matrix $\Gamma^{(\rho)}$ with matrix elements $\Gamma^{(\rho)}_{kl} = \mathrm{Tr}(F_l^\dagger E_k \rho)$, we can rewrite (B.7) as an Hilbert-Schmidt inner product

$$F_\rho(\mathcal{T}_1, \mathcal{T}_2) = \max_U \left| \mathrm{Tr}(\Gamma^{(\rho)} U) \right|^2 = \max_U \left| \langle U^\dagger, \Gamma^{(\rho)} \rangle \right|^2 \tag{B.16}$$

which then allows us to use the Hölder inequality $|\langle A, B\rangle| \leq \|A\|_p \|B\|_{p^*}$ where $1/p + 1/p^* = 1$ and $\|\cdot\|_p$ is the Schatten p-norm. To get upper bounds which do not involve a maximization over unitaries, for U we can pick the operator (infinity) norm, the trace norm or the Hilbert-Schmidt norm. Respectively:

$$F_\rho(\mathcal{T}_1, \mathcal{T}_2) \leq \left\| \Gamma^{(\rho)} \right\|_1^2 \tag{B.17}$$

$$F_\rho(\mathcal{T}_1, \mathcal{T}_2) \leq N^2 \left\| \Gamma^{(\rho)} \right\|_\infty^2 \tag{B.18}$$

$$F_\rho(\mathcal{T}_1, \mathcal{T}_2) \leq N^2 \left\| \Gamma^{(\rho)} \right\|_2^2 = N^2 \sum_{kl} \left| \mathrm{Tr}(F_l^\dagger E_k \rho) \right|^2 . \tag{B.19}$$

Bibliography

[1] Cédric Bény and Ognyan Oreshkov. General conditions for approximate quantum error correction and near-optimal recovery channels. *Physical review letters*, 104(12):120501, 2010.

[2] A. Yu. Kitaev. Fault tolerant quantum computation by anyons. *Annals Phys.*, 303:2–30, 2003.

[3] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505, 2002.

[4] Guifré Vidal. Class of quantum many-body states that can be efficiently simulated. *Physical review letters*, 101(11):110501, 2008.

[5] Guifre Vidal. Entanglement renormalization. *Physical review letters*, 99(22):220405, 2007.

[6] Miguel Aguado and Guifre Vidal. Entanglement renormalization and topological order. *Physical review letters*, 100(7):070404, 2008.

[7] Isaac H Kim and Michael J Kastoryano. Entanglement renormalization, quantum error correction, and bulk causality. *Journal of High Energy Physics*, 2017(4):40, 2017.

[8] John Preskill. http://www.theory.caltech.edu/people/preskill/ph229/.

[9] Daniel Lidar and Todd Brun. *Quantum Error Correction*. Cambridge University Press, September 2013.

[10] John Watrous. Theory of quantum information, 2011.

[11] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. December 2000.

[12] P. W. Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *eprint arXiv:quant-ph/9508027*, August 1995.

[13] Michael M. Wolf. www-m5.ma.tum.de/foswiki/pub/m5/allgemeines /michaelwolf/qchannellecture.pdf, 2012.

[14] Emanuel Knill and Raymond Laflamme. Theory of quantum error-correcting codes. *Physical Review A*, 55(2):900, 1997.

[15] Claude Crépeau, Daniel Gottesman, and Adam Smith. Approximate quantum error-correcting codes and secret sharing schemes. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 285–301. Springer, Berlin, Heidelberg, 2005.

[16] Yi-Chan Lee, Courtney Brell, and Steven T Flammia. Topological quantum error correction in the kitaev honeycomb model. *arXiv preprint arXiv:1705.01563*, 2017.

[17] Steven T Flammia, Jeongwan Haah, Michael J Kastoryano, and Isaac H Kim. Limits on the storage of quantum information in a volume of space. *arXiv preprint arXiv:1610.06169*, 2016.

[18] Sergey Bravyi, David Poulin, and Barbara Terhal. Tradeoffs for reliable quantum information storage in 2d systems. *Physical review letters*, 104(5):050503, 2010.

[19] R. Orús. A practical introduction to tensor networks: Matrix product states and projected entangled pair states. *Annals of Physics*, 349:117–158, October 2014.

[20] David Perez-Garcia, Frank Verstraete, Michael M Wolf, and J Ignacio Cirac. Matrix product state representations. *arXiv preprint quant-ph/0608197*, 2006.

[21] Ulrich Schollwöck. The density-matrix renormalization group in the age of matrix product states. *Annals of Physics*, 326(1):96–192, 2011.

[22] F. Verstraete and J. I. Cirac. Renormalization algorithms for Quantum-Many Body Systems in two and higher dimensions. *eprint arXiv:cond-mat/0407066*, July 2004.

[23] Steven R. White. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.*, 69:2863–2866, Nov 1992.

[24] Ulrich Schollwöck. The density-matrix renormalization group. *Reviews of modern physics*, 77(1):259, 2005.

[25] G. Evenbly and G. Vidal. Algorithms for entanglement renormalization. 79(14):144108, April 2009.

[26] Daniel Gottesman. The Heisenberg representation of quantum computers. In *Group theoretical methods in physics. Proceedings, 22nd International Colloquium, Group22, ICGTMP'98, Hobart, Australia, July 13-17, 1998*, pages 32–43, 1998.

[27] Sergey Bravyi and Barbara Terhal. A no-go theorem for a two-dimensional self-correcting quantum memory based on stabilizer codes. *New Journal of Physics*, 11(4):043029, 2009.

[28] Jeongwan Haah and John Preskill. Logical operator tradeoff for local quantum codes. *Physical Review A*, 86(3):032308, 2012.

[29] Barbara M Terhal. Quantum error correction for quantum memories. *Reviews of Modern Physics*, 87(2):307, 2015.

[30] V Giovannetti, S Montangero, Matteo Rizzi, and Rosario Fazio. Homogeneous multiscale-entanglement-renormalization-ansatz states: An information theoretical analysis. *Physical Review A*, 79(5):052314, 2009.

[31] Benjamin Schumacher. Sending entanglement through noisy quantum channels. volume 54, pages 2614–2628. quant-ph/9604023, October 1996.

[32] A. Uhlmann. The "transition probability" in the state space of a *-algebra. *Reports on Mathematical Physics*, 9(2):273 – 279, 1976.