Alma Mater Studiorum-Università di Bologna
Dipartimento di Ingegneria Elettrica e dell'Informazione
"Guglielmo Marconi"

Corso di Laurea Magistrale in
Ingegneria Elettronica

DIRECT DISPERSIVE PARITY MEASUREMENTS FOR DYNAMICALLY GENERATED LOGICAL QUBITS

Elaborato in

dispositivi, circuiti ed algoritmi per il calcolo quantistico m

Relatore:
Prof. Antonio Gnudi

Presentato da:
Edoardo Casadei

Correlatori:
Prof.ssa Elena Gnani
Dott. Alessandro Ciani

Anno accademico 2024/2025

Abstract

Italian version

Con questo lavoro di tesi si presenta lo studio di metodi per ottenere modelli di rumore equivalenti per misure dirette a due qubit e l'applicazione di essi a piattaforme superconduttive in regime dispersivo. L'interesse in tale argomento è dovuto al fatto che di recente sono stati pubblicati codici di correzione degli errori detti Floquet codes o dynamical codes che sfruttano queste misure di parità a peso due codificando i qubit in maniera dinamica. L'obiettivo era dunque capire come queste misure potessero indurre errori e testare come questi codici di correzione reagissero al modello. I metodi sono stati studiati e applicati ad un particolare tipo di errore semplificato che consiste in un dephasing all'interno del sottospazio di parità pari di un sistema a due qubit, dopodichè il modello è stato implementato in un simulatore per il calcolo della soglia in funzione dell'ampiezza del segnale di ingresso al risonatore di readout.

English version

With this thesis work we present the study of methods to obtain equivalent error models for direct two-qubit parity measurements and their applications to superconducting platforms in dispersive regime. The interest in this topic is due to recently published error correcting codes called Floquet codes or dynamical codes that use these weight-2 parity measurements to encode dynamical qubits. The goal was then to understand how these measurements could induce errors and benchmark how these dynamical codes respond to the error model. The methods have been studied and applied to a particular simplified error consisting in an even-parity subspace dephasing for a two-qubit system and we obtained a threshold simulation as a function of the amplitude of the signal applied to the resonator in the readout phase.

Contents

Introduction

Since the first ideas proposed by Richard P. Feynman [7] in 1982 regarding the hint to build quantum machines to unlock new methods to simulate quantum systems, scientists and engineers from all over the world started developing theories and experiments to predict or verify the capabilities of these machines called quantum computers. There are a lot of results showing that quantum machines can perform specific tasks with even exponentially lower complexity compared to the classical counterpart thanks to the so-called quantum parallelism. For example the factorization of prime numbers with a quantum computer is possible within an exponentially faster algorithm called Shor's algorithm [31] or binary search in a database can be performed quadratically faster thanks to Grover's algorithm [11]. This led to waves of enthusiasm in the field and researchers became eager to find new fast algorithms and applications for the quantum speed-up obtained with a set of known algorithms. These applications can go from drug discovery and forecast predictions to logistics, finance and much more.

Regarding the best candidates to host physical realizations of quantum computers, there are for example superconducting platforms, neutral atoms and semiconductor qubits but many other platforms are being developed like topological qubits engineering Majorana bound states. All these platforms have their strong features and weaknesses but one single thing is common to all of them: quantum processes are extremely weak against noise and so this implies the necessity of quantum error correcting codes to reach a point where these technologies can become scalable and useful.

This thesis stands in the middle between the physics of superconducting platforms and quantum error correcting codes, so first of all we will introduce the reader to the needed concepts of quantum error correction in chapter 1 and quantum electrodynamics of superconducting circuits in chapter 2; then an overview on open quantum systems for error models will be presented in chapter 3 and in the end – in chapter 4 – all the knowledge will be gathered in a workflow from a master equation that describes the interactions in the quantum system to an equivalent stochastic channel that models the errors. Studying error models of physical platforms is very important in modern quantum computing for two reasons: first of all we can benchmark existing quantum error correcting codes to see how they perform when trying to correct these errors, then it is also extremely important to know the errors happening in a platform because it can lead to the design of error-aware error correcting codes made ad hoc for the particular physical realization. In this project we chose to mix the construction of a model and to observe how a new class of quantum error correcting codes called Floquet codes [13] perform with the specific error model we developed.

Chapter 1

Quantum Error Correction

Quantum Error Correction (QEC) can be considered as the branch of Quantum Information Theory that studies how to contrast decoherence and correct the undesired evolution of (open) quantum systems. The concepts of this chapter are inspired by the contents of [26], [32], [25] and [2]. We recommend the reader to check Appendices A, B, C and D to ensure you have most of the knowledge needed to understand the topics of this chapter.

1.1 Basic concepts

Quantum states are encoded using a (usually large) number of redundant qubits. This abundance of non-logically needed dimensions will help us to preserve the information contained in k logical qubits encoded in the total n-qubit system.

Definition 1.1.1 ([26]). *Let $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$ be the total Hilbert space that describes the n-qubit system. We define a **quantum error correcting code** as a subspace $\mathcal{C} \subseteq \mathcal{H}$.*

Error correction operations are performed periodically to minimize (in principle annihilate) the continuous degradation of quantum information caused by a certain error $\mathcal{E}$. The whole procedure of error correction (detection followed by recovery) is usually performed measuring some observables, whose eigenvalues are called error syndromes $\boldsymbol{\beta}$, and then applying a proper error recovery operator $\mathcal{R}(\boldsymbol{\beta})$ that can evolve the state back to the original code space $\mathcal{C}$, hopefully removing the error. An error correcting cycle can be then summarized starting from a state $|\phi\rangle \in \mathcal{C}$ through the following steps:

$$|\phi\rangle \xrightarrow{\text{External noise } \mathcal{E}} |\phi'\rangle \xrightarrow{\text{Syndrome measurement}} \boldsymbol{\beta} \xrightarrow{\text{Syndrome recovery } \mathcal{R}} |\phi\rangle$$

The recovery operator $\mathcal{R}$ and usually (but not always) the noise model $\mathcal{E}$ are trace preserving quantum operation.
To get a successful error correcting cycle we must impose that for all density operators ρ whose support is the code space $\mathcal{C}$:

$$(\mathcal{R} \circ \mathcal{E})(\rho) \propto \rho \tag{1.1}$$

where the proportionality symbol is used instead of the equality symbol because $\mathcal{E}$ can be a non-trace preserving quantum operation. Equation 1.1 is the starting point to determine the properties that our quantum error correcting code must satisfy such as the following theorem.

Theorem 1.1.1 (**Quantum Error Correction Conditions** [26]). *Let $\mathcal{C} \subseteq \mathcal{H}$ be a quantum error correcting code and $\mathcal{P}$ be the projector on such a code space. Consider now a generic quantum operation $\mathcal{E}$, identified as the error operation, which is defined by the Kraus operators $\{E_i\}$, usually referred to as errors.*

We claim that $\mathcal{P}E_i^\dagger E_j\mathcal{P} = \alpha_{ij}\mathcal{P}$ with $\boldsymbol{\alpha}$ self-adjoint matrix is a necessary and sufficient condition for the existence of a recovery operator $\mathcal{R}$ that can correct $\mathcal{E}$.
The conditions $\mathcal{P}E_i^\dagger E_j\mathcal{P} = \alpha_{ij}\mathcal{P}$ are denoted as quantum error correction conditions.

