Alma Mater Studiorum · Università di Bologna

Scuola di Scienze
Dipartimento di Fisica e Astronomia
Corso di Laurea in Fisica

Quantum Error Correction and the Toric Code

Relatore:
Dott. Davide Vodola

Presentata da:
Andrea Gaspari

Anno Accademico 2019/2020

Abstract

Quantum error correction is the main subject of this thesis. After a general introduction of the fundamentals of quantum mechanics and quantum computing, the problem of quantum error correction is presented and further analysed using two different approaches, one, more practical, based on quantum circuits and one, purely theoretical, based on *stabilizer formalism*. Examples of the principal quantum codes are progressively supplied to help the comprehension. To conclude the attention is drawn to the *Toric code* which represents one of the most promising platforms to store quantum information.

Sommario

La correzione degli errori quantistici è il principale argomento affrontato in questa tesi. Dopo una generale introduzione dei fondamentali di meccanica quantistica e di computazione quantistica, il problema viene delineato e successivamente analizzato seguendo due diversi approcci, uno, più pratico, basato sull'utilizzo di circuiti quantistici e uno, puramente teorico, basato sull'utilizzo dello *stabilizer formalism*. Esempi dei principali codici quantistici vengono gradualmente proposti per aiutare la comprensione. In conclusione l'attenzione viene incentrata sul *Toric code*, uno dei codici più promettenti per la realizzazione di memorie quantistiche.

Contents

Introduction

Moving to the realm of quantum computation is an extremely topical challenge nowadays: quantum effects that rule the physics of microscopic world offer many possibilities for information processing but also have their drawbacks. The probabilistic nature and the fragility of quantum systems are two of the main problems that make quantum processors remarkably difficult to be realised. This is why theoretical studies, taking inspiration from the basic concepts and algorithms of coding theory and classical error correction, developed the brand new field of Quantum Error Correction (QEC) regarding the structure, the properties and the operations necessary in a code to protect quantum information. The subsequent advent of fault-tolerant computation completed the frame and convinced the science community of the fact that quantum computing was possible.

In this thesis the principal aspects of QEC are discussed in order to provide a highly accessible introduction to the subject. The attention is also brought to some of the many codes developed in the early years of researches, in particular the aim is pointed to those which allows to understand in the most intuitive way the basic concepts behind quantum error correcting protocols but are also able to give a perception of their potential. When possible visual examples of quantum circuits and error scenarios are provided to ensure an *active* reading.

Regarding the structure of this thesis, it is composed of three chapters. In the first, basic elements of quantum mechanics, standard components of quantum computing and elementary models of quantum errors are discussed. In the second one the problem of achieving quantum error correction is presented, supported with many analogies and differences with the classical version of problem. In this part an essential formalism to describe rigorously QEC is also introduced and two basic error code are analysed. In the third chapter the *Toric code*, i.e. one of the most promising codes that can be employed as a quantum memory, is described. In particular the description of the error correction on the Toric code finds a remarkable analogy with a classical statistical model whose ordered and disordered phases can be mapped to the regions of the Toric code where error correction succeeds or fails.

Chapter1

Quantum Mechanics fundamentals for QEC

In this chapter the formal structure, based on Dirac's notation, used to describe quantum systems is introduced. In particular, the focus is on 2-state quantum systems which are used as *quantum bit*, the fundamental unit of quantum information. Many examples of 2-state quantum systems can be provided but the most common are the ones based on atomic spin states or on electronic states of an ion. Subsequently a brief introduction on quantum gates follows while some basic error models and how those affect quantum algorithms are listed and reviewed at the end.

1.1 The formal structure of the qubits

In quantum mechanics every independent state i of a quantum system is associated with a normalized ket $|\psi_i\rangle$, defined up to a phase factor, in the space of states $\mathcal{H}$, which is a Hilbert's complex linear vector space. It is then possible to define the counterpart of the ket vector with the operation of *conjunction* that returns the corresponding bra vector $\langle\psi_i|$. The property that characterises a vector space as an Hilbert's space is the definition of an inner product on the space itself. In this case, defined an orthonormal basis $\{|\psi_i\rangle\}$ and taken two generic states $|\psi\rangle = \sum_i c_i |\psi_i\rangle$ and $|\psi'\rangle = \sum_i d_i |\psi_i\rangle$ it comes as:

$$\left\langle\psi\middle|\psi'\right\rangle = \sum_i c_i^* d_i \tag{1.1.1}$$

which is Hermitian $\langle\psi|\psi'\rangle^* = \langle\psi'|\psi\rangle$ and non-negative $\langle\psi|\psi\rangle \geq 0$.
As anticipated, a qubit is a 2-state quantum system that lives in a $\mathcal{H}^2$ space, but, unlike classical bit, can exist as superposition of its two basis states, usually denoted as $|0\rangle$ and $|1\rangle$ which, without loss of generality, are assumed to form an orthonormal base of $\mathcal{H}^2$. Orthogonality and normalization conditions are synthetized in the equation:

$$\langle i|j\rangle = \delta_{i,j} \quad \forall i,j = 0,1. \tag{1.1.2}$$

An arbitrary state of an individual qubit $|\psi\rangle$ can be then represented as:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle \tag{1.1.3}$$

and, for the sake of simplicity, it is assumed that $|\psi\rangle$ satisfies the normalization condition:

$$\langle\psi|\psi\rangle = 1 \tag{1.1.4}$$

which implies that $|\alpha|^2 + |\beta|^2 = 1$, hence the total probability of measuring the qubit in one of the two states is unitary.
Although superposition already allows unexpected scenarios in which it is possible to operate at the very same time on both states of a qubit, most advantages of quantum computing over classical one come from *entangled states*. To describe clearly what is an entangled state it is necessary first to introduce compound system. Compound system can be intuitively imagined in our case as multiple-qubit systems and are represented by tensor-product Hilbert spaces; for example, a system composed of two qubits has a state $|\psi\rangle$ belonging to the tensor product of the Hilbert spaces of the two individual qubits, which implies that $|\psi\rangle \in \mathcal{H}^{2\otimes 2}$. However, in this and in even larger $\mathcal{H}^{2^{\otimes n}}$ spaces, there is a fundamental and physically crucial difference between two kind of states: the *"factorizable states"*, which can be expressed in the form:

$$|\psi\rangle = \prod_{\substack{\otimes \\ i=1}}^{n} |\psi_i\rangle = (\alpha_1\,|0\rangle_1 + \beta_1\,|1\rangle_1) \otimes [...] \otimes (\alpha_n\,|0\rangle_n + \beta_n\,|1\rangle_n) \tag{1.1.5}$$

and the entangled states which are *"unfactorizable"*. Considering again the 2-qubit system, a factorizable state could be, for instance:

$$|\psi_f\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) = \frac{1}{2}\Big[(|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle)\Big] \tag{1.1.6}$$

while an example of entangled state could be:

$$|\psi_e\rangle = \alpha\,|00\rangle + \beta\,|11\rangle\,. \tag{1.1.7}$$

1.2 Generalities of quantum operators and quantum gates

Quantum gates are the fundamental unit for quantum computing and quantum circuits and can be represented as operators acting on the states of one or more qubits, depending on the nature of the gate. As stated previously, a qubit can be represented with a 2-dimensional ket, thus, the operator algebra acting on those kets is 4-dimensional. To preserve the normalization condition of a state $|\psi\rangle$, quantum operators must preserve the inner product, which means, that, given an operator $\hat{A}$ that transform $|\psi\rangle \rightarrow |\psi'\rangle$, it follows:

$$\langle\psi'|\psi'\rangle = \langle\psi|\,\hat{A}^{\dagger}\hat{A}\,|\psi\rangle = 1 \tag{1.2.1}$$

$$\Rightarrow \hat{A}^{\dagger}\hat{A} = \hat{\mathbb{1}}. \tag{1.2.2}$$

Operators satisfying Eq.(1.2.2) are defined *unitary*.
This imply that all quantum gates are also *reversible*, in fact, equation (1.2.2) prove that $\hat{A^{-1}}$ exists and it is equal to $\hat{A}^{\dagger}$, thus:

$$\hat{A}^{\dagger}\,|\psi'\rangle = \hat{A}^{\dagger}\hat{A}\,|\psi\rangle = \hat{\mathbb{1}}\,|\psi\rangle = |\psi\rangle \tag{1.2.3}$$

A handy basis $\mathfrak{B}$ of 2×2 complex matrices to represent qubit operators is given by the set of the identity matrix $\mathbb{1}$ and the *Pauli Matrices* $\{\sigma_x, \sigma_y, \sigma_z\}$, defined as follows:

$$\mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{1.2.4}$$

All those matrices are unitary but also hermitian, which implies that:

$$\mathbb{1} = \mathbb{1}^{-1} \; and \; \sigma_i = \sigma_i^{-1} \quad for \; i = \{x, y, z\}, \tag{1.2.5}$$

moreover, the the Pauli matrices satisfy the following relations:

$$\sigma_i\sigma_j = \delta_{ij}\mathbb{1} + i\epsilon_{ijk}\sigma_k \tag{1.2.6}$$

$$[\sigma_i, \sigma_j] = 2i\epsilon_{ijk}\sigma_k \tag{1.2.7}$$

$$\{\sigma_i, \sigma_j\} = 2\delta_{ij}\mathbb{1} \tag{1.2.8}$$

in which ϵ_{ijk} stands for the *Levi-Civita symbol*, $[\cdot,\cdot]$ for the commutator and $\{\cdot,\cdot\}$ for the anticommutator.
Consequently, identified each matrix with the corresponding operator, hence $\mathbb{1} \rightarrow \hat{\mathbb{1}}$ and $\sigma_i \rightarrow \hat{\sigma_i}$ $\forall i = x, y, z$, it is possible to build a base $\hat{\mathfrak{B}} = \{\hat{\mathbb{1}}, \hat{\sigma_x}, \hat{\sigma_y}, \hat{\sigma_z}\}$ of the operators acting on a qubit, which, as expected, is 4-dimensional. With slight abuse of notation, Pauli operators are conventionally identified as $X \equiv \hat{\sigma_x}$, $Y \equiv -\hat{\sigma_y}$ *and* $Z \equiv \hat{\sigma_z}$. The properties of *unitarity* and *hermitianity* are inherited from the elements of the base and relations (1.2.6), (1.2.7) and (1.2.8) hold. It follows that single-qubit operators $\hat{A}$ can be expressed, up to a normalization constant, as:

$$\hat{A} = \alpha\hat{\mathbb{1}} + \beta\hat{\sigma_x} + \gamma\hat{\sigma_y} + \delta\hat{\sigma_z} \qquad with \; \alpha, \beta, \gamma, \delta \in \mathbb{C}, \tag{1.2.9}$$

while multi-qubit operators acting on compound system consist in a tensor product of single-qubit operators:

$$\hat{A} = \hat{A}_1 \otimes \hat{A}_2 \otimes [\ldots] \otimes \hat{A}_n. \tag{1.2.10}$$

where $\hat{A}_k$ indicates an operator acting on the k-th qubit. Finally, using the results obtained, in particular the matrix representations (1.2.4) and equations (1.2.9), (1.2.10); it is possible to proceed with clarity to describe the quantum gates of greater interest.

Quantum gates are typically theorized starting from classical ones and, as those latter, can be distinguished depending on how many input they got: there exist *unary* gates, which operate only on one qubit, *binary* gates which instead act on two qubits, and so on. Due to the fact that quantum gates are reversible by definition, the numbers of qubits in input must be the same of the qubits in output, otherwise the gates are not *injectives*, hence not *bijectives* nor reversibles. An immediate consequence of this property is, for instance, that the classical AND gate, largely used in classical computing, could not have an exact quantum analogue. Although almost any classical gate has an adapted analogue in quantum information, for the purposes of this thesis, only few gates are reviewed in detail, which are the *NOT* gate, the R_π gate, the *Hadamard transform*, an arbitrary *Phase rotation* gate, and the *CNOT* gate. For each, the *Truth Table* and the corresponding matrix are provided to explicate their functional operation, in addition to the quantum circuit symbol.

The quantum **NOT gate**, which has a double representation $-\boxed{X}-$ or $-\oplus-$, is one of the most intuitive gate, it operates a flip on the states of a single qubit and has its analogous in the classical NOT gate. By looking at his matrix representation it is trivial to note it corresponds to the Pauli operator $\hat{\sigma_x}$, thus, is also named *X-gate*.

Input	Output
$\|0\rangle$	$\|1\rangle$
$\|1\rangle$	$\|0\rangle$

Tab. 1.1: *X-gate* truth table.

$$\hat{\sigma_x} \equiv X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Fig. 1.1: *X-gate* matrix representation.

For the purpose of clarity, given a qubit in a state $|\psi\rangle$, *X-gate* operates as follows:

$$X|\psi\rangle = X(\alpha|0\rangle + \beta|1\rangle) = \alpha(X|0\rangle) + \beta(X|1\rangle) = \alpha|1\rangle + \beta|0\rangle = |\psi'\rangle \tag{1.2.11}$$

Because of its similarity with the previous one, the $\mathbf{R}_\pi$ **gate** $-\boxed{Z}-$ is the following introduced. Despite the fact it may not seems immediate because are associated to different matrices and got non-identical truth tables, *X-gate* and R_π*-gate* really perform the very same operation, just in *different orthonormal basis.* Once again, considering the matrix representation, it could be easily noted that this gate actually corresponds to the Pauli operator $\hat{\sigma_z}$, hence is also identified as *Z-gate*.