Proof. We prove first the **sufficiency** of the condition. It is assumed that $\boldsymbol{\alpha}$ is a self-adjoint matrix and so it can be diagonalized such that $\boldsymbol{d} = u^\dagger \boldsymbol{\alpha}\, u$ is a diagonal matrix and u is a unitary matrix. Consider now the following transformation on the errors $\{E_i\}$ built from the matrix elements of u and defined as:

$$F_k = \sum_i u_{ik} E_i.$$

By Theorem D.0.1 highlighted in Appendix D we know that the errors $\mathcal{F}$ and $\mathcal{E}$ are identical. We can now work on the expression $\mathcal{P}F_k^\dagger F_\ell \mathcal{P}$ as follows:

$$\mathcal{P}F_k^\dagger F_\ell \mathcal{P} = \sum_{ij} u_{ki}^\dagger u_{j\ell}\mathcal{P}E_i^\dagger E_j\mathcal{P} = \sum_{ij} u_{ki}^\dagger u_{j\ell}\alpha_{ij}\mathcal{P} = d_{k\ell}\mathcal{P}. \tag{1.2}$$

Equation 1.2 is referred to as simplified QEC conditions and they are completely equivalent to the form introduced in the claim. We can now apply the left polar decomposition theorem D.0.2 highlighted in Appendix D to the expression $F_k\mathcal{P}$ such that for some unitary operator $\mathcal{U}_k$:

$$F_k\mathcal{P} = \mathcal{U}_k\sqrt{\mathcal{P}F_k^\dagger F_k\mathcal{P}} = \mathcal{U}_k\sqrt{d_{kk}\mathcal{P}} = \sqrt{d_{kk}}\mathcal{U}_k\mathcal{P}.$$

The last equality can be justified because, choosing as square root of $\mathcal{P}$ the only possible positive semidefinite operator such that $(\sqrt{\mathcal{P}})^2 = \mathcal{P} = \mathcal{P}^2$, this implies $\sqrt{\mathcal{P}} = \mathcal{P}$. It is also immediate to see that F_k rotates the code space (projector) into a new subspace defined by $\mathcal{P}_k = \mathcal{U}_k\mathcal{P}\mathcal{U}_k^\dagger = \frac{1}{\sqrt{dkk}}F_k\mathcal{P}\mathcal{U}_k^\dagger$. An important observation is that the QEC conditions imply that those subspaces are orthogonal. This statement can be proven as follows:

$$\mathcal{P}_\ell\mathcal{P}_k = \mathcal{P}_\ell^\dagger\mathcal{P}_k = \frac{1}{\sqrt{d_{\ell\ell}d_{kk}}}\mathcal{U}_\ell\mathcal{P}F_\ell^\dagger F_k\mathcal{P}\mathcal{U}_k^\dagger = \frac{d_{k\ell}}{\sqrt{d_{\ell\ell}d_{kk}}}\mathcal{U}_\ell\mathcal{P}\mathcal{U}_k^\dagger = \mathcal{U}_k\mathcal{P}\mathcal{U}_k^\dagger\delta_{k\ell}.$$

Adding now to the set of orthogonal projectors $\mathcal{P}_k$ one more orthogonal projector to satisfy the completeness relation $\sum_k \mathcal{P}_k = \mathbb{1}$, we can define a projective measurement to extract the syndromes and recovery our original state thanks to the $\mathcal{U}_k^\dagger$ operators. Consider ρ and σ as, respectively, the before and after error application density operators that represent the quantum state of the system. The recovery operation is then defined as:

$$\mathcal{R}(\sigma) = \sum_k \mathcal{U}_k^\dagger\mathcal{P}_k\sigma\mathcal{P}_k\mathcal{U}_k. \tag{1.3}$$

If we express this operation as a function of the original density operator ρ it is possible to finally show that the imposed condition is sufficient to get a perfect recovery:

$$\mathcal{R}(\mathcal{E}(\rho)) = \sum_{k,\ell} \mathcal{U}_k^\dagger \mathcal{P}_k^\dagger F_\ell \rho F_\ell^\dagger \mathcal{P}_k \mathcal{U}_k = \frac{1}{d_{kk}} \sum_{k,\ell} \mathcal{U}_k^\dagger \mathcal{U}_k \mathcal{P} F_k^\dagger F_\ell \rho F_\ell^\dagger F_k \mathcal{P} \mathcal{U}_k^\dagger \mathcal{U}_k =$$

$$= \frac{1}{d_{kk}} \sum_{k,\ell} d_{k\ell}^2 \mathcal{P}\rho\mathcal{P} = \sum_{k,\ell} d_{kk}\delta_{k\ell}\rho \propto \rho.$$

This proves the sufficiency of the imposed condition.

The **necessity** of the imposed condition can be proved in the following way: suppose that the error represented by the $\{E_i\}$ can be corrected by a recovery operation $\mathcal{R}$ defined by some $\{R_j\}$. At this point we can define the map $\mathcal{E}_\mathcal{C}$ acting on a generic density operator such that $\mathcal{E}_\mathcal{C}(\rho) = \mathcal{E}(\mathcal{P}\rho\mathcal{P})$. We are sure that $\mathcal{P}\rho\mathcal{P} \in \mathcal{C}$ and so the recovery on that state can be performed through $\mathcal{R}$. Mathematically this leads us to a recovery equation for the operator $\mathcal{E}_\mathcal{C}$ such that:

$$\mathcal{R}(\mathcal{E}_\mathcal{C}(\rho)) = \mathcal{R}(\mathcal{E}(\mathcal{P}\rho\mathcal{P})) \propto \rho.$$

After this consideration we can expand the symbols and obtain the equation:

$$\sum_{i,j} R_j E_i \mathcal{P}\rho\mathcal{P} E_i^\dagger R_j^\dagger = c\mathcal{P}\rho\mathcal{P}$$

where $c \in \mathbb{C}$. This leads us to the equivalence between the quantum operation defined by the elements $R_j E_i \mathcal{P}$ and the quantum operation defined by the single element $\sqrt{c}\mathcal{P}$. Defining now the elements $E_{ij} = R_j E_i \mathcal{P}$ and $F_k = F = \sqrt{c}\mathcal{P}$ we can apply Theorem D.0.1 and so we get the equations:

$$R_k E_i \mathcal{P} = c_{ki}\mathcal{P} \xrightarrow{\dagger} \mathcal{P} E_i^\dagger R_k^\dagger = c_{ki}^* \mathcal{P}$$

that can be multiplied to obtain:

$$\mathcal{P} E_i^\dagger R_k^\dagger R_k E_j \mathcal{P} = c_{ki}^* c_{kj} \mathcal{P}.$$

Reminding now that $\mathcal{R}$ is trace preserving, hence $\sum_k R_k^\dagger R_k = \mathbb{1}$, and summing over k the previous equation we obtain:

$$\mathcal{P} E_i^\dagger E_j \mathcal{P} = \alpha_{ij}\mathcal{P}$$

with $\alpha_{ij} = \sum_k c_{ki}^* c_{kj} = \alpha_{ji}^*$. The last equation proves that if $\mathcal{E}$ is a correctable error via an $\mathcal{R}$ operation the QEC conditions are necessary, hence the theorem is proven. □

In the literature it is also common to express the QEC conditions taking their matrix element respect to an orthonormal basis of the code space $\{|\psi\rangle_k \in \mathcal{C}\}$ obtaining the so-called *Knill-Laflamme conditions* for QEC [20],[25]:

$$\langle\psi_k|E_i^\dagger E_j|\psi_\ell\rangle = \alpha_{ij}\delta_{k\ell}. \tag{1.4}$$

In the previous theorem, the error $\mathcal{E}$ was referred as *correctable error*. This terminology is correct, but we can get much more than that because the set of all the errors $\{E_i\}$ defines a *set of correctable errors*. We can then enunciate the following theorem.

Theorem 1.1.2 ([26])**.** *Let $\mathcal{C} \subseteq \mathcal{H}$ be a quantum error correcting code. Consider now the correctable error operation $\mathcal{E}$ defined by the errors $\{E_i\}$ and the associated recovery operator $\mathcal{R}$.*

The error operation $\mathcal{F}$ defined by the errors $F_j = \sum_i m_{ji} E_i$ can be corrected by the same quantum operation $\mathcal{R}$.

Proof. To ensure that $\mathcal{E}$ is a correctable error, the set of errors E_i must satisfy the QEC conditions $\mathcal{P}E_iE_j^\dagger\mathcal{P} = \alpha_{ij}\mathcal{P}$. Without loss of generality we can assume $\boldsymbol{\alpha}$ to be diagonal like we did for the proof of the simplified conditions for the previous theorem, hence $\boldsymbol{\alpha} = \boldsymbol{d}$, with $\boldsymbol{d}$ diagonal. We know that $\boldsymbol{\alpha}$ is a self-adjoint matrix; this implies that diag($\boldsymbol{d}$) is made of real elements (they are the eigenvalues of $\boldsymbol{\alpha}$). From equation 1.3 we know that, to obtain the appropriate recovery operation, $\mathcal{R}$ must be defined by the elements $\mathcal{U}_k^\dagger\mathcal{P}_k$. The theorem can be then proven starting from this expression:

$$\mathcal{R}(\mathcal{F}(\rho)) = \sum_{k,j}\mathcal{U}_k^\dagger\mathcal{P}_kF_j\rho F_j^\dagger\mathcal{P}_k\mathcal{U}_k = \sum_{k,j,i,\ell} m_{j,i}m_{j,\ell}^*\mathcal{U}_k^\dagger\mathcal{P}_kE_i\rho E_\ell^\dagger\mathcal{P}_k\mathcal{U}_k.$$

We can now plug in the last equation the relation

$$\mathcal{P}_k = \frac{1}{\sqrt{d_{kk}}}\mathcal{U}_k\mathcal{P}E_k^\dagger = \frac{1}{\sqrt{d_{kk}}}E_k\mathcal{P}\mathcal{U}_k^\dagger$$

to obtain

$$\frac{1}{d_{kk}}\sum_{k,j,i,\ell} m_{j,i}m_{j,\ell}^*\mathcal{U}_k^\dagger\mathcal{U}_k\mathcal{P}E_k^\dagger E_i\rho E_\ell^\dagger E_k\mathcal{P}\mathcal{U}_k^\dagger\mathcal{U}_k = \frac{1}{d_{kk}}\sum_{k,j,i,\ell} m_{j,i}m_{j,\ell}^*\mathcal{P}E_k^\dagger E_i\mathcal{P}\rho\mathcal{P}E_\ell^\dagger E_k\mathcal{P} =$$

$$= \frac{1}{d_{kk}}\sum_{k,j,i,\ell} m_{j,i}m_{j,\ell}^*d_{ki}d_{k\ell}\mathcal{P}\rho\mathcal{P} = \sum_{k,j}|m_{jk}|^2d_{kk}\rho \propto \rho.$$

This proves the statement because we recovered the density operator corrupted by generic combination of E_i errors. □

Now we can ask ourselves: how many errors can a quantum code code correct? A possible answer to this question is hidden in the *distance* d of a code. To define the distance we first need to introduce the weight of an n-qubit Pauli operator, which is the number of qubits the operator acts on non trivially, hence the number of qubits where it applies an X, Y or Z operator.

Definition 1.1.2 ([32])**.** *Given a code $\mathcal{C} \in \mathcal{H}$ we define the distance d of the code $\mathcal{C}$ as the minimum weight of any logical operator acting on the code space.*

We now observe that given a code of distance $d = 2t+1$ we can correct up to t-weight errors E_j. That is because, plugging this information in equation 1.4, the product $E_i^\dagger E_j$ cannot be a logical operator if the maximum weight of the errors is t. This statement is a lower bound in the sense that errors with weight lower than t can be always corrected, but it does not prove that errors with higher weight cannot be corrected.

1.2 Stabilizer codes

Stabilizer codes are a cornerstone of QEC. They are the straightforward generalization of classical error correcting codes and they offer a more compact description compared to a set of state vectors that generate the code space. These codes are based on the so-called *stabilizer formalism* and we will highlight as follows the framework used and its main results.

The main idea behind the stabilizer formalism is that the quantum states of the code space $\mathcal{C}$ encode k logical qubits in n physical qubits and $\mathcal{C}$ can be easily described by the set (to be precise, the group) of operators that stabilizes it. What do we mean by an operator that stabilizes a state? A state $|\psi\rangle \in \mathcal{H}$ is said to be stabilized by the operator $\mathcal{O}$ if $\mathcal{O}|\psi\rangle = |\psi\rangle$. We can now define the stabilizer group as follows:

Definition 1.2.1 ([32]). *Let $\mathcal{S}$ be a subgroup of the Pauli group $\mathcal{G}_n$. $\mathcal{S}$ is a **stabilizer group** if it is Abelian and $-\mathbb{1} \notin \mathcal{S}$.*

A stabilizer group can be identified by one of its possible sets of independent generators $\{s_1, ..., s_m\}$ such that $\mathcal{S} = \langle s_1, s_2, ..., s_m \rangle$ with $s_j \in \mathcal{G}_n \, \forall j$.