Input	**Output**
$\|0\rangle$	$\|0\rangle$
$\|1\rangle$	$-\|1\rangle$

Tab. 1.2: *Z-gate* truth table.

$$\hat{\sigma_z} \equiv Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Fig. 1.2: *Z-gate* matrix representation.

Taking into account the former state $|\psi\rangle$, one *Z-gate* application return:

$$Z|\psi\rangle = Z(\alpha|0\rangle + \beta|1\rangle) = \alpha(Z|0\rangle) + \beta(Z|1\rangle) = \alpha|0\rangle - \beta|0\rangle = |\psi''\rangle \tag{1.2.12}$$

To elucidate the tight relation between *X*- and *Z*- *gates*, two new orthonormal states, denoted with the kets $|+\rangle$ and $|-\rangle$, need to be introduced:

$$\begin{aligned} |+\rangle &= \frac{|0\rangle + |1\rangle}{\sqrt{2}} \\ |-\rangle &= \frac{|0\rangle - |1\rangle}{\sqrt{2}} \end{aligned} \tag{1.2.13}$$

Orthogonality and normalization condition are easily verified, in fact, using eq.(1.1.2), it can be shown that:

$$\begin{aligned} \langle +|-\rangle &= \frac{1}{2}\Big(\langle 0|0\rangle - \langle 0|1\rangle + \langle 1|0\rangle - \langle 1|1\rangle\Big) = 0 \\ \langle +|+\rangle &= \frac{1}{2}\Big(\langle 0|0\rangle + \langle 0|1\rangle + \langle 1|0\rangle + \langle 1|1\rangle\Big) = 1 \\ \langle -|-\rangle &= \frac{1}{2}\Big(\langle 0|0\rangle - \langle 0|1\rangle - \langle 1|0\rangle + \langle 1|1\rangle\Big) = 1 \end{aligned} \tag{1.2.14}$$

With respect to this new base elements, truth tables of *X-gate* and *Z-gate* are:

Input	**Output**
$\|+\rangle$	$\|+\rangle$
$\|-\rangle$	$-\|-\rangle$

Tab. 1.3: *X-gate* truth table referred to $|+\rangle, |-\rangle$ basis.

Input	**Output**
$\|+\rangle$	$\|-\rangle$
$\|-\rangle$	$\|+\rangle$

Tab. 1.4: *Z-gate* truth table referred to $|+\rangle, |-\rangle$ basis.

which recall tables 1.1 and 1.2 but at reverse, thus proving their similarity.

This new basis also allows to introduce the **Hadamard transform gate** $-\boxed{H}-$ which is a pure quantum gate usually addressed to create superposition. In particular, it corresponds exactly to the *change of basis matrix* from the $\{|0\rangle, |1\rangle\}$ to the $\{|+\rangle, |-\rangle\}$ bases and, thanks to the fact that is *self-reversible*, also vice-versa.

Input	**Output**
$\|0\rangle$	$\|+\rangle$
$\|1\rangle$	$\|-\rangle$

Tab. 1.5: *H-gate* truth table.

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

Fig. 1.3: *H-gate* matrix representation.

Self-reversibility can be proved by solving the matrix product:

$$H^2 = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \times \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} = \mathbb{1} \qquad (1.2.15)$$

Arbitrary phase rotation $-\boxed{R_\theta}-$ is another pure quantum gate, defined as a function of an angular parameter θ which determines the amplitude of the rotation. Its importance relies on its capability to generate a *relative shift* on a given state that, in contrast with *global shifts*, has physical relevance and affects *interference phenomena.* Moreover, if $e^{i\theta} \neq \pm 1$, this gate is no longer Hermitian and thus loses self-reversibility.

Input	**Output**
$\|0\rangle$	$\|0\rangle$
$\|1\rangle$	$e^{i\theta}\|1\rangle$

Tab. 1.6: *R_θ-gate* truth table.

$$R_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$$

Fig. 1.4: *R_θ-gate* matrix representation.

The last gate reviewed is the **CNOT gate**, abbreviation for *controlled NOT*, which is binary. In quantum circuit diagrams the controlling qubit is identified with the symbol —●— while the qubit that goes under NOT operation, also called *target qubit*, is represented as —⊕—. It is a fundamental gate for quantum computing, widely used to creates entangled states. To avoid misinterpretations it is conventionally indicated as C_iNOT_j to specify the controlling action of the i-th qubit over the j-th one. By being binary its truth table traces 4 different possible input scenarios and so does the matrix representation which, this time, is based on a 4 × 4 matrix.

Input	**Output**
$\|00\rangle$	$\|00\rangle$
$\|01\rangle$	$\|01\rangle$
$\|10\rangle$	$\|11\rangle$
$\|11\rangle$	$\|10\rangle$

Tab. 1.7: C_1NOT_2*-gate* truth table.

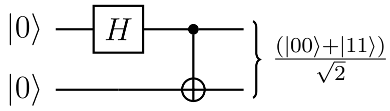
$$C_1NOT_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Fig. 1.5: C_1NOT_2*-gate* matrix representation.

It may be hastily thought that the *CNOT-gate* only operates on the state of the target qubit, but this is not necessarily true; in this sense, the next non-trivial example is ought to help. Supposed an input state $|\psi\rangle = (|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle)$, the application of a C_1NOT_2*-gate* can be then resolved through the following steps :

$$\begin{aligned} C_1NOT_2\ |\psi\rangle &= C_1NOT_2\ (|00\rangle - |01\rangle + |10\rangle - |11\rangle) = \\ &= |00\rangle - |01\rangle + |11\rangle - |10\rangle = \\ &= |0\rangle \otimes (|0\rangle - |1\rangle) - |1\rangle \otimes (|0\rangle - |1\rangle) = \\ &= (|0\rangle - |1\rangle) \otimes (|0\rangle - |1\rangle). \end{aligned} \tag{1.2.16}$$

To conclude, an example of a simple circuit to obtain an entangled state is given by:



where the final state is also known as $|\Phi^+\rangle$ *Bell state*[1].

1.3 Quantum errors modeling for quantum processing

In this last section some details regarding common sources of error during quantum information processing are provided before finally entering the field of QEC. It must be said, in fact, that in the previous parts an extremely ideal setup has been considered with no mention of possible errors occurring. However, a non-ideal quantum system could undergo many events that alter its state, hence causing a loss of information. Quantum errors that could show up in a quantum system are tightly bound to the specific dynamic of the system itself but, in terms of quantum computing, the relevant measure only concern the expected success probability of a quantum algorithm. This facts allows to put aside for a moment the idea of aiming to a meticulous understanding of all the causes behind every possible error of different systems in favor of a more effective approach aimed to group those errors from how they affect quantum computation, thus identifying some common sources of error. Following the logical scheme introduce in the article [2], some standard types of error are briefly reviewed.

1.3.1 Coherent quantum errors

The first source of error considered is related to coherent, systematic control errors. This type of error is typically due to an incorrect knowledge of the system dynamics, mistakenly assumed described by an Hamiltonian $H^{'}$ while it is actually governed by H. It cause otherwise perfect controls on $H^{'}$ to perform systematic coherent inaccuracies for H. However, by being coherent and systematic, it is possible to model those errors quite easily and without moving to the *density matrix formalism.* For instance, considered a straightforward algorithm that perfectly performs on a single qubit system, described by $H^{'}$, a sequence of N identity operators $\hat{\mathbb{1}}$; if the qubit is initialized in the $|0\rangle$ state it returns a final state:

$$|\psi\rangle_{final} = \prod_{i=1}^{N} \hat{\mathbb{1}}_i \ |0\rangle = |0\rangle \, . \tag{1.3.1}$$

However, assumed with purpose of simplification that the incorrect characterization of the system, which dynamics is actually governed by H, leads the algorithm to perform a series of $(\hat{\mathbb{1}} + i\epsilon\hat{\sigma_x})$ instead of $\hat{\mathbb{1}}$ operations, for $\epsilon \ll 1$, then, using the *Pauli exponential formula* (see appendix A.1[3]):

$$\exp\Big(i\theta\vec{n}\cdot\hat{\vec{\sigma}}\Big) = \cos(\theta)\hat{\mathbb{1}} + i\sin(\theta)\vec{n}\cdot\hat{\vec{\sigma}} \tag{1.3.2}$$

the final state can be expressed as:

$$\begin{aligned} |\psi\rangle_{final} &= \prod^{N} \exp(i\epsilon\hat{\sigma_x}) \ |0\rangle = \exp(iN\epsilon\hat{\sigma_x}) \ |0\rangle = \\ &= \Big(\cos(N\epsilon)\hat{\mathbb{1}} + i\sin(N\epsilon)\hat{\sigma_x}\Big)|0\rangle = \\ &= \cos(N\epsilon)|0\rangle + i\sin(N\epsilon)|1\rangle \, . \end{aligned} \tag{1.3.3}$$

which represent a superposition of both states $|0\rangle$ and $|1\rangle$. In particular, in according with the assumption made, the probability of measuring the system in the expected state $|0\rangle$ passes from being unitary to being:

$$P(|0\rangle) = \cos^2(N\epsilon) \approx 1 - (N\epsilon)^2 \tag{1.3.4}$$

thus having a probability of error of amplitude:

$$P_{error} = P(|1\rangle) = \sin^2(N\epsilon) \approx (N\epsilon)^2 \tag{1.3.5}$$

which is small given that $N\epsilon \ll 1$. This kind of result is fundamental in order to formulate and achieve an appropriate *fault-tolerant* quantum computation.

1.3.2 Quantum decoherence

Quantum decoherence is a consequence of the coupling between a quantum system and an *environment*, which causes a loss of coherence between the quantum states and thus a transition of the system into a classical ensemble. At first look then, to prevent the system from moving into classical realm, it may seems necessary to completely isolate it from the ambient in order to avoid the decoherence process. Nevertheless, by doing so, a system would maintain coherence indefinitely but it would also be not possibile to investigate it or operate on it, making it useless for quantum computing. Thus, being able to handle and mitigate decoherence effects and preserve coherence of the states is why QEC protocols are needed. The modeling of quantum decoherence source of errors is based on *density operator formalism*, fundamental for the representation of *mixed states*, which no longer can be expressed as kets but are characterized by matrices in the form:

$$\rho = \sum_{k=1}^{N} p_k \left|\psi_k\right\rangle \left\langle\psi_k\right| . \tag{1.3.6}$$

With respect to the canonical base, a mixed state for an individual qubit takes the form:

$$\rho = \rho_0 \left|0\right\rangle \left\langle 0\right| + \rho_1 \left|1\right\rangle \left\langle 1\right| = \begin{pmatrix} \rho_0 & 0 \\ 0 & \rho_1 \end{pmatrix} \tag{1.3.7}$$

Examples of this type of errors are control errors that arise from stochastic parameters, in particular, remarking the argumentation discussed in [4], a brief overview regarding *phase damping* is provided in the interest of clarity. Phase damping is a quantum noise process which describes a loss of quantum information without loss of energy and can be modelled as follows. Assume a qubit in a state $\left|\psi\right\rangle = \alpha\left|0\right\rangle + \beta\left|1\right\rangle$ undergoes a rotation operation $\mathbf{R}(\theta)$, where the angle θ, named *phase kick angle* is random. The randomness of this parameter could be generated, for instance, from a deterministic interaction with an environment. If the *phase kick angle* has a Gaussian distribution with mean value 0 and variance 2λ, the output state from this process is given by the *density matrix* obtained from averaging over θ (see appendix A.2):

$$\rho = \frac{1}{\sqrt{4\pi\lambda}} \int_{-\infty}^{+\infty} d\theta \; R(\theta) \left|\psi\right\rangle \left\langle\psi\right| R^{\dagger}(\theta) \exp\left(-\frac{\theta^2}{4\lambda}\right) \tag{1.3.8}$$

$$= \begin{pmatrix} |\alpha|^2 & \alpha\beta^* e^{-\lambda} \\ \alpha^*\beta e^{-\lambda} & |\beta|^2 \end{pmatrix} \tag{1.3.9}$$

which is a mixed state. By observing this latter equation, in fact, it could be noted how the off-diagonal elements of ρ decay exponentially to 0 with time, thus causing the the matrix to be diagonal at the end. This is a characteristic result of phase damping.