Definition 1.2.2 ([32]). *Let $\mathcal{S}$ be a stabilizer group. We define the **stabilizer code** on $\mathcal{S}$ as the vector space $\mathcal{L}(\mathcal{S}) = \left\{ |\psi\rangle \in \mathcal{H} : s |\psi\rangle = |\psi\rangle \ \forall s \in \mathcal{S} \right\}$*

We can see that each element of the stabilizer group fixes a certain degree of freedom (DOF) of the n-qubits system to its $+1$ eigenspace. In the proper basis this corresponds to the loss of a logical qubit. From this observation we can see that if we have a set of $n - k$ independent stabilizer generators, then $\dim(\mathcal{L}(\mathcal{S})) = 2^k$ and we are encoding k logical qubits. This last statement (proved formally in theorem D.0.4 in Appendix D) leads to the existence of k logical couples $(\tilde{X}_j, \tilde{Z}_j)_{j=1,2,...,k}$. Those elements must commute with the elements of the stabilizer group and so they are part of the so-called centralizer of $\mathcal{S}$ in $\mathcal{G}_n$ defined as $\mathcal{C}_{\mathcal{G}_n}(\mathcal{S}) = \left\{ g \in \mathcal{G}_n : \forall s \in \mathcal{S}, [g, s] = 0 \right\}$. From the definition of the centralizer we can say that the logical operators of a stabilizer code are elements of $\mathcal{C}(\mathcal{S})/\mathcal{S}$ and so we can define the distance of a stabilizer code as $d = \min_{p \in \mathcal{C}(\mathcal{S})/\mathcal{S}} |p|$, where $|p|$ is the weight of p.

Unitary operations and the stabilizer formalism

It turns out that, within the stabilizer formalism, it is more practical to compute the unitary evolution of the stabilizer group instead of the evolution of the quantum states [26]. Consider now a unitary operator U acting on the Hilbert space $\mathcal{H}$ and $|\psi\rangle \in \mathcal{L}(\mathcal{S})$. If this is true, then $\forall s \in \mathcal{S}$ we can observe that $U |\psi\rangle = Us |\psi\rangle = Us U^\dagger U |\psi\rangle$. From this trivial calculation it is immediate to see that, once we apply a unitary operation, the evolved code space $U\mathcal{L}(\mathcal{S})$ is stabilized by the group $U\mathcal{S}U^\dagger$ which consists of the stabilizer group conjugated under the action of U; hence the set of stabilizer generators s_j gets updated this way and the new stabilizer group is generated by the set $Us_j U^\dagger$.

1.2.1 Calderbank-Shor-Steane codes (CSS codes)

Here we introduce a way to compose the so-called CSS codes [32]. This class of codes is a particular sub-class of stabilizer codes and has been a game changer in the field: the toric code introduced by A.Kitaev in [19] is a perfect example of a CSS code. Nowadays to match some constraints imposed by the physical realizations of quantum computing platforms some variants of the toric code were introduced: like the unrotated surface code [3] and the rotated surface code [27]. These last codes are the de-facto standard for correcting errors on NISQ platforms.

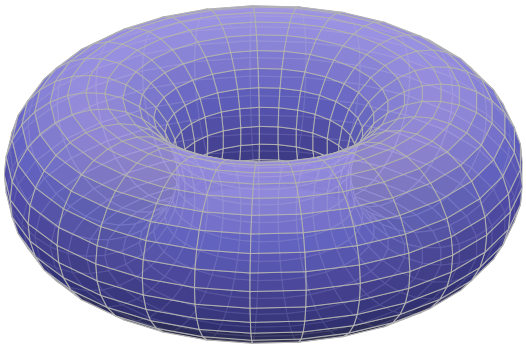
To build a CSS code we need two classical linear codes $\mathcal{C}_1$ (used to correct X errors) and $\mathcal{C}_2$ (used to correct Z errors) defined by their parity check matrix $\mathbf{H}_1$ and $\mathbf{H}_2$ (we refer the reader to A for a a very brief introduction on classical linear codes). Consider now r_j^1 as a row of the parity check matrix $\mathbf{H}_1$ and define the Z-type parity check operators as

$$s(Z)_j = \bigotimes_{i=1}^{n} Z_i^{r_j^1}. \tag{1.5}$$

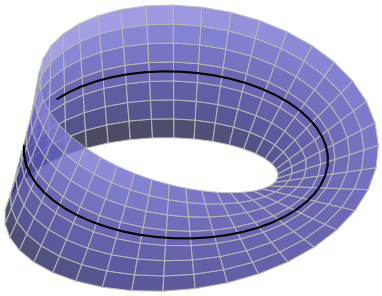
To make the code distance greater than one we can add X-type parity checks from the rows of the matrix $\mathbf{H}_2$ into the operators $s(X)_j$ built as before but substituting the exponent $r_j^1 \to r_j^2$. This condition is useful but it does not guarantee that the group is Abelian, and so we would violate the stabilizer formalism. To impose the hypothesis of the stabilizer formalism we can simply state as an axiom that the dual code of $\mathcal{C}_1$ must be contained in $\mathcal{C}_2$, hence $\mathcal{C}_1^{\perp} \subseteq \mathcal{C}_2$. If we define now the groups defined by the Z-type and X-type parity checks respectively as $\mathcal{S}_1(Z)$ and $\mathcal{S}_2(X)$ we can then define the stabilizer group of the CSS code as the group $\mathcal{S} = \langle \mathcal{S}_1(Z), \mathcal{S}_2(X) \rangle$ which is generated by the $2n - k_1 - k_2$ independent parity checks so that the quantum code encodes $k_1 + k_2 - n$ logical qubits. The distance of the quantum code will be the $\min(d(\mathcal{C}_1), d(\mathcal{C}_2))$.

1.3 Topological quantum error correction

Starting from the seminal works of A.Kitaev [19] and S.Bravyi [3], an important approach to QEC is based on locally defined parity checks on qubits fixed on a 2D lattice embedded in a certain manifold. Manifolds used for this purpose can be both orientable surfaces, like a 2-torus [19], or non-orientable surfaces, like the Möbius strip [33]. This approach is the so-called topological quantum error correction (tQEC).



(a) An orientable surface used in tQEC: a 2-torus with rectangular tessellation. Source code inspired by: Draw Torus with Tikz in LaTeX – with Equation on CodeSpeedy



(b) A non-orientable surface used in tQEC: a Möbius strip with rectangular tessellation. Source code inspired by: Moebius Strip using TikZ on StackExchange

The performance of these codes is usually studied by their threshold and their pseudo-threshold. We will refer to the threshold, inspiring from [8], as the physical error rate per measurement round at which it is no longer useful to enlarge the 2D array of qubits in terms of logical error rate. In other words it defines a splitting in two working regions: if the physical error rate is lower than the threshold then enlarging the 2D array lowers the logical error rate, while if the physical error rate is higher than the threshold the logical error rate increases with the dimension of the 2D array of qubits. To define the pseudo-threshold we need a definition of the linear size L of a topological code, which is the size of the 2D array of qubits such that $n = O(L^2)$ [32].

Definition 1.3.1 ([32]). *We define the* ***pseudo-threshold*** *of a topological code $\mathcal{T}$ with associated linear distance L as the critical physical error rate $p_c^{\mathcal{T}}(L)$ such that it is equal to the logical error rate of the code $\bar{p}_{\mathcal{T}}(p, L)$ and $\bar{p}_{\mathcal{T}}(p, L) < p_c^{\mathcal{T}}(L)$ $\forall p < p_c^{\mathcal{T}}(L)$ with p being the physical error rate.*

The idea of the pseudo-threshold is to capture an L dependent crossover point that defines when the code is helpful. For its L-dependency the pseudo-threshold contains more information compared to the asymptotic threshold $p_\infty^{\mathcal{T}} = \lim_{L\to+\infty} p_c^{\mathcal{T}}(L)$ [32].

1.3.1 Homological stabilizer codes

A way to build quantum error correcting codes with the philosophy of tQEC within the stabilizer formalism is to rely on topology and homology; we will refer to this approach as homological stabilizer code (HSC) formalism. We are going to develop a general framework that will be applied straightforwardly into the definition of the toric code. The contents of this subsection are inspired from [25], [32], [2] and [14].

Consider a manifold $\mathcal{M}$ and fix one possible tessellation $T(\mathcal{M})$ to obtain the sequence of i-simplices $\{T_i\}$, e.g. – for the toric code – vertices (0-simplices), edges (1-simplices) and faces (2-simplices). Associate now a vector space C_i over the field $\mathbb{F}_2$ with mod 2 sum as its binary operation for every element of the sequence of i-simplices, hence we obtain another sequence of vector spaces $\{C_i\}$. In these vector spaces it is possible to define an inner product between two vectors $a, b \in C_i$ – called i-chains – such that $\langle a, b\rangle = \sum_j a_j b_j$ mod 2. When this inner product is 0, the vectors a and b are said to be orthogonal. Once we defined the two sequences, we need a proper operator to obtain the boundaries of an element of C_i, e.g. given a 1-chain it has to extract its extreme vertices or given a 2-chain it has to extract the edges that define the border between the internal and external part of it.

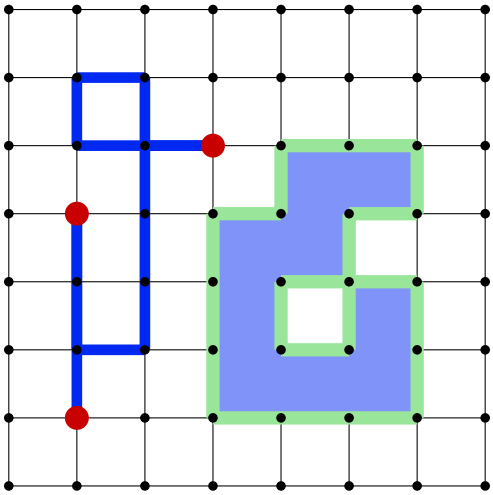


Figure 1.2: An 8×8 open boundary square tessellation where an example on how we want boundary extraction to act is showed. The green 1-chain represents the desired output we would like to obtain from the light blue 2-chain while the red 0-chain represents the desired output we would like to obtain from the dark blue 1-chain.

It turns out that these operators, called **boundary operators** are a sequence of homomorphisms $\partial_i : C_i \to C_{i-1}$ and they respect the property $\partial_i \circ \partial_{i+1} = 0\ \forall i$. Mathematically speaking, this property tells us that the image of the boundary operator ∂_{i+1} belongs to the kernel of the previous boundary operator ∂_i. It is common to refer to the elements of $\ker(\partial_i)$ as i-cycles. From now on we will denote the space of i-cycles – $\ker(\partial_i)$ – as Z_i and the space of i-boundaries – $\mathrm{Im}(\partial_{i+1})$ – as B_i. It is also immediate that not every i-cycle is an i-boundary and in literature i-boundaries are sometimes referred to as trivial

cycles because if one adds them to a non-trivial cycle one obtains another non-trivial cycle. Another interesting point to highlight is that from a linear algebra point of view, i-boundaries are not a complete set for the i-cycles space.
Now that both the sequences $\{C_i\}$ and $\{\partial_i\}$ are defined, we can introduce the so-called chain complex as follows:

Definition 1.3.2. *Consider the sequences of Abelian groups $\{C_i\}$ and boundary operators $\{\partial_i\}$. The sequence $\{C_i, \partial_i\}$ is called* ***chain complex****.*

For our purposes it is useful to express these definitions and concepts in the other direction too, hence there will be a sequence of homomorphisms called **coboundary operators** $\delta_i : C_i \to C_{i+1}$ with the property $\delta_{i+1} \circ \delta_i = 0 \; \forall i$.

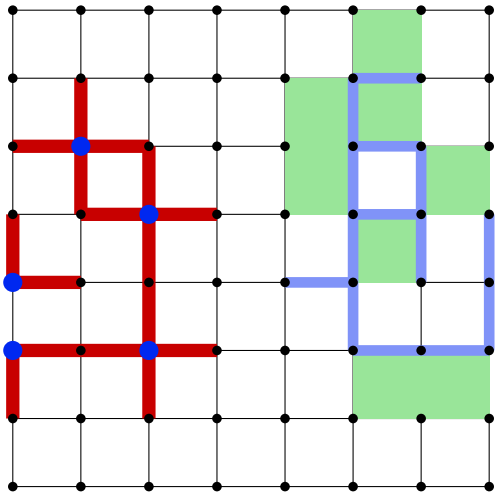


Figure 1.3: An 8×8 open boundary square tessellation where an example on how we want co-boundary extraction to act is showed. The green 2-chain represents the desired output we would like to obtain from the light blue 1-chain while the red 1-chain represents the desired output we would like to obtain from the dark blue 0-chain.

Definition 1.3.3. *Consider the sequences of Abelian groups $\{C_i\}$ and coboundary operators $\{\delta_i\}$. The sequence $\{C_i, \delta_i\}$ is called* ***cochain complex****.*

The elements of $\ker(\delta_i)$ and $\mathrm{Im}(\delta_{i-1})$ are referred to respectively as i-cocycles and i-coboundaries. As we did for the definition in the other direction, we define the space of i-cocycles $\ker(\delta_i)$ as Z^i and the space of i-coboundaries $\mathrm{Im}(\delta_{i-1})$ as B^i. To complete the mathematical background to describe the toric code we have to define the i-th homology and cohomology groups as follow.