1.3.3 Simple models of measurement, initialization, loss and leakage errors

Other sources of error such as measurement errors, qubit initialization, qubit loss and qubit leakage can be all modeled after the coherent or incoherent schemes already discussed, depending on the physical mechanism of the error. Measurement errors, for instance, can be modeled in the same way as environmental decoherence and can be described in two ways. The first is by using a *positive operator-valued measure*, also named POVM, that relies on the operators:

$$F_0 = (1 - p_M) \left|0\right\rangle \left\langle 0\right| + p_M \left|1\right\rangle \left\langle 1\right| \tag{1.3.10}$$

$$F_1 = (1 - p_M) \left|1\right\rangle \left\langle 1\right| + p_M \left|0\right\rangle \left\langle 0\right| \tag{1.3.11}$$

that satisfy the condition $F_0 + F_1 = \mathbb{1}$ and in which p_M indicates the probability of measurement errors. It must be kept in mind that this kind of measure differs from standard projector-valued measure, in fact F_0 and F_1 are not *projectors*, i.e. $F_0^2 \neq F_0$ and ${F_1}^2 \neq F_1$. The second method is by mapping the density matrix of the qubit as:

$$\rho \to \rho' = (1 - p_M)\rho + p_M X \rho X \tag{1.3.12}$$

assuming subsequently a perfect measurement operation in relation to the basis $\{\left|0\right\rangle, \left|1\right\rangle\}$. Both those models return exactly the same probabilities for each state; in fact, defined $A_0 = \left|0\right\rangle \left\langle 0\right|$ and $A_1 = \left|1\right\rangle \left\langle 1\right|$ as the measurement projectors on the canonical base, in the first case:

$$P(0) = \mathrm{Tr}(F_0\rho) = (1 - p_M)\,\mathrm{Tr}(A_0\rho) + p_M\,\mathrm{Tr}(A_1\rho) \tag{1.3.13}$$

$$P(1) = \mathrm{Tr}(F_1\rho) = (1 - p_M)\,\mathrm{Tr}(A_1\rho) + p_M\,\mathrm{Tr}(A_0\rho) \tag{1.3.14}$$

while in the second:

$$\begin{aligned} P(0) = \mathrm{Tr}\Big(A_0\rho'\Big) &= (1 - p_M)\,\mathrm{Tr}(A_0\rho) + p_M\,\mathrm{Tr}(X A_0 X \rho) \\ &= (1 - p_M)\,\mathrm{Tr}(A_0\rho) + p_M\,\mathrm{Tr}(A_1\rho) \end{aligned} \tag{1.3.15}$$

$$\begin{aligned} P(1) = \mathrm{Tr}\Big(A_1\rho'\Big) &= (1 - p_M)\,\mathrm{Tr}(A_1\rho) + p_M\,\mathrm{Tr}(X A_1 X \rho) \\ &= (1 - p_M)\,\mathrm{Tr}(A_1\rho) + p_M\,\mathrm{Tr}(A_0\rho) \end{aligned} \tag{1.3.16}$$

with perfect correspondence between the equations (1.3.13)-(1.3.15) and (1.3.14)-(1.3.16). The main difference between these two models is in which state the measured qubit is projected. By using operators F_0 or F_1, the qubit collapses in:

$$\frac{M_i \rho M_i^\dagger}{\mathrm{Tr}(F_i \rho)} \quad with\ i = 0, 1 \tag{1.3.17}$$

where

$$M_0 = \sqrt{1 - p_M} \left|0\right\rangle \left\langle 0\right| + \sqrt{p_M} \left|1\right\rangle \left\langle 1\right| \tag{1.3.18}$$

$$M_1 = \sqrt{1 - p_M} \left|1\right\rangle \left\langle 1\right| + \sqrt{p_M} \left|0\right\rangle \left\langle 0\right| \tag{1.3.19}$$

thus resulting initialized in a not known state while, for the second model, it ended up in the state $\left|0\right\rangle$ or $\left|1\right\rangle$, depending on which projector, A_0 or A_1, is used. However, as measurements are generally followed by either discarding or reinitializing the qubit in a known state, both models are typically accepted.

Moving to qubit initialization source of errors, both coherent and incoherent representation are suitable. Considering first a coherent modeling, an initialized state is represented as a pure state that contains a non-zero probability for the qubit to be in the incorrect state. This imply that, assumed $\left|0\right\rangle$ as the target of the initialization, then:

$$\left|\psi\right\rangle = \alpha \left|0\right\rangle + \beta \left|1\right\rangle \tag{1.3.20}$$

for $1 \gg |\beta|^2 > 0$, that indicates the probability of measuring the system in the undesired target. In contrast, if a incoherent approach is employed, then initialization errors can be modeled fundamentally in the same way of measurement errors; in fact, given a probability p_I of initialization error, then, by initializing a qubit starting in the $\left|0\right\rangle$ state, the process return a density matrix in the form:

$$\rho = (1 - p_I) \left|0\right\rangle \left\langle 0\right| + p_I \left|1\right\rangle \left\langle 1\right| . \tag{1.3.21}$$

For completeness, just a mention to loss and leakage errors, which go past the purposes of pure QEC. A loss error occurs when a qubit is literally removed from the system, which implies a reduction of the dimensionality of the qubits space by a factor of two and a impossibility to interact with the qubit itself. Nonetheless, as this error affects the physical object (i.e. the qubit), it can be barely addressed as a quantum information error, but more as a integrity error. For this reason, qubit loss errors usually requires additional non-demolition detection mechanisms on top of standard QEC protocols to be corrected. Similar strategies are also employed to recover from leakage errors that cause the system to exit its proper qubits subspace. Leakage errors are due to the fact that quantum systems employed as qubit typically do not consist of just 2-state, but are mainly many-state systems. Thus, inaccuracies in the application of controls and decoherence effects can end in states like:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle + \gamma\,|2\rangle \tag{1.3.22}$$

which can lead to unexpected results and unwanted dynamics in several different ways as quantum computation is designed for just 2-state qubits.

Chapter2

Introduction to stabilizer formalism and standard QEC codes

In the first part of this chapter some fundamentals regarding QEC and a brief introduction to *stabilizer formalism* are provided. Subsequently, the 3-qubit and 9-qubit codes are both presented and analyzed using either quantum circuits and the stabilizer formalism.

2.1 Quantum error correction

The ambition of QEC is to develop a quantum error correcting code able to correct any errors that could affect quantum information; however, this purpose goes beyond the contents of this thesis which is limited to discussing quantum error caused by noise, thus without any concern about initialization, quantum gate and measurements errors. As it happened for quantum gates, also QEC moves its first steps based on classical error correction milestones, in particular, the starting point are *repetition codes*, one of the main strategies to protect information in classical computing. Repetition codes essentially consist, as the name suggests, in the creation of maps where every bit of information is copied multiple times: the consistency and the cost in terms of memory of each code clearly depend on how many copies are made. If the error probability p_{err} is small enough, a very basic map, in which every logical bit is associated to three physical bits according to the relations :

$$\begin{aligned} 0 &\rightarrow 000 \\ 1 &\rightarrow 111 \end{aligned} \tag{2.1.1}$$

can already handle any single error with a great level of security. In fact, any physical state that presents one error can be brought back to the original state; for instance, the state 001, can be restored to 000, which is more likely to be the starting state. In this case, misinterpretation only happens if two or three errors occur simultaneously, thus with a probability $P_{mis} = P_{2Errs} + P_{3Errs} = 3p_{err}^2(1 - p_{err}) + p_{err}^3$. If this probability is not tolerated, a heavier mapping or a way to reduce p_{err} must be employed.

Although this may seems a very solid strategy, it does not apply for QEC because of the so called *"No cloning theorem"*. This theorem is a consequence of the linearity of quantum mechanics[5] that precludes the possibility of the existence of cloning-operator such:

$$\hat{U}\left|\psi\right\rangle = \left|\psi\right\rangle \otimes \left|\psi\right\rangle . \tag{2.1.2}$$

To overcome this obstacle, QEC takes advantage of redundant encoding procedures able to *enlarge the Hilbert subspaces*. To make things clear, a trivial encoding process, only able to detect a single error, is given by the relations:

$$\begin{aligned} \left|0\right\rangle &\rightarrow \left|0\right\rangle_L = \left|00\right\rangle \\ \left|1\right\rangle &\rightarrow \left|1\right\rangle_L = \left|11\right\rangle \end{aligned} \tag{2.1.3}$$

which map the logical states of a qubit into two physical qubits. By doing so, considered a generic individual logical state $\left|\psi\right\rangle = \alpha\left|0\right\rangle + \beta\left|1\right\rangle$, this is converted in an encoded state $\left|\psi\right\rangle_e \in \mathcal{H}^4$:

$$\left|\psi\right\rangle_e = \alpha\left|00\right\rangle + \beta\left|11\right\rangle . \tag{2.1.4}$$

In the case an error occurs, for instance $E = X_1 \otimes \mathbb{1}$, the state would be transformed in $\left|\psi\right\rangle_{err} = \alpha\left|10\right\rangle + \beta\left|01\right\rangle$, but $\left|10\right\rangle$ and $\left|01\right\rangle$ does not have any logical sense so an error is detected, nonetheless, with this map not enough information is protected and $\left|\psi\right\rangle_{err}$ is left with no clues on what was its original state. It is important to stress the fact that $\left|\psi\right\rangle_e$ does not correspond to a double copy of the starting $\left|\psi\right\rangle$, which, in contrast, would have result in a state:

$$\begin{aligned} \left|\psi\right\rangle_c = (\alpha\left|0\right\rangle + \beta\left|1\right\rangle)^{\otimes 2} &= (\alpha\left|0\right\rangle + \beta\left|1\right\rangle) \otimes (\alpha\left|0\right\rangle + \beta\left|1\right\rangle) = \\ &= \alpha^2\left|00\right\rangle + \alpha\beta(\left|01\right\rangle + \left|10\right\rangle) + \beta^2\left|11\right\rangle \neq \left|\psi\right\rangle_e . \end{aligned} \tag{2.1.5}$$

Another great complication deriving from quantum physics is the necessity of avoiding direct measurements to preserve superposition. Understanding the *syndrome* that affects a system hence forced QEC to make use of *ancillae qubit*. Examples on how those bits are employed in quantum circtuits are shown in the following sections. Moreover, unlike classical bits, qubits can experience two logical operations: *bit flips* and *phase flips*, hence QEC have to face both *bit flip errors* and *phase flip errors*. For simplicity, considered the canonical base, those can be respectively schematized as follows:

$$\textit{Bit flip error: } \left|0\right\rangle \leftrightarrow \left|1\right\rangle \tag{2.1.6}$$

$$\textit{Phase flip error: } \left|0\right\rangle \rightarrow \left|0\right\rangle ,\ \left|1\right\rangle \rightarrow -\left|1\right\rangle . \tag{2.1.7}$$

It is immediate to note that they recall exactly the operations of an *X-gate* (Tab.1.1) and a *Z-gate* (Tab.1.2), reason why they are also named *X-error* and *Z-error*. For the sake of curiosity one may wonder why does not also exist *Y-errors*. The answer is that *Y-errors* actually exist but considering the relation $Y = -iXZ$, which can be easily verified, and due to linearity of quantum mechanics, *Y-error* can be treated as two independent *X*- and *Z-error*. Finally, one last aspect of quantum errors is that those are *intrinsically continuous*, which means that qubits usually do not experience full bit or phase flips, but are rather affected by flips in the form $\exp(i\alpha X)$ or $\exp(i\beta Z)$. Nevertheless, using the Eq.(1.3.2), those formulas can be *discretized*, hence returning to a scenario with a full bit-flip or phase-flip to correct.

2.2 Stabilizer formalism

The basic idea behind the stabilizer formalism relies on the fact that many quantum states can be more easily described using operators and eigenvalues instead of using the vector representation provided by quantum mechanics. In particular, it is possible to identify special operators, called *stabilizers*, that actually form a group under the operation of matrix multiplication, which implies that they all benefit of the properties of this structure. Consequently, a clever use of *group theory* allows us to study quantum states and quantum errors much more efficiently and compactly compared with a state vector description, making stabilizer formalism almost essential for QEC.

Given a system of n qubits, a group of great interest is the *Pauli group* $\mathcal{P}_n$. Its definition derive after the Pauli group for a single qubit $\mathcal{P}$, which is the collection of all Pauli matrices, previously presented in (1.2.4), up to phase factors $\pm 1, \pm i$:

$$\mathcal{P} = \{\pm\mathbb{1}, \pm i\mathbb{1}, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}. \tag{2.2.1}$$

As stated, this set of matrices forms a group under the operation of matrix multiplication. Phase factors ± 1 and $\pm i$ are necessary to assure closure between the elements of the group. Subsequently, the general definition of the Pauli group for n qubits $\mathcal{P}_n$ consists of all n-fold tensor products of Pauli matrices, again, up to phase factors ± 1 and $\pm i$:

$$\mathcal{P}_n = \{\textit{Tensor product } \otimes \textit{ of } \mathbb{1}, X, Y, Z \textit{ on } n \textit{ qubits, with phase} \pm 1, \pm i\}. \tag{2.2.2}$$

Recalling relations (1.2.6), (1.2.7) and (1.2.8); the following properties can be verified:

- Any element $R \in \mathcal{P}_n$ squares to $\pm\mathbb{1}$;
- Any two elements $R, Q \in \mathcal{P}_n$ either commute or anticommute, hence $[R, Q] = 0$ or $\{R, Q\}$=0.

Finally, a subgroup $S \subseteq \mathcal{P}_n$ is called a *Stabilizer group* if and only if $-\mathbb{1} \notin S$. It follows that, despite its apparent simplicity, this condition also implies for S to be an *Abelian group* due to the commutation and anticommutation relations of its elements. Further, given a stabilizer group S, the subspace $\mathcal{C}_S \subseteq \mathcal{H}^{2^n}$ of all states $|\psi\rangle$ that are stabilized by S, i.e. all the eigenstates of eigenvalue one for the elements of S, is called *codespace* or *stabilizer code induced by* S:

$$\mathcal{C}_S = \{ \ |\psi\rangle \in \mathcal{H}^{2^n} \ | \ \forall R \in S, \ R\,|\psi\rangle = |\psi\rangle \ \}. \tag{2.2.3}$$

Both the definitions of stabilizer group and stabilizer code are of principal interest in QEC. If a stabilizer group S has $r \leq n$ independent generators, then it could be proved[4] that $\dim[S] = 2^r$ and $\dim[\mathcal{C}_S] = 2^{n-r}$, where n indicates the number of physical qubits utilized and $n - r$ the logical qubits represented. Whenever possible, S is typically represented through its independent generators $S =< g_1, g_2, \ldots, g_r >$.

To complete the framework, the definitions of *logical operators* and *distance* of a code are provided. As stated in the previous section, the logical operations that act on a logical qubit are bit flips and phase flips. Hence, without loss of generality, assumed a system with $\dim[\mathcal{C}_S] = 2$ and described by a standard logical base $\mathcal{B}_L = \{|0\rangle_L, |1\rangle_L\}$, logical operators can be schematize as:

$$\textit{Bit flip } \bar{X} : \ |0\rangle_L \leftrightarrow |1\rangle_L \tag{2.2.4}$$

$$\textit{Phase flip } \bar{Z} : \ |0\rangle_L \to |0\rangle_L \,, \ |1\rangle_L \to -\,|1\rangle_L \,. \tag{2.2.5}$$

It is trivial to show that for systems composed of just one qubit $|0\rangle_L \equiv |0\rangle$ and $|1\rangle_L \equiv |1\rangle$, thus the logical operators correspond respectively to the *X-gate* and *Z-gate* operations. However, in general, this is not true; reason why equations (2.2.4),(2.2.5) must not be confused with (2.1.6) and (2.1.7) because logical operators act always on a logical level while error operators are more likely to act on a physical one. Made this premise, the definition of logical operators using stabilizer formalism is no longer so intuitive and requires first the introduction of *Centrilizers.* Considering a stabilizer group $S \subseteq \mathcal{P}_n$, the centralizer of S in $\mathcal{P}_n$ is defined as:

$$C_{\mathcal{P}_n}(S) = \{ \ R \in \mathcal{P}_n \mid RQ = QR \ \textit{ for all } Q \in S \ \}. \tag{2.2.6}$$

Since S is an abelian group, it implies that $S \subseteq C_{\mathcal{P}_n}(S)$. An element $O \in C_{\mathcal{P}_n}(S) \setminus S$ is called a logical operator for $\mathcal{C}_S$. Therefore, logical operators are elements of $\mathcal{P}_n$ that commute with every elements of the stabilizer group able to change the logical state of the system while keeping it confined in the codespace.

Lastly, an intuitive definition of code distance, useful to evaluate the consistency of a code itself, can be given *adapting* the classical *Hamming distance*[6] to stabilizer codes. In particular, given two *codewords* $c_1, \ c_2 \in \mathcal{C}_S$, their Hamming distance can be expressed as:

$$d_H(c_1, c_2) = \min\Big\{ \ |R| \ \Big| \ R \in C_{\mathcal{P}_n}(S) \setminus S, \ R\,|c_1\rangle = |c_2\rangle \ \Big\} \tag{2.2.7}$$

where $|R|$ indicates the *weight* of R corresponding to the number of qubits on which it acts non-trivially. By using this metric[1], the code distance $d(\mathcal{C}_S)$ is defined as:

$$d(\mathcal{C}_S) = \min\Big\{ \; d_H(c_i, c_j) \;\Big|\; c_i, c_j \in \mathcal{C}_S \;\Big\} \equiv \min\Big\{ \; |R| \;\Big|\; R \in C_{\mathcal{P}_n}(S) \setminus S \;\Big\}. \tag{2.2.8}$$

Therefore, employing $d(\mathcal{C}_S) \equiv d$ it is possible to know the maximum number of errors a stabilizer code $\mathcal{C}_S$ can correct using the formula:

$$M = \frac{d-1}{2} \tag{2.2.9}$$

that give an idea of why code distance is such important. Furthermore, in analogy with *coding theory*, also in QEC codes are conventionally identified with $[[n, k, d]]$, where n indicates the number of physical qubits, k the number of logical qubits and d the code distance. Double brackets are only employed to distinguish quantum from classical codes.

2.3 3-Qubit Code

Despite the fact this is not a *full quantum code*, the 3-qubit code is traditionally considered excellent for getting started and familiarizing with the basic concepts of either Quantum Error Correction and Stabilizer formalism. Not being a full quantum code is a consequence of its impossibility to correct simultaneously for both a bit and a phase flip error, which also causes the division between the so called 3-qubit bit-flip code and 3-qubit phase-flip code. However, except for few details, their QEC protocols actually go through the very same logic steps, thus, for the sake of brevity, a further analysis is only reported for the 3-qubit bit-flip code.

The encoding process of the 3-qubit bit-flip code extends the 2-dimensional Hilbert space of a logical qubit to an 8-dimensional space by mapping it into three physical qubits. This codification can be intuitively schematized by the relations:

$$\begin{aligned} |0\rangle &\to |0\rangle_L = |000\rangle \\ |1\rangle &\to |1\rangle_L = |111\rangle \,, \end{aligned} \tag{2.3.1}$$

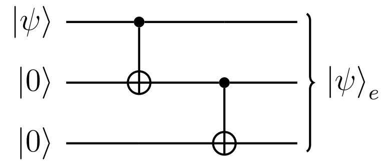
which transform a generic logical state $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$ in:

$$|\psi\rangle_e = \alpha\,|000\rangle + \beta\,|111\rangle \,. \tag{2.3.2}$$

To obtain such a state with a quantum circuit two additional physical qubits initialized

[1] It can be verified that $d_H(\cdot,\cdot)$ is non-negative, symmetric and satisfies the triangle inequality by using a geometrical representation of the codespace[6].

to $|0\rangle$ and two *CNOT-gates* are required:



$$\begin{aligned} |\psi\rangle_e &= C_2NOT_3C_1NOT_2\,|\psi\rangle\,|0\rangle\,|0\rangle = \\ &= C_2NOT_3(\alpha\,|00\rangle + \beta\,|11\rangle)\,|0\rangle = \\ &= \alpha\,|000\rangle + \beta\,|111\rangle = \alpha\,|0\rangle_L + \beta\,|1\rangle_L\,. \end{aligned} \tag{2.3.3}$$

Short considerations on the code distance immediately clarify why this code can only handle one *X-error* and perhaps suggest what it lacks to be a full code. In particular, the logical operator $\bar{X}$, which is defined as:

$$\bar{X} = X_1 \otimes X_2 \otimes X_3; \tag{2.3.4}$$

is characterized by a weight $|\bar{X}| = 3$, thus the code distance is equal to $d_H(|0\rangle_L\,, |1\rangle_L) = 3$ and, from equation (2.2.9), the maximum number of correctable errors is set to one. This means that if an *X-error* occurs, the incorrect state is no longer confined in the codespace but is still closer to the starting logical state, reasons why it can be detected and restored. In contrast, the $\bar{Z}$ logical operator, which can be expressed in multiple forms:

$$\bar{Z} = Z_1 \otimes Z_2 \otimes Z_3 \quad or \tag{2.3.5}$$

$$\bar{Z} \sim Z_1 \otimes \mathbb{1}_2 \otimes \mathbb{1}_3 \sim \mathbb{1}_1 \otimes Z_2 \otimes \mathbb{1}_3 \sim \mathbb{1}_1 \otimes \mathbb{1}_2 \otimes Z_3; \tag{2.3.6}$$

has a weight $|\bar{Z}| = 1$ and so, considered for greater clarity the logical states $|+\rangle_L\,, |-\rangle_L$ introduced in eq.(1.2.13), which, after the encoding, become:

$$\begin{aligned} |+\rangle_L &= \frac{|000\rangle + |111\rangle}{\sqrt{2}} \\ |-\rangle_L &= \frac{|000\rangle - |111\rangle}{\sqrt{2}} \end{aligned} \tag{2.3.7}$$

it can be noticed that $d_H(|+\rangle_L\,, |-\rangle_L) = 1$. This implies that if a *Z-error* occurs, it is not only impossible to correct it, but it could not even been detected due to the fact that the affected state would still be confined in the codespace.

The encode is followed by a *Detection process* against noise errors, which, in this case, is supposed to recognize if an *X-error* has occurred and, eventually, on which qubit. As stated in section 2.1, detection in QEC involves the use of ancillae qubits to protect the superposition. In particular, for the 3-qubit code, the basic idea to extract the syndrome is to perform two parity checks on the data block and store the information in the ancillae qubits. The need to execute two parity checks is due to the fact the code can face four different error scenarios: a trivial no-error situation and the cases a single *X-error* occurs on any of the three qubits.

The detection process described above can be performed with the employment of a quantum circuit like:

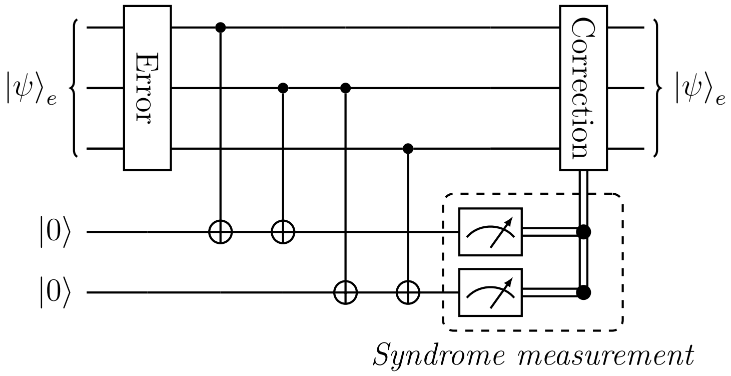


Fig. 2.1: 3-qubit quantum circuit diagram for a single $X - error$ detection and correction.

in which the parity checks, made on qubits q_1q_2 and q_2q_3, are achieved by using a total of four *CNOT-gates*. Finally, as already shown in Fig.2.1, by measuring directly the ancillae qubits it is possible to extract the syndrome and understand which correction is required. For the sake of clarity, in Tab.2.1 are listed and briefly described all the possible error scenarios.

Error location	q_1q_2	q_2q_3	*State after detection:* $\lvert data\rangle \lvert ancillae\rangle$	*Correction*
No error	$+$	$+$	$\alpha\lvert 000\rangle\lvert 00\rangle + \beta\lvert 111\rangle\lvert 00\rangle$	*No correction*
Qubit 1	$-$	$+$	$\alpha\lvert 100\rangle\lvert 10\rangle + \beta\lvert 011\rangle\lvert 10\rangle$	X *on Qubit* 1
Qubit 2	$-$	$-$	$\alpha\lvert 010\rangle\lvert 11\rangle + \beta\lvert 101\rangle\lvert 11\rangle$	X *on Qubit* 2
Qubit 3	$+$	$-$	$\alpha\lvert 001\rangle\lvert 01\rangle + \beta\lvert 110\rangle\lvert 01\rangle$	X *on Qubit* 3

Tab. 2.1: List of possible scenarios for quantum circuit in Fig.2.1.

To describe the 3-qubit code with stabilizer formalism the first fundamental aspect is to identify a proper stabilizer group S. Although this is not typically an easy task, by using the above results, it can be recognised without much effort that the generators of S correspond exactly to the operations of parity check q_1q_2 and q_2q_3. Those controls are respectively equivalent to measuring the eigenvalues of $Z_1 \otimes Z_2 \otimes \mathbb{1}, \mathbb{1} \otimes Z_2 \otimes Z_3 \in \mathcal{P}_3$, hence, to be consistent with the formalism, it is preferable to express them in the latter form, using operators. As shown in Tab.2.2, it immediately follows that the codespace is defined as $\mathcal{C}_{\mathcal{P}_3}(S) = \{|0\rangle_L, |1\rangle_L\}$, and, due to the fact that both stabilizers consist of two Z operators, which do not modify the basis states, it can be easily proven they also commute between themselves and with the logic operators $\bar{X}$ and $\bar{Z}$, thus satisfying all the properties needed.

State	$Z_1 \otimes Z_2 \otimes \mathbb{1}$	$\mathbb{1} \otimes Z_2 \otimes Z_3$
$\|000\rangle$	1	1
$\|001\rangle$	1	-1
$\|010\rangle$	-1	-1
$\|011\rangle$	-1	1
$\|100\rangle$	-1	1
$\|101\rangle$	-1	-1
$\|110\rangle$	1	-1
$\|111\rangle$	1	1

Tab. 2.2: Eigenvalues of the basis states of $\mathcal{H}^8$ in relation to the elements of the stabilizer group.