Definition 1.3.4. *Consider the sequences of the spaces of i-cycles $\{Z_i\}$ and i-boundaries $\{B_i\}$ defined by the sequence of boundary operators $\{\partial_i\}$. We call the sequence of groups $\{(H_i, \oplus)\}$ as the sequence of* ***homology groups****, where H_i is the quotient set Z_i/B_i and $\oplus$ is the mod 2 sum. The group $(H_i, \oplus)$ is referred as the i-th homology group.*

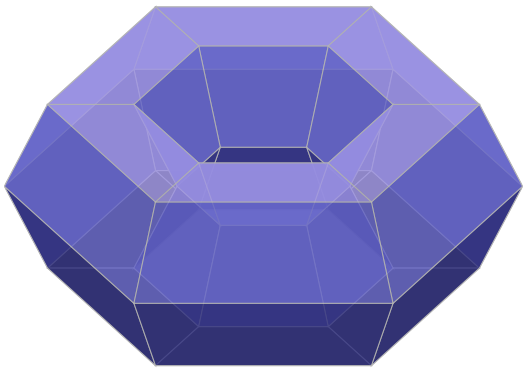
Definition 1.3.5. *Consider the sequences of the spaces of i-cocycles $\{Z^i\}$ and i-coboundaries $\{B^i\}$ defined by the sequence of boundary operators $\{\delta_i\}$. We call the sequence of groups $\{(H^i, \oplus)\}$ as the sequence of* ***cohomology groups****, where H^i is the quotient set Z^i/B^i and $\oplus$ is the mod 2 sum. The group $(H^i, \oplus)$ is referred as the i-th cohomology group.*

The group H_i (or H^i) can be seen as the equivalence classes of the i-cycles space Z_i (or the i-cocycles space Z^i) mod B_i (or mod B^i). In other words, let ζ and η be elements of Z_i (or Z^i); they belong to the same equivalence class if $\zeta + \eta = \beta$ with $\beta \in \text{Im}(\partial_{i+1})$ (or $\beta \in \text{Im}(\delta_{i-1})$). Regarding other useful properties that come out from topology, it can be proven that, given a manifold with genus g, where the genus can be identified as the number of holes of the manifold, then the number of logical qubits that can be encoded is $k = 2g$; this is due to the fact that the first homology and cohomology groups satisfy the isomorphism $H_1 \cong \mathbb{F}_2^{2g} \cong H^1$. This mathematical framework is fundamental for the construction of the stabilizer group and the logical operators of HSCs, like the toric code, but from now on we will need to decide what type of code we want to define.

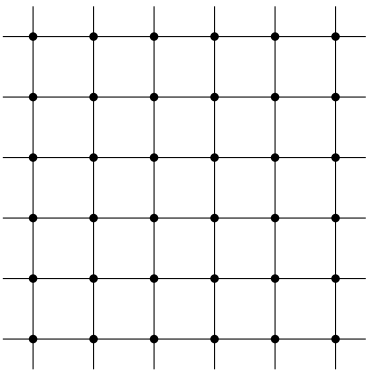
1.3.2 Toric code

The toric code is a milestone of QEC. Thanks to the protection offered by topological DOF it is possible to obtain very high performances and can be fully described as a CSS code within the HSC formalism. Before reading this section we suggest to look at Appendix C to align with the symplectic notation of Pauli operators. The contents of this chapter are inspired by [32] and[25].

Consider now the manifold $\mathcal{M}$ to be a 2-torus and apply to it an $L \times L$ square tessellation T. This tessellation creates L^2 vertices, $2L^2$ edges and L^2 faces and can be represented on an $L \times L$ square grid with periodic boundary conditions.



(a) A 2-torus with 6×6 rectangular tessellation. Source code inspired by: Draw Torus with Tikz in LaTeX – with Equation on CodeSpeedy.



(b) A 6×6 square tessellation with periodic boundary conditions imposed between both left-right sides and top-bottom sides of the grid.

Figure 1.4: Fig. 1.4a and Fig. 1.4b are equivalent from a topological point of view.

Being the 2-torus a 2-manifold, it leads to the definition of the bounded chain complex $C_0 \xleftarrow{\partial_1} C_1 \xleftarrow{\partial_2} C_2$ and the bounded cochain complex $C_0 \xrightarrow{\delta_0} C_1 \xrightarrow{\delta_1} C_2$ both made of three groups connected by two homomorphisms. Following the general construction discussed in section 1.3.1, we associate a qubit to each 1-simplex and define the elements of the stabilizer group of the toric code as the Z-operators associated to a basis of the 1-boundary space B_1 and the X-operators associated to a basis of the 1-coboundary space B^1. In the literature the basis for both B_1 and B^1 are usually obtained by applying the ∂_2 and δ_0 operators to the canonical basis of their respective domain. With this procedure we obtain a square made of 4 edges for each plaquette from the application of ∂_2 and a cross made of 4 edges for each vertex from the application of δ_0. We can clearly see that only $L^2 - 1$ of both the set of squares and crosses are linearly independent and they build a basis set

for their relative boundary or coboundary space. Once we obtained these vectors we can map them to our stabilizer group: for the set of squares we associate to each element a Z-operator $\mathcal{Z}_i$ acting on each of the four qubits on the sides of the square, while for the set of crosses we associate an X-operator $\mathcal{X}_j$ acting on each of the four qubits on the arms of the cross like in Fig. 1.5.

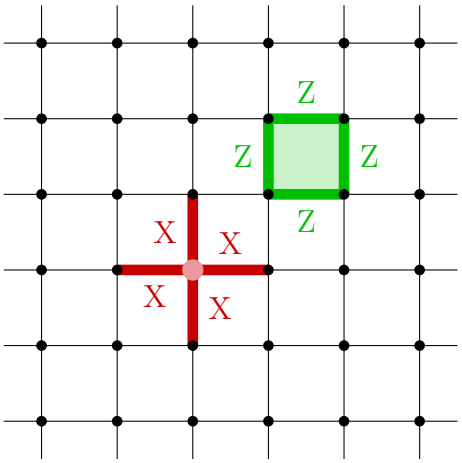


Figure 1.5: The figure shows how $\mathcal{Z}_i$ – the dark green square edges – and $\mathcal{X}_j$ – the dark red cross of edges – are respectively associated to a vector of the canonical basis of C_2 – the light green plaquette – and of C_0 – the light red vertex – via ∂_2 and δ_0 respectively and the choice of the Pauli operator to associate to the edges.

We can then express these concepts formally with the following definition:

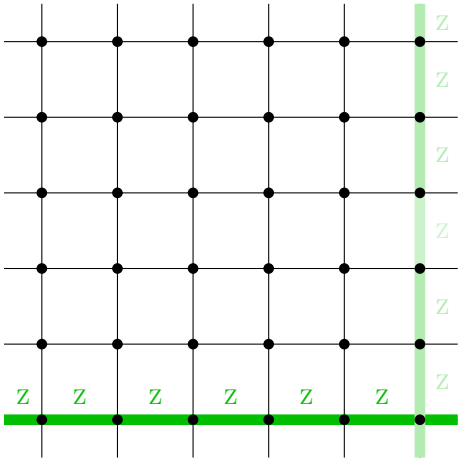
Definition 1.3.6. *Let the manifold $\mathcal{M}$ be a 2-torus and apply to it an $L \times L$ square tessellation T. Consider the bounded chain complex $\{C_i, \partial_i\}$ and the bounded cochain complex $\{C_i, \delta_i\}$ of the T-tessellated 2-torus $\mathcal{M}$. We define the* ***stabilizer group of the toric code*** *as the group $\mathcal{S} = \langle \{\mathcal{Z}_i\}, \{\mathcal{X}_j\} \rangle$ such that, in symplectic representation, $\mathcal{Z}_i = P(\mathbf{0}|s_i)$ and $\mathcal{X}_j = P(s^i|\mathbf{0})$, where $\{s_i\}$ and $\{s^i\}$ are two basis of the 1-boundary vector space C_1 respectively obtained applying ∂_2 to the canonical basis of C_2 – $\{e_i^2\}$ – and δ_0 to the canonical basis of C_0 – $\{e_i^0\}$– discarding then the linearly dependent vectors.*