In conclusion, to truly appreciate and quantify the benefits of redundant encoding in terms of reliability, a brief example involving the 3-qubit code is provided. Assumed, for simplicity, that an error E performs a coherent rotation $\exp(i\epsilon X)$ *with* $\epsilon \ll 1$ and operate on a single qubit. Given an initial state $|\psi\rangle$, this would be transformed in a final state $|\psi\rangle_f$ equals to:

$$|\psi\rangle_f = \exp(i\epsilon X)\,|\psi\rangle = \cos(\epsilon)\,|\psi\rangle + i\sin(\epsilon)X\,|\psi\rangle\,. \tag{2.3.8}$$

Assumed then that E performs identically on each qubit, in the encoded case it would take the form:

$$\begin{aligned} E^{\otimes 3} &= \exp(i\epsilon X)^{\otimes 3} = (\cos(\epsilon)\mathbb{1} + i\sin(\epsilon)X)^{\otimes 3} = \\ &= \cos^3(\epsilon)(\mathbb{1}\otimes\mathbb{1}\otimes\mathbb{1}) + i\cos^2(\epsilon)\sin(\epsilon)(X_1\otimes\mathbb{1}\otimes\mathbb{1} + \mathbb{1}\otimes X_2\otimes\mathbb{1} + \mathbb{1}\otimes\mathbb{1}\otimes X_3) - \\ &- \cos(\epsilon)\sin^2(\epsilon)(X_1\otimes X_2\otimes\mathbb{1} + X_1\otimes\mathbb{1}\otimes X_3 + \mathbb{1}\otimes X_2\otimes X_3) - \\ &- i\sin^3(\epsilon)(X_1\otimes X_2\otimes X_3). \end{aligned} \tag{2.3.9}$$

and applied on the encoded version of $|\psi\rangle \to |\psi\rangle_e$, it would return a $|\psi\rangle_{err}$:

$$\begin{aligned} |\psi\rangle_{err} &= E^{\otimes 3}\,|\psi\rangle_e = \\ &= \cos^3(\epsilon)\,|\psi\rangle_e + \\ &+ i\cos^2(\epsilon)\sin(\epsilon)(X_1\otimes\mathbb{1}\otimes\mathbb{1} + \mathbb{1}\otimes X_2\otimes\mathbb{1} + \mathbb{1}\otimes\mathbb{1}\otimes X_3)\,|\psi\rangle_e - \\ &- \cos(\epsilon)\sin^2(\epsilon)(X_1\otimes X_2\otimes\mathbb{1} + X_1\otimes\mathbb{1}\otimes X_3 + \mathbb{1}\otimes X_2\otimes X_3)\,|\psi\rangle_e - \\ &- i\sin^3(\epsilon)(\bar{X})\,|\psi\rangle_e\,; \end{aligned} \tag{2.3.10}$$

that clearly presents states that cannot be correctly restored. However, once coupled with a circuit as the one in Fig.2.1, $|\psi\rangle_{err}$ undergoes a detection process which leads to:

$$\begin{aligned} |\psi\rangle_{err} |\phi\rangle_{Syndrome} = & \Big[\cos^3(\epsilon) - i\sin^3(\epsilon)\bar{X}\Big] |\psi\rangle_e |00\rangle + \\ & + \Big[i\cos^2(\epsilon)\sin(\epsilon)(X_1 \otimes \mathbb{1} \otimes \mathbb{1}) - \cos(\epsilon)\sin^2(\epsilon)(\mathbb{1} \otimes X_2 \otimes X_3)\Big] |\psi\rangle_e |10\rangle + \\ & + \Big[i\cos^2(\epsilon)\sin(\epsilon)(\mathbb{1} \otimes X_2 \otimes \mathbb{1}) - \cos(\epsilon)\sin^2(\epsilon)(X_1 \otimes \mathbb{1} \otimes X_3)\Big] |\psi\rangle_e |11\rangle + \\ & + \Big[i\cos^2(\epsilon)\sin(\epsilon)(\mathbb{1} \otimes \mathbb{1} \otimes X_3) - \cos(\epsilon)\sin^2(\epsilon)(X_1 \otimes X_2 \otimes \mathbb{1})\Big] |\psi\rangle_e |01\rangle ; \end{aligned} \tag{2.3.11}$$

and subsequently, considering the error corrections in Tab.2.1, it is eventually returned the final state:

$$\begin{aligned} |\psi'\rangle_f = & \Big[\cos^3(\epsilon) + 3i\cos^2(\epsilon)\sin(\epsilon)\Big] |\psi\rangle_e - \\ & - \Big[3\cos(\epsilon)\sin^2(\epsilon) + i\sin^3(\epsilon)\Big]\bar{X} |\psi\rangle_e . \end{aligned} \tag{2.3.12}$$

On one hand, for the *"unencoded"* case the *Fidelity* of $|\psi\rangle_f$ can be easily estimated as:

$$F_{un} = |\langle\psi| E |\psi\rangle|^2 = \cos^2(\epsilon) \approx 1 - \epsilon^2; \tag{2.3.13}$$

on the other hand, the use of the 3-qubit code, assures with a probability $P(\textit{no-err}) = 1 - 3\epsilon^2 + O(\epsilon^4)$ a fidelity:

$$F_e(\textit{no-err}) = \frac{|\cos^3(\epsilon)|^2}{|\cos^3(\epsilon)|^2 + |i\sin^3(\epsilon)|^2} = \frac{cos^6(\epsilon)}{cos^6(\epsilon) + \sin^6(\epsilon)} \approx 1 - \epsilon^6; \tag{2.3.14}$$

while, with a probability $P(\textit{err }) = 3\epsilon^2 + O(\epsilon^4)$:

$$\begin{aligned} F_e(\textit{err}) = & \frac{|3i\cos^2(\epsilon)\sin(\epsilon)|^2}{|3i\cos^2(\epsilon)\sin(\epsilon)|^2 + |-3i\cos(\epsilon)\sin^2(\epsilon)|^2} = \\ = & \frac{\cos^4(\epsilon)\sin^2(\epsilon)}{\cos^4(\epsilon)\sin^2(\epsilon) + \cos^2(\epsilon)\sin^4(\epsilon)} = \\ = & \frac{1}{1 + \frac{\sin^2(\epsilon)}{\cos^2(\epsilon)}} = \cos^2(\epsilon) \approx 1 - \epsilon^2. \end{aligned} \tag{2.3.15}$$

In an ideal set then, the 3-qubit code is able to suppress with a probability $\propto 1 - 3\epsilon^2$ the error from $O(\epsilon^2) \rightarrow O(\epsilon^6)$ and to keep it of the same order if a physical error is detected. Nevertheless, if $\epsilon \ll 1$, most of the cycles of correction would detect no error, thus QEC is largely preferable.

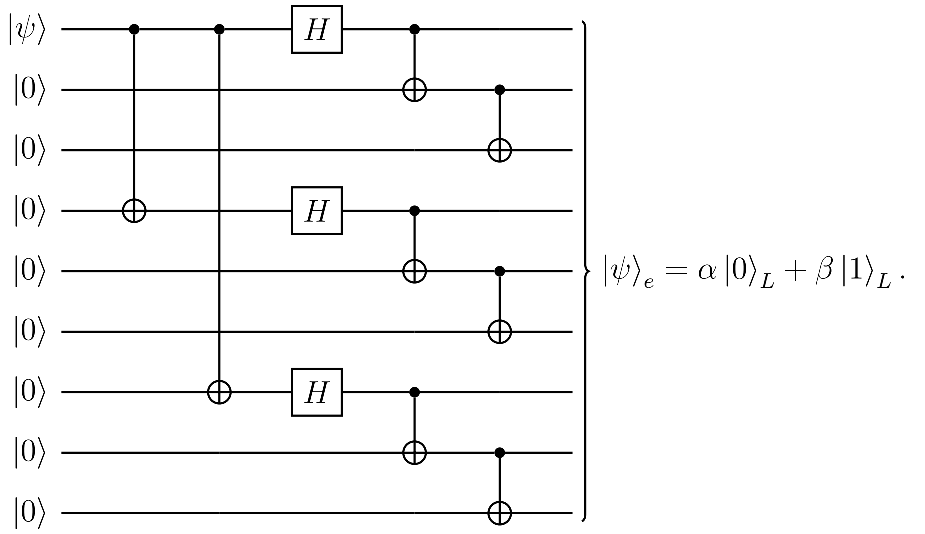
One last thing that must be stressed is how QEC protocols cannot, in general, completely restore a corrupted state to the original one resulting instead in a final state that is a superposition of the correct and incorrect states ((2.3.8),(2.3.12)); reason why fault-tolerant computation is also required in quantum information.

2.4 9-Qubit Code

The 9-qubit code, also named Shor's code after him theorizing it in 1995, was the first full quantum code discovered. Largely based on the 3-qubit redundant encoding, this is a *degenerate* single-error correcting code able to protect a logical qubit from one bit-flip or phase-flip error, and even both, on any physical qubits of the data block. With regards to its degeneracy, which generally occurs when different types of error affect in the same way the codespace, a lucid explanation can be found later in the analysis. The encode of the 9-qubit code, as the name suggests, involves the use of nine physical qubits and maps the logical states in:

$$\begin{aligned} |0\rangle \rightarrow |0\rangle_L &= \frac{1}{2\sqrt{2}}(|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \otimes (|000\rangle + |111\rangle) \\ |1\rangle \rightarrow |1\rangle_L &= \frac{1}{2\sqrt{2}}(|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle) \otimes (|000\rangle - |111\rangle) \end{aligned} \tag{2.4.1}$$

expanding the 2-dimensional logical space into a physical $\mathcal{H}^{2^9}$ space. The circuit needed to realize this encoding clearly takes inspiration from the one seen for the 3-qubit code, but on a larger scale and given a generic state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, it returns :



The logical operators related to those states have not very complicated forms but rather curious, in particular:

$$\bar{X} = Z_1 \otimes Z_2 \otimes Z_3 \otimes Z_4 \otimes Z_5 \otimes Z_6 \otimes Z_7 \otimes Z_8 \otimes Z_9 \tag{2.4.2}$$

$$\bar{Z} = X_1 \otimes X_2 \otimes X_3 \otimes X_4 \otimes X_5 \otimes X_6 \otimes X_7 \otimes X_8 \otimes X_9. \tag{2.4.3}$$

By looking at the basis states of the entire data block it is possible to identify three sub-blocks formed by 3 qubits. For each of them, single *X-error* detection and correction remark exactly the ones employed for the 3-qubit code and can be performed using the very same circuit (Fig.2.1). A direct consequence to point out then is that, according to this QEC protocol, the full data block can handle a maximum of three *X-errors* under the assumption of only one occurring in every sub-block.

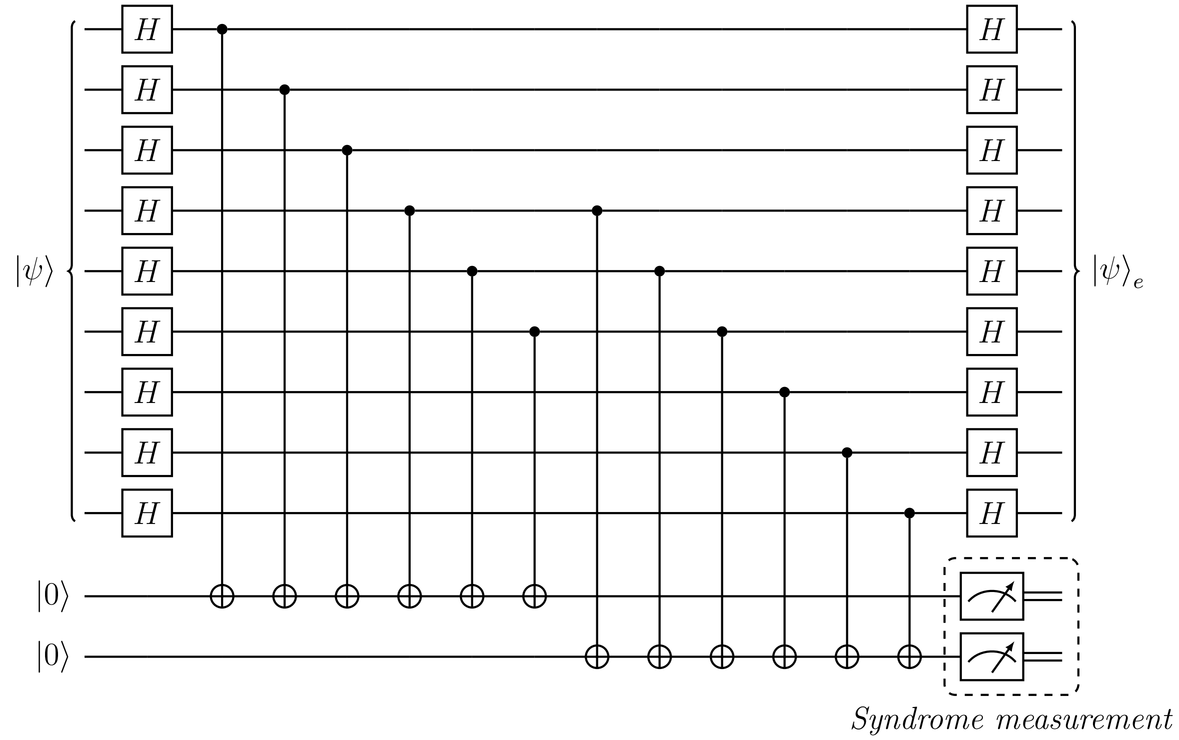


Fig. 2.2: 9-qubit quantum circuit diagram to performs parity checks between sub-blocks b_1b_2 and b_2b_3

Moving to *Z-error* detection and correction, a clever reformulation of the basis elements really highlights how those work. In particular, using the encoded states presented in Eq.(2.3.7), logical states can be rewritten as:

$$\begin{aligned} |0\rangle_L &= |+\rangle_L \otimes |+\rangle_L \otimes |+\rangle_L = |\mathbf{+++}\rangle \\ |1\rangle_L &= |-\rangle_L \otimes |-\rangle_L \otimes |-\rangle_L = |\mathbf{---}\rangle \\ \textit{introducing: } & |\mathbf{+}\rangle \equiv |+\rangle_L \;\; \textit{and} \;\; |\mathbf{-}\rangle \equiv |-\rangle_L \,. \end{aligned} \tag{2.4.4}$$

Once more, then, it is possible to effectively detect the error by performing two parity checks and subsequently extract the syndrome from the ancilla qubits to restore the data block. Although the procedure is the same, the fact that this time parity checks are between 3-qubit states makes quantum circuits much more complicated. An example is given in Fig.2.2 where the comparisons are performed between sub-blocks b_1b_2 and

b_2b_3. This is also why the code is degenerate, in fact, every qubit in a specific sub-block affected by a *Z-error* would end up in causing the same syndrome, and, vice-versa, provided which sub-block have to be restored, the correction is *qubit-invariant*, in the sense it can be applied on each of the three qubits. Using stabilizer formalism, the 9-qubit code can be entirely described with a stabilizer group made of eight operators: six of them corresponding to the parity checks performed inside the sub-blocks and two equivalent to the *sign comparisons* between the sub-blocks themselves. For clarity, all those stabilizers are collected in Tab.2.3, where can be also found their expressions in terms of operators belonging to $\mathcal{P}_9$. Commutativity can be verified between all the stabilizers and the logical operators.

Generator	*Comparison*	*Full operator expression*								
S_1	q_1q_2	Z_1	Z_2	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_2	q_2q_3	$\mathbb{1}$	Z_2	Z_3	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_3	q_4q_5	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	Z_4	Z_5	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_4	q_5q_6	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	Z_5	Z_6	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_5	q_7q_8	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	Z_7	Z_8	$\mathbb{1}$
S_6	q_8q_9	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	Z_8	Z_9
S_7	b_1b_2	X_1	X_2	X_3	X_4	X_5	X_6	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$
S_8	b_2b_3	$\mathbb{1}$	$\mathbb{1}$	$\mathbb{1}$	X_4	X_5	X_6	X_7	X_8	X_9

Tab. 2.3: Stabilizers of the 9-qubit code

To conclude, it is mentioned that logical operators (Eqs.(2.4.2), (2.4.3)) can be expressed in many other reduced forms obtainable through specific combinations of the operators themselves with the code stabilizers. For instance, combining $\bar{X}, S_2, S_4$ and S_6 returns a valid logical operator:

$$\bar{X}' = Z_1 \otimes \mathbb{1} \otimes \mathbb{1} \otimes Z_4 \otimes \mathbb{1} \otimes \mathbb{1} \otimes Z_7 \otimes \mathbb{1} \otimes \mathbb{1} \equiv \bar{X}. \tag{2.4.5}$$

Chapter3

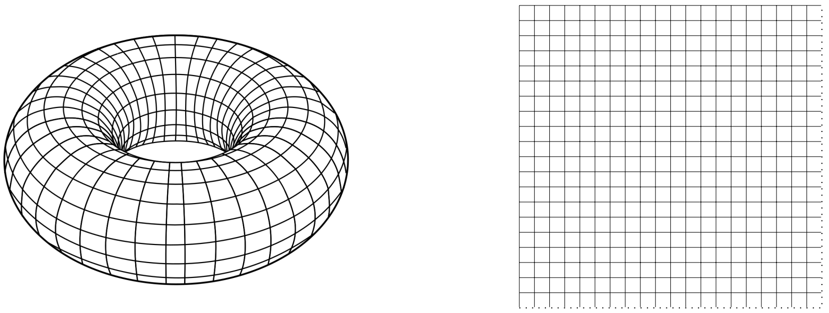
The Toric Code and the Threshold Theorem

In this last section a more complicated and relevant stabilizer code, named *Toric code*, is reviewed. In particular the analysis is meant to accurately describe its structure and its principal properties. Subsequently, QEC protocols regarding the Toric code are discussed in order to study the possibility of fault-tolerant quantum computation and eventually estimate the accuracy threshold of the code itself. This would inevitably lead to the formulation of the *Threshold theorem*.

3.1 The Toric Code

Our attention is now brought to *Surface codes*, and in particular to the Toric code. Surface codes are a subclass of stabilizer codes characterized by a geometrical representation, usually as lattices embedded on geometrical surfaces. In some special case then, taking advantage of the topological formalism and of the properties of the surface considered, remarkably simplify the description of the code itself.

The Toric code, as shown in Fig.(3.1), is modelled as a square lattice with *periodic boundaries*, a property, this last, which grants *translation invariance* and allows it to be imagined as a 2D lattice embedded on a Torus $\mathbb{T}^2$, from which it was named after.



(a) 2D square lattice embed on a torus [7].

(b) 2D square lattice with periodic boundaries.

Fig. 3.1: (a), Visualization of the cellulation on a torus. **(b)**, Cellulation of a torus mapped in a planar lattice: dot edges represent periodic boundaries.

This identification derives from an operation called Cellulation[8] of the torus which can be bijectively mapped with a 2D square periodic lattice. To each edge ℓ is assigned one qubit: assuming an $L \times L$ square lattice $\mathcal{L}^2$, this results in a Toric code formed by a total of $N_{edges} = 2L^2$ qubits, thus belonging to a $\mathcal{H}_{Toric} = \otimes_{2L^2} \mathcal{H}^2$ Hilbert space. Further, two higly local operators, A_v and B_p, are associated to each *vertex* v and to each *plaquette* p, respectively. These operators are defined as:

$$A_v = \prod_{\substack{\otimes \\ j \in v}} X_j \quad \textit{where } j \textit{ indicates all the qubits around } v; \tag{3.1.1}$$

$$B_p = \prod_{\substack{\otimes \\ j \in p}} Z_j \quad \textit{where } j \textit{ indicates all the qubits enclosing } p. \tag{3.1.2}$$

Consider figure 3.2 for a clearer visualization of the entire structure.

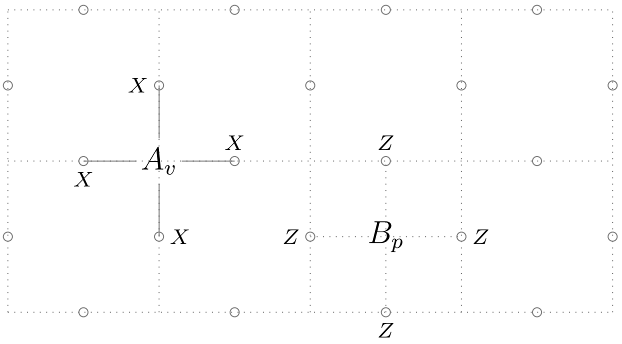


Fig. 3.2: Detailed view of the Toric code structure where can be seen qubits are placed on edges and are also shown one vertex and one plaquette operator.

Using the *Euler characteristic*[9] for a torus:

$$\chi(\mathbb{T}^2) = N_{vertices} - N_{edges} + N_{plaquettes} = 0; \tag{3.1.3}$$

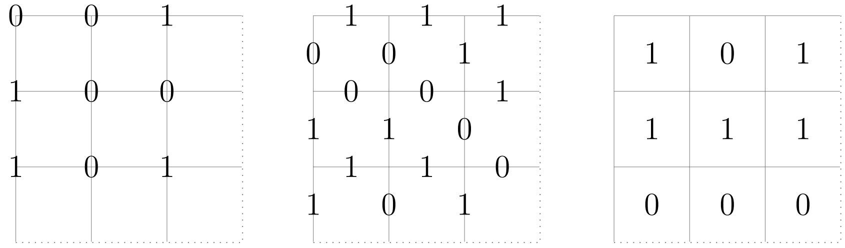
it is therefore possible to know the total number of vertices $N_{vertices} = L^2$ and plaquettes $N_{plaquettes} = L^2$, which form the set of all vertices $V_{\mathcal{L}^2}$ and plaquettes $P_{\mathcal{L}^2}$, and so the total number of vertex and plaquette operators.

Operators A_v and B_p are also called *checks operators* and, although X and Z anticommute, it can verified those all mutually commute between themselves. In particular it is trivial to note that vertex operators commute with vertex operators as they can possibly execute only one same operation on a shared qubit, and so does plaquette operators with plaquette operators. Moreover, vertex operators and plaquette operators, when adjacent, can only overlap with each other by sharing two qubit operations, for instance, $X_i \otimes X_j$ and $Z_i \otimes Z_j$, but this also implies two cancelling minus arises when considering the commutator, hence they also commute. In the case they are not even adjacent, commutativity is trivial. Checks operators thus form an Abelian group which is the Toric code's stabilizer group S_{Toric}. Its dimensionality is a direct consequence of the lattice geometrical properties. Due to periodic boundaries every qubit experiences the action of exactly two vertex and two plaquette operators, then, recalling Eq.(1.2.6), it follows that:

$$\prod_{v \in V_{\mathcal{L}^2}} A_v = \prod_{p \in P_{\mathcal{L}^2}} B_p = 1. \tag{3.1.4}$$

Bearing in mind this constraint, each vertex or plaquette operator can be then expressed as the product of the others $L^2 - 1$ operators and S_{Toric} loses two degrees of freedom. No more relations exists among those operators so it possible to conclude that $\dim(S_{Toric}) = 2(L^2 - 1)$ and that 2 logical qubits are encoded in the Toric code.

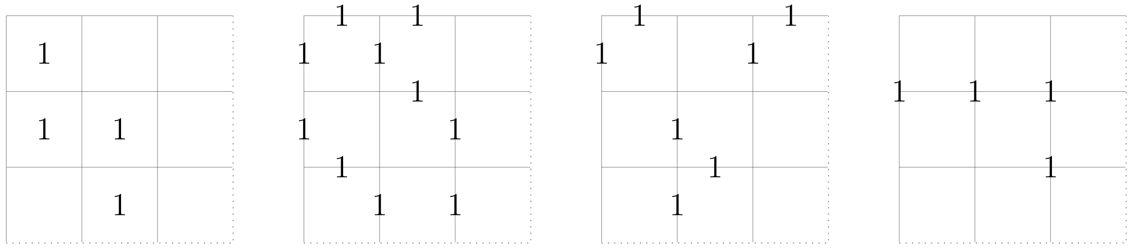
To identify the logical operators of a Toric code[10] it requires first being able to identify Pauli operators which commute with all the checks operators. Nonetheless, to find and define such Pauli operators it is convenient to previously introduce some useful elements of *Homology*[8], to be specific the $\mathbb{Z}^2$ Homology, which allows to face effectively this problem. In particular, *zero-chains, one-chains, two-chains* and *boundary operators* are the essential objects needed in our case. Speaking in general, *n-chains* are substantially maps which assign, in the case of a $\mathbb{Z}^2$ Homology, elements of $\mathbb{Z}^2 = \{0, 1\}$ to geometrical entities. The prefixes proposed are intuitively referred to the dimension of the geometrical entity: *zero-* identifies a map for vertices, *one-* for edges and *two-* for plaquettes. Simple examples of these chains can be found in Fig.3.3.



(a) *Zero-chain.* (b) *One-chain.* (c) *Two-chain.*

Fig. 3.3: Visualization of different types of $\mathbb{Z}^2$-chains.

In order to simplify and with slight abuse of notation *n-chains* are also represented omitting zeros. All those class of objects respectively form a group once paired with *modulo-2 addition* in $\mathbb{Z}^2$, which means that, given two *n-chain* c_1, c_2, their modulo-2 summation still return a *n-chain*. Returning then the attention just to *zero-*, *one-* and *two-chains*, linear boundary operators ∂_2 and ∂_1 are defined. In particular ∂_2 transforms *two-chains* into *one-chains*, and, subsequently, ∂_1 *one-chains* into *zero-chains*: the boundary of a plaquette is composed of the four edges enclosing it while the boundary of an edge consists of the two vertices at its ends. Some examples can be found in Fig.3.4.



(a) *Two-chain.* (b) *One-chain* resulting from (a). (c) *One-chain.* (d) *Zero-chain* resulting from (c).