Being the generators of $\mathcal{S}$ only Pauli operators, we know that $\mathcal{S}$ is a subgroup of the Pauli group $\mathcal{G}_n$ and that $-\mathbb{1} \notin \mathcal{S}$. To complete the compatibility check with the stabilizer formalism we can verify that $\mathcal{S}$ is an Abelian subgroup of the Pauli group by noticing that square-form Z-operators commute with cross-form X-operators because they touch each other with zero or two edges, so the symplectic inner product between their symplectic vectors is zero.

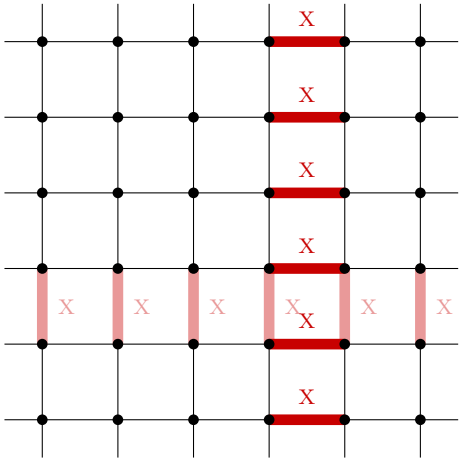
From this definition we can also highlight some other aspects about the stabilizer group. In our system we know that, having $2L^2$ edges, there are $n = 2L^2$ physical qubits while the dimension of the stabilizer group is $n - k = 2L^2 - 2$. These dimensions imply that the code represents $k = 2$ logical qubits and this result is compatible with the genus dependent relation $k = 2g$, with $g = 1$ for the 2-torus, introduced before. This observation is valid because the genus is the only topological invariant that characterize a 2-torus.

Now that the stabilizer group is defined and we understood that we can represent two logical qubits, we have to define the logical observables. The key to define these operators

lies in the first homology group H_1 and the first cohomology group H^1. Let's start by defining the Z operators of the toric code; from H_1 we observe there are 4 equivalence classes: the identity operator mod B_1, a meridian line made of horizontal edges mod B_1, a longitude line made of vertical edges mod B_1 and the composition of the previous two (an oblique chain mod B_1). It is clear that this quotient group is generated by two elements: we choose them to be a meridian line and a longitudinal line as it is normally done in conventional literature. This observation also finds a theoretical confirmation from the existing isomorphism between H_1 and $\mathbb{F}_2^2$. We take now the vectors associated to the two generators of the first homology group and define the logical Z-operators $\tilde{\mathcal{Z}}_0$ and $\tilde{\mathcal{Z}}_1$ of the toric code as the operators constructed with Z operators acting on the edges that belong to a meridian line ($\tilde{\mathcal{Z}}_0$) and to a longitude line ($\tilde{\mathcal{Z}}_1$). We can do exactly the same procedure for the logical X-operators $\tilde{\mathcal{X}}_0$ and $\tilde{\mathcal{X}}_1$ with the first cohomology group and find out that the two logical operators are the X-operators associated to a meridian line made of vertical edges ($\tilde{\mathcal{X}}_1$) and a longitude line made of horizontal edges ($\tilde{\mathcal{X}}_0$) built the same way we did for the logical Z-operators. The indexes are defined to assure that $\{\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i\} = 0$ and $[\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_{\bar{i}}] = 0$; it is trivial to observe that the Z-operator associated to the meridian line is coupled with the X-operator associated to the longitude line and vice versa. A graphical representation on how to obtain these logical operators can be found in the following figure.



(a) The dark green 1-chain of Z operators represents the logical operator $\tilde{\mathcal{Z}}_0$ while the light green 1-chain of Z operators represents the logical operator $\tilde{\mathcal{Z}}_1$



(b) The dark red 1-chain of X operators represents the logical operator $\tilde{\mathcal{X}}_0$ while the light red 1-chain of X operators represents the logical operator $\tilde{\mathcal{X}}_1$

Figure 1.6: An example on how to construct and couple the four logical operators: $\tilde{\mathcal{Z}}_0$ is coupled with $\tilde{\mathcal{X}}_0$ while $\tilde{\mathcal{Z}}_1$ is coupled with $\tilde{\mathcal{X}}_1$. This pairing is induced by the commutation relations these operators must respect and it can be easily checked that two operators with different associated Pauli type anticommute if their respective 1-chain share an odd number of edges while they commute if they share an even number of edges.

From these considerations we can then express the following definition:

Definition 1.3.7. *Let the manifold $\mathcal{M}$ be a 2-torus and apply to it an $L \times L$ square tessellation T. Consider $\{h_0, h_1\}$ and $\{h^0, h^1\}$ as a set of generating vectors of the first homology group H_1 and of the first cohomology group H^1 of the T-tessellated 2-torus $\mathcal{M}$. We define the* ***logical operators of the toric code*** *as the canonical Pauli operator couples written within the symplectic representation: $\left(\tilde{\mathcal{Z}}_0 = P\left(\mathbf{0}|h_i\right), \tilde{\mathcal{X}}_0 = P\left(h^j|\mathbf{0}\right)\right)$ and $\left(\tilde{\mathcal{Z}}_1 = P\left(\mathbf{0}|h_{\bar{i}}\right), \tilde{\mathcal{X}}_1 = P\left(h^{\bar{j}}|\mathbf{0}\right)\right)$ where the generating vectors are coupled such that $\langle h_i, h^j\rangle = \langle h_{\bar{i}}, h^{\bar{j}}\rangle = 1$ with $i, j = 0 \vee 1$.*

The last definition, is based on the fact that if two basis vectors are not orthogonal, then their corresponding Pauli operators do not commute. This consideration is trivial if the operators to check are an X-only operator and a Z-only operator thanks to symplectic representation of Pauli operators.