Fig. 3.4: (a)-(b), Visualization of ∂_2 acting on a *two-chain.* **(c)-(d)**, Visualization of ∂_1 acting on a *one-chain.*

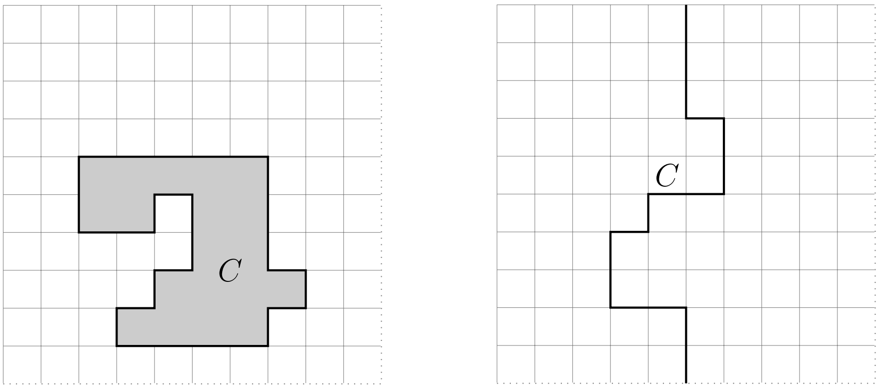
A chain whose boundary is trivial is called a *cycle*; for instance, the *zero-chain* of a plaquette is a cycle.

Now, recalling Eq.(1.2.10), any Pauli operators acting on the Toric code can be expressed as a tensor product of X's times a tensor product of Z's, which can be identified as $\mathcal{P}_{\mathcal{L}^2}^{\otimes X}$ and $\mathcal{P}_{\mathcal{L}^2}^{\otimes Z}$ with:

$$\mathcal{P}_{\mathcal{L}^2} = \mathcal{P}_{\mathcal{L}^2}^{\otimes X} \otimes \mathcal{P}_{\mathcal{L}^2}^{\otimes Z}. \tag{3.1.5}$$

Considering first $\mathcal{P}_{\mathcal{L}^2}^{\otimes Z}$, its action can be thought as a $\mathbb{Z}^2$ *one-chain* by assigning the value 1 to the edges where a Z operation actually occurs and 0 anywhere else (i.e. the edges on which $\mathbb{1}$ is applied). It can be then verified that $\mathcal{P}_{\mathcal{L}^2}^{\otimes Z}$ trivially commutes with all plaquette operators but, with respect to vertex operators, the commutativity only holds if an even number of Z acts on the edges adjacent to every vertices. This last condition can be also reformulated by saying that the *one-chain* representing $\mathcal{P}_{\mathcal{L}^2}^{\otimes Z}$ have to be cyclical. Regarding $\mathcal{P}_{\mathcal{L}^2}^{\otimes X}$, it can be described in the very same way of its Z-counterpart by just considering the dual lattice $\mathcal{L}'\ ^2$, defined as the the lattice where vertices and plaquettes are interchanged. Finally, unifying both results, it can be concluded that a Pauli operator $\mathcal{P}_{\mathcal{L}^2}$ commutes with all the check operators iff, once decomposed as in Eq.(3.1.5), its Z and X components are respectively represented by a cyclical *one-chain* in $\mathcal{L}^2$ and $\mathcal{L}'\ ^2$.

To eventually identify logical operators it must be now taken into account the fact that cyclical *one-chains* are of two distinct types: *Homologically trivial* and *Homologically non-trivial.* The former can be expressed as the boundary of a *two-chain* while the latter cannot; intuitively this means that, while the firsts enclose an *interior*, the seconds do not. Consider figure 3.5 to appreciate lucidly this distinction. At this point, it must be pointed out how a Z's tensor product that ends up in a homologically trivial *one-cycle* can be also expressed as a composition of all the plaquette operators contained in it, hence resulting in an operator contained in the stabilizer group that acts trivially on the codespace. And again, thanks to the square lattice *self-duality*, this exact consideration extents to X's tensor products associated with homologically trivial *one-cycles* in the dual lattice.

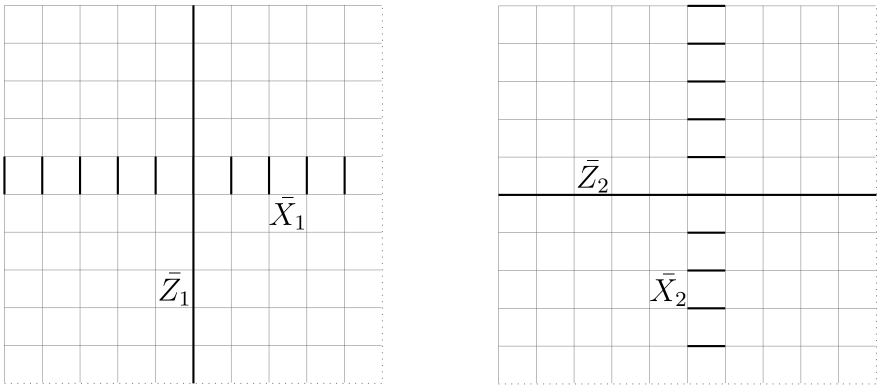


(a) *Homologically trivial* cyclical *one-chain.*

(b) *Homologically non-trivial* cyclical *one-chain.*

Fig. 3.5: **(a)**, Visualization of the interior of a *homologically trivial* cyclical *one-chain.* **(b)**, Visualization of a *homologically non-trivial* cyclical *one-chain*, in this case an interior cannot be identified.

In contrast, Z's products, and similarly X's in $\mathcal{L}'\,^2$, corresponding to homologically non-trivial *one-cycles* still commute with all the elements of the stabilizer group but cannot be represented as a combination of those, hence they are not stabilizers and act non trivially on the encoded qubits. In the end, it is possible to identify the logical operators $\bar{Z}_1$ and $\bar{Z}_2$ with the two non-trivial cycles of a torus, and $\bar{X}_1, \bar{X}_2$ with the two non-trivial cycles of the dual torus (i.e. the one resulting from the dual lattice). Anticommutativity between $\bar{X}_1, \bar{Z}_1$ and $\bar{X}_2, \bar{Z}_2$ can be easily verified; in particular, considering Fig.3.6, it could be immediately seen how those operators execute just an operation on a shared qubit, hence ordinary anticommutation relation between X and Z is inherited.



(a) Operators $\bar{X}_1$ and $\bar{Z}_1$.

(b) Operators $\bar{X}_2$ and $\bar{Z}_2$.

Fig. 3.6: **(a)**, Visualization of the operators $\bar{X}_1$ and $\bar{Z}_1$ as *homologically non-trivial* cycles. **(b)**, Visualization of the operators $\bar{X}_2$ and $\bar{Z}_2$ as *homologically non-trivial* cycles.

To complete the description, a somehow general definition of the logical states of the Toric code is given using *physical sense.* Due to its lattice structure, one of the main properties of this code is it have a *topological order*, which, in the first place, is clearly a geometrical trait; nevertheless, it also acquires physical sense with the introduction of the Hamiltonian[11] H_{Toric}, defined as:

$$H_{Toric} = -\sum_{v\,\in\,V_{\mathcal{L}^2}} A_v - \sum_{p\,\in\,P_{\mathcal{L}^2}} B_p. \tag{3.1.6}$$

De facto, the purpose of an Hamiltonian is to measure the *energy* of a system. In our case, to understand how to quantify the energy in a Toric code it must be considered first the relation:

$$(A_v)^2 = (B_p)^2 = \mathbb{1} \qquad \forall\, v\, \in V_{\mathcal{L}^2} \textit{ and } \forall\, p\, \in P_{\mathcal{L}^2}; \tag{3.1.7}$$

which is a direct consequence of the self-reversibility of X's and Z's operator and implies that all A_v and B_p have eigenvalues ±1. Thus, check operators can be imagined as highly local 2-state systems. It is then possible to define the *ground state* $|\psi_0\rangle$ of the Toric code as the state characterised by the lowest energy possible $H_{0\ Toric} = -(N_{vertices}+N_{plaquettes})$, such a state must satisfy the conditions:

$$A_v\,|\psi_0\rangle = |\psi_0\rangle \quad \forall\, v\, \in V_{\mathcal{L}^2} \tag{3.1.8}$$

$$B_p\,|\psi_0\rangle = |\psi_0\rangle \quad \forall\, p\, \in P_{\mathcal{L}^2}. \tag{3.1.9}$$

In the most general case, given any $|\psi\rangle$ and using projectors, $|\psi_0\rangle$ can be expressed as:

$$|\psi_0\rangle = \frac{1}{4}\Big\{ \prod_{v\,\in\,V_{\mathcal{L}^2}} (\mathbb{1} + A_v) \prod_{p\,\in\,P_{\mathcal{L}^2}} (\mathbb{1} + B_p) \Big\}\,|\psi\rangle\,; \tag{3.1.10}$$

but, conventionally, this form is shrinked in:

$$|\psi_0\rangle = \prod_{v\,\in\,V_{\mathcal{L}^2}} (\mathbb{1} + A_v)\, \underbrace{|0\rangle \otimes \cdots \otimes |0\rangle}_{N_{edges}\ times}\,. \tag{3.1.11}$$

up to a normalization constant. Finally, it is reasonable to identity $|00\rangle_L$ with $|\psi_0\rangle$, hence, the other logical states can be expressed as:

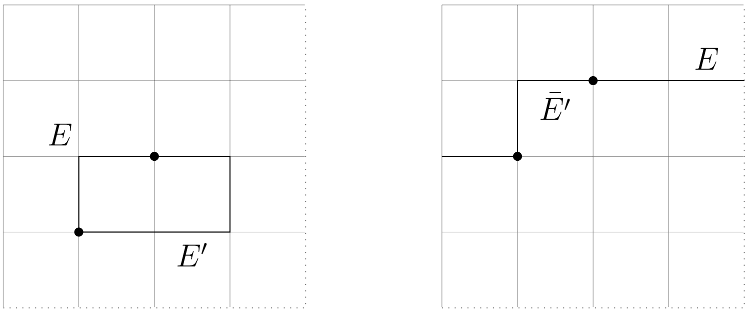
$$|10\rangle_L = \bar{X}_1\,|00\rangle_L \tag{3.1.12}$$

$$|01\rangle_L = \bar{X}_2\,|00\rangle_L \tag{3.1.13}$$

$$|11\rangle_L = \bar{X}_1\bar{X}_2\,|00\rangle_L\,. \tag{3.1.14}$$

3.2 Fault-tolerant computation and accuracy threshold

Quantum error correction for the Toric code also benefits of its geometrical properties and makes extensive use of topological formalism as well. The extraction of the syndrome is performed by measuring check operators: in a correct configuration all of them have value $+1$. If errors occur, the operators affected assume value -1 and, thanks to the fact each of them occupy a precise position in the lattice, it is possible to build two $\mathbb{Z}^2$ *zero-chains*: one related to Z errors in $\mathcal{L}^2$ and one for X's in $\mathcal{L}'^2$. Then, once more, thanks to the self-duality, it is possible to correct both X and Z error chains in the very same way, thus, for brevity, only Z errors will be further considered. Given a syndrome, it is however ambiguous infer the exact *error-chain* E that have caused it, in fact many *one-chains* of *error links* could typically suit in the scenario. Nevertheless, as long as the *correction-chain* E' applied is *homologically equivalent* and has the same boundary of the one actually occurred, error correction is successful. This property can be easily verified noting that $E'E \in S_{Toric}$, hence their composition acts trivially on the code state, as can be seen in Fig.3.7(a). To be specific, *homological equivalence* implies that the correction applied is in the form $E' = E + C$, where C is a homologically trivial cycle. In contrast, if the infer leads to hypothesize a *homologically inequivalent* correction $\bar{E}'$, like in Fig.3.7(b), error correction fails.



(a) *Homologially equivalent error-chains.*

(b) *Homologically inequivalent error-chains.*

Fig. 3.7: **(a)**, Example of ambiguous syndrome: visualization of a valid *correcting-chain* E'. Note how $E+E'$ form a trivial cycle. **(b)**, Example of incorrect *correcting-chain* $\bar{E}'$. In this case $E + \bar{E}'$ form a non-trivial cycle.

The attention is now aimed to the robustness of the Toric code. To discuss this aspect theoretically and at its fundamental level, some proper assumptions are made. To be specific, only errors caused by noise are considered and syndrome measurements are always considered perfectly executed. Moreover, X and Z errors are considered uncorrelated and with an equal probability p to occur. Finally, to preserve the high locality of quantum computing acting on the code, the syndrome measurements necessary to establish the recovery process take place in an isolated classical computer which performs instantaneously. According to this conditions it can be immediately noted that, given a Toric code represented by $\mathcal{L}^2$, this is generally affected by $\sim p(2L^2)$ errors. Hence, considering its code distance, L, error-recovery would substantially always fail. Hypothetically speaking, just $\frac{L}{2}$ errors are sufficient to corrupt the data block; nevertheless this scenario is incredibly atypical, and the solidity of the Toric code is way higher. Actually, under the hypothesis of a completely stochastic error generation, achieving a probability p small enough for the number of errors to scale linearly with the block size, in the limit for $L \to \infty$, would assure a successful error recovery with a probability tending to one. Despite its generality, this consideration inevitably anticipate the concept of accuracy threshold.

To be more specific, and recalling that error correction fails when the *error-chain* deduced $\bar{E}'$ is homologically inequivalent to the *error-chain* occurred E, i.e. it can be expressed as $\bar{E}' = E + W$ with W denoting a non-trivial cycle; it is possible to affirm that fault-tolerant computation is successfully achievable when error probability p lies down an accuracy threshold limit p_c, standing for *critical point probability*, iff, in the limit for $L \to \infty$:

$$\sum_E P(E) \cdot \sum_W P(E + W|E) = 0 \tag{3.2.1}$$

where $P(E + W|E)$ corresponds to the conditional probability to obtain the *error-chain* $\bar{E}'$ once fixed the *error-chain* E and the summation is performed over all the possible non-trivial cycles W. Finally, introducing again some *physical sense* and some *statistics*, the value of p_c can be estimated as follows.

Fixed an *error-chain* E, its *one-chain* representation is the result of a *characteristic function*, a map, acting on all the edges of the lattice $n_E(\ell)$ which assigns the value +1 if ℓ is part of E and 0 anywhere else. The probability $P(E)$ can be then expressed as:

$$P(E) = \prod_{\ell \in \mathcal{L}^2} (1-p)^{1-n_E(\ell)} \, p^{n_E(\ell)}. \tag{3.2.2}$$

Subsequently, the conditional probability $P(E'|E)$ of an hypothetical chain E' with the same boundary of E can be expressed in a similar form considering the relation $E' = E + C$ where C indicates a cycle. In particular, taking advantage of both the characteristic functions, $n_C(\ell)$ and $n_E(\ell)$; $n_{E'}(\ell)$ can be written, making the proportionality with ℓ implicit, as:

$$n_{E'} = (1 - n_E)n_C + (1 - n_C)n_E \tag{3.2.3}$$

which implies that:

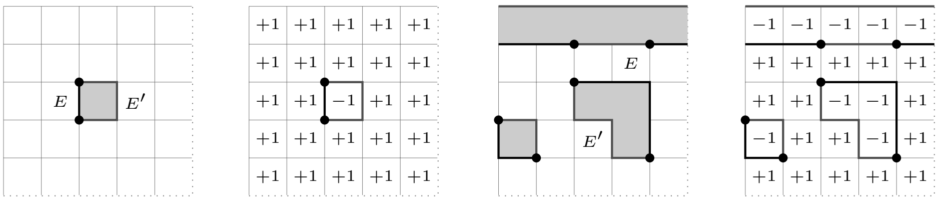
$$\begin{aligned} P(E'|E) &= \prod_{\ell \in \mathcal{L}^2} (1-p)^{1-n_{E'}}\, p^{n_{E'}} \\ &= \prod_{\ell \in \mathcal{L}^2} (1-p)^{1-n_C-n_E+2n_C n_E}\, p^{n_C+n_E-2n_C n_E} \\ &= \prod_{\ell \in \mathcal{L}^2} (1-p)^{1-n_E(\ell)}\, p^{n_E(\ell)} \Big(\frac{p}{1-p}\Big)^{n_C-2n_C n_E} . \\ &= P(E) \prod_{\ell \in \mathcal{L}^2} \Big(\frac{p}{1-p}\Big)^{n_C(1-2n_E)} \end{aligned} \tag{3.2.4}$$

There, is where *physical sense* is once more needed. This result indeed, can be interpreted much more efficiently making an analogy with a high local classical statistical model with *quenched disorder*, named *Random-Bond Ising Model* [12]. The basic idea is the following: *error-chains* can be thought as wall domains in an *Ising ferromagnet* and their boundaries corresponds to *Ising vortices*, which, in our case, are fixed due to perfect syndrome measurements assumption; hence, the disorder is quenched. The set of chains that have such vertices as boundary shows the properties of a thermal ensemble where the temperature is somehow related to p, in fact, as the error probability increase, also the numbers of vertices become higher and many more configurations start populating the initial set, this can be seen as the wall domains *heating up* and start fluctuating more vigorously. At a critical temperature the walls domains condense and the system experience a phase transition. This phase transition corresponds to the phase boundary between a data block on which fault-tolerant error recovery is successful and one in which it fails.

To clarify this identification it is useful to introduce the Hamiltonian operator that describes the coupling of atomic spins in absence of an external field interacting with the system:

$$H_{Coupling} = -J \sum_{<i,j>} \eta_{i,j}\sigma_i\sigma_j \tag{3.2.5}$$

where $\sigma_i(p) \in \{+1,-1\}$ is a map of all the plaquettes of $\mathcal{L}'^2$ and η_{ij} is the coupling between the plaquettes $i-j$ and takes the value -1 if the link ℓ shared between the plaquettes i and j is affected by an error while it takes value $+1$ if no error happens on ℓ. The sum over $< i, j >$ indicates that coupling only occurs between adjacent spins, which is therefore an highly local effect. It is then possible to interpret the lattice and the error syndrome under a complete different light, in fact it can be defined a *ground-state* represented by a configuration where an unique domain exist and spins are all aligned, conventionally they are all assumed at +1. Then, the introduction of errors determines an increase of *energy* due to the creation of wall domains where spins are *antiparallel*. Consider Fig.3.8 to visualize this identification.



(a) Single error configuration.

(b) *Cold* configuration.

(c) Multiple errors configuration.

(d) *Hot* configuration.

Fig. 3.8: (a)-(b), Visualization of a single error scenario, guessing a possible *correction-chain* is surely easier in this case. We can interpret this configuration as a *cold* one. **(c)-(d)**, Visualization of a multiple errors scenario, in this case infer the *correction-chain* to protect quantum information is very complicate and the probability of failing are very high. It is immediate to note a great level of disorder.

By introducing the elements of this new lattice, it follows that in equation (3.2.4):

$$n_C(\ell) = \sigma_i \sigma_j \tag{3.2.6}$$

where i, j identifies the adjacent plaquettes to ℓ and finally, the relation:

$$\exp(-2J_\ell) = \begin{cases} \frac{p}{1-p} & \textit{for } \ell \notin E; \\ \frac{1-p}{p} & \textit{for } \ell \in E. \end{cases} \tag{3.2.7}$$

between the coupling and the bond probability defines the so called *Nishimori line* in the phase diagram of the model from which it can be obtained the value $p_c = 0.1094 \pm 0.0002$ [10].

Conclusions

In this thesis a very contemporary and thrilling problem which finds fertile ground in theoretical physics is analysed. The hope to achieve one day the so called *Quantum supremacy* surely provide great motivations, nonetheless, as a matter of fact, quantum information processing have to face a wide range of problems caused by the stochastic nature of quantum mechanics. Using simplified models, those problematics are briefly discussed in this thesis, and, while some of them are expected to be controllable through engineeristic improvements, others are addressed by quantum error correction protocols. Despite an initial scepticism discouraged QEC researches in its early years, in 1995 Shor proved that quantum error correction was possible with an incredibly intuitive and pedagogical code. Reasons why, the Shor's code, with the simpler 3-qubit code, are also presented in this thesis. Their descriptions subsequently find a great ally in the *Stabilizer formalism* which allows to shorten, simplify and make them even clearer.

Finally, in the last chapter, the *Toric code*, which shows great potential in terms of achieving fault-tolerant quantum computation thanks to its geometrical and physical properties, is analyzed. Moreover, it is a great example of how being able to identify analogies with problems of different nature can greatly enhance the comprehension of the first. Nevertheless the description of the *Toric code* proposed is far away to be actually complete. Its modelization has been approached under very ideal assumptions: being able to progressively relax them is surely one stimulating challenge.

AppendixA

Explicit calculation

A.1 Exponential form 1.3.2

$$\begin{aligned}\exp\Big(i\theta\vec{n}\cdot\hat{\vec{\sigma}}\Big) &= \sum_{k=0}^{\infty}\frac{(i\theta)^k}{k!}(\vec{n}\cdot\hat{\vec{\sigma}})^k = \\ &= \sum_{k=0}^{\infty}\frac{i^{2k}\theta^{2k}}{(2k)!}(\vec{n}\cdot\hat{\vec{\sigma}})^{2k} + \sum_{k=0}^{\infty}\frac{i^{2k+1}\theta^{2k+1}}{(2k+1)!}(\vec{n}\cdot\hat{\vec{\sigma}})^{2k+1} = \\ &= \sum_{k=0}^{\infty}\frac{(-1)^k\theta^{2k}}{(2k)!}\hat{\mathbb{1}} + \sum_{k=0}^{\infty}\frac{i(-1)^k\theta^{2k+1}}{(2k+1)!}(\vec{n}\cdot\hat{\vec{\sigma}}) = \\ &= cos(\theta)\hat{\mathbb{1}} + isen(\theta)(\vec{n}\cdot\hat{\vec{\sigma}})\end{aligned} \tag{A.1.1}$$

A.2 Density matrix 1.3.9

Calculations of ρ_{11} and ρ_{22} are trivial, reason why only the estimation of ρ_{12} is made explicit. The quantity ρ_{21} satisfies $\rho_{21} = \rho_{12}^*$ given the hermiticity of the density matrix.

$$\begin{aligned}\rho_{12} &= \frac{1}{\sqrt{4\pi\lambda}}\int_{-\infty}^{+\infty} d\theta\ a(b^*e^{-i\theta})e^{-\frac{\theta^2}{4\lambda}} = \\ &= \frac{ab^*}{\sqrt{4\pi\lambda}}\int_{-\infty}^{+\infty} d\theta\ e^{-(\frac{\theta^2}{4\lambda}+i\theta)} = \\ &\textit{completing the square:}\quad (\frac{\theta^2}{4\lambda}+i\theta) = (\frac{\theta}{2\sqrt{\lambda}}+i\sqrt{\lambda})^2+\lambda \\ &= \frac{ab^*}{\sqrt{4\pi\lambda}}e^{-\lambda}\int_{-\infty}^{+\infty} d\theta\ e^{-(\frac{\theta}{2\sqrt{\lambda}}+i\sqrt{\lambda})^2} = \\ &\textit{replacing:}\quad (\frac{\theta}{2\sqrt{\lambda}}+i\sqrt{\lambda}) = z \quad \textit{from which} \quad d\theta = 2\sqrt{\lambda}dz \\ &= \frac{2ab^*\sqrt{\lambda}}{\sqrt{4\pi\lambda}}e^{-\lambda}\int_{-\infty}^{+\infty} dz\ e^{-z^2} = \frac{ab^*}{\sqrt{\pi}}e^{-\lambda}\sqrt{\pi} = ab^*e^{-\lambda}\end{aligned} \tag{A.2.1}$$

Bibliography

[1] Eleanor Rieffel and Wolfgang Polak. *Quantum Computing: A Gentle Introduction.* 1st. The MIT Press, 2011. ISBN: 9780262015066.

[2] Simon J Devitt, William J Munro, and Kae Nemoto. "Quantum error correction for beginners". In: *Reports on Progress in Physics* 76.7 (June 2013), p. 076001. ISSN: 1361-6633. DOI: 10.1088/0034-4885/76/7/076001. URL: http://dx.doi.org/10.1088/0034-4885/76/7/076001.

[3] Roberto Zucchini. "Quantum Mechanics: lecture notes." 2020. Lectures delivered at the University of Bologna.

[4] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition.* Cambridge University Press, 2010. DOI: 10.1017/CBO9780511976667.

[5] Daniel Gottesman. *Stabilizer Codes and Quantum Error Correction.* 1997. arXiv: quant-ph/9705052 [quant-ph].

[6] R. W. Hamming. "Error detecting and error correcting codes". In: *The Bell System Technical Journal* 29.2 (1950), pp. 147–160.

[7] Yassine Mrabet. *simple Torus.* Created with Inkscape. 2007. URL: https://commons.wikimedia.org/wiki/File:Simple_Torus.svg.

[8] Dan Browne. "Topological Codes and Computation: lecture notes." May 2014. A lecture course given at the University of Innsbruck.

[9] M. Nakahara. *Geometry, topology and physics.* 2003.

[10] Eric Dennis et al. "Topological quantum memory". In: *Journal of Mathematical Physics* 43.9 (2002), pp. 4452–4505. DOI: 10.1063/1.1499754. eprint: https://doi.org/10.1063/1.1499754. URL: https://doi.org/10.1063/1.1499754.

[11] M. F. Araujo de Resende. "A pedagogical overview on 2D and 3D Toric Codes and the origin of their topological orders". In: *Reviews in Mathematical Physics* 32.02 (Aug. 2019), p. 2030002. ISSN: 1793-6659. DOI: 10.1142/s0129055x20300022. URL: http://dx.doi.org/10.1142/S0129055X20300022.

[12] Barbara M. Terhal. "Quantum error correction for quantum memories". In: *Reviews of Modern Physics* 87.2 (Apr. 2015), pp. 307–346. ISSN: 1539-0756. DOI: 10.1103/revmodphys.87.307. URL: http://dx.doi.org/10.1103/RevModPhys.87.307.