1.4 Subsystem stabilizer codes

Subsystem codes – introduced in [28] – are a generalization of stabilizer codes. These codes do not guarantee better performances compared to stabilizer codes and they are built removing some of the logical qubits encoded via a stabilizer group. The advantage of these codes is mainly that sometimes we can get lower weight operators to measure compared to their stabilizer version. Let's start to formalize them by defining the so-called gauge group:

Definition 1.4.1 ([32])**.** *Consider now a stabilizer group $\mathcal{S} = \langle s_1, s_{n-k}\rangle$ with associated pairs of logical operators $\{(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i)_{i=1,\dots,k}\}$. Consider a subset of the logical couples $\{(\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i)_{i\in\mathbb{G}}\}$ with $\mathbb{G} \subseteq \{1, ..., k\}$; we refer to them as the logical operators of the gauge qubits. We define now the* ***gauge group*** *as $\mathcal{G} = \langle \mathcal{S}, \{\tilde{\mathcal{X}}_i, \tilde{\mathcal{Z}}_i\}_{i\in\mathbb{G}}\rangle$.*

First of all we can observe that the gauge group is non-Abelian and so it is immediate that the stabilizer group associated to a gauge group is the center of the gauge group itself. Mathematically [32]:

$$\mathcal{S} = \mathcal{C}(\mathcal{G}) = \mathcal{C}_{\mathcal{G}}(\mathcal{G}) = \{\ell \in \mathcal{G} \mid \forall g \in \mathcal{G}, [\ell, g] = 0\}. \tag{1.6}$$

Because of the gauge group being a non-Abelian group, it is not trivial how to measure the operators of the gauge group (all the degrees of freedom of the group cannot be fixed). These codes, removing $|\mathbb{G}|$ degrees of freedoms, can represent $k - |\mathbb{G}|$ logical qubits and these logical pairs commute with $\mathcal{G}$, hence they are uneffected by measuring elements of $\mathcal{G}$. The logical operators of a subsystem code – like logical operators of stabilizer codes – are not unique because they can be multiplied by elements of $\mathcal{G}$; if they are multiplied only by elements of $\mathcal{S}$ they are referred to as bare logical operators, while if multiplied by at least a logical operator of a gauge qubit they are referred to as dressed logical operators. The difference between the stabilizer counterpart is that all logical operators of stabilizer codes cannot be dressed because all the representable qubits contain information. The multiplication by a gauge qubit logical operator is allowed because it just changes the state of the gauge qubits, and we are not interested in them. The possibility of choosing the logical operators we prefer can lead to different kinds of optimizations. The choice of the logical operators imposes also the need of a new definition for the distance of a subsystem code as $d = \min_{\ell\in\mathcal{C}(\mathcal{S})\setminus\mathcal{G}} |\ell|$.

1.5 Floquet codes

Floquet codes are a new class of quantum error correcting codes introduced by Haah and Hastings in [13]. The key idea behind these codes is that their stabilizer group and the logical operators of the code are time dependent. This feature is possible because the measured operators in different rounds of measurements may not commute.

To define these codes, first of all consider a gauge group $\mathcal{G} = \langle g_1, ..., g_{|\mathcal{G}|} \rangle$ and define a set of possible subsets of $\mathcal{G}$ named $\mathcal{M} = \{\mathcal{M}_i\}_{i\in\{1,...,P\}}$ with $\mathcal{M}_i = \{m_i^j\}_{j\in\{1,...,|\mathcal{M}_i|\}}$. Floquet codes are defined by a P-periodic sequence of measurements such that at time t we mesaure the subset of operators contained in $\mathcal{M}_{t \bmod \mathrm{P}}$. From the measurement at round t it is possible to obtain a sequence of syndromes associated to the operators $m^j_{t \bmod \mathrm{P}}$ denoted as $\boldsymbol{\beta}(\mathcal{M}_{t \bmod \mathrm{P}}, t) = \{b_i(m_i^j, t)\}_{j=1,...,|\mathcal{M}_{t \bmod \mathrm{P}}|}$ with $b_i(m_i^j, t) \in \{\pm 1\}$. Once we have a set of syndromes we indicate with $\boldsymbol{\beta}(\mathcal{M}_{t \bmod \mathrm{P}}, t) \cdot \mathcal{M}_{t \bmod \mathrm{P}}$ the set of operators associated to the t-th measurement round multiplied by the sign of the associated syndrome. As we anticipated, this periodicity of measurements leads to the possibility of having a time-varying stabilizer group and in this context the stabilizer group is instead called the **istantaneous stabilizer group** (ISG). This time dependency of the stabilizer group will imply that the logical operators might be time varying too and they will not always be trivial to find. To obtain them we will consider the specific cases analysed.

1.5.1 Honeycomb code

The honeycomb code is the first example of a Floquet code. It is defined by weight-2 parity checks and it is a candidate to challenge the "supremacy" of the toric (surface) code in modern QEC.

Consider now a manifold $\mathcal{T}$ to be a 2-torus and apply to it an hexagonal tessellation, then associate a qubit for each vertex of the tessellation like in Fig. 1.7.

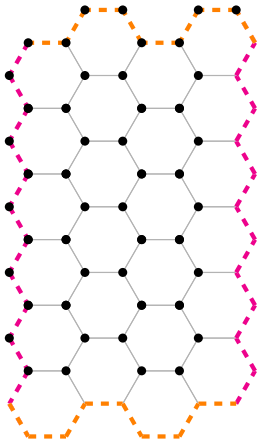


Figure 1.7: An hexagonal tessellation of the 2-torus with qubits on the vertices is represented. It produces n_p faces, $3n_p$ edges and $2n_p$ vertices. If we connect the upper and lower orange sides and the left and right magenta sides we get a 2-torus. Note that this tessellation is not the same used in HSC 1.3.1 where qubits were on the edges.

Once we have defined the tessellation of the torus we now index the plaquettes as type-i plaquettes with $i \in \{0, 1, 2\}$ such that a plaquette of type j cannot touch another plaquette of the same type. This is explained graphically in fig. 1.8.

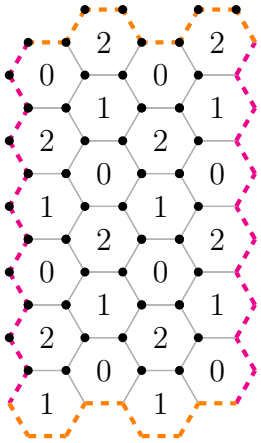
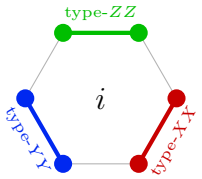
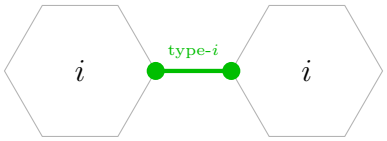


Figure 1.8: An hex-tessellated 2-torus with qubits on the vertices and indexed plaquettes.

We now want to associate two possible indices to each edge: the first index is defined by the direction of the 1-simplex and defines an operator to apply to the two qubits that are boundaries of the edge, while the second is defined by the type of plaquettes the edge connects. These definitions are represented graphically in Fig. 1.9.



(a) How we associate to edges a type of weight-2 Pauli operator based on their direction.



(b) How the plaquette dependent index of the edges is defined. In the example we construct a type-i and type-ZZ check.

Figure 1.9: Following the procedure of the two upper figures we can obtain the edges that define $\mathcal{M}_i$ sets for the measurement rounds of the honeycomb code.

Now that all the edges are indexed, we can define the set $\mathcal{M} = \{\mathcal{M}_i\}_{i\in\{0,1,2\}}$ of the honeycomb code. The element $\mathcal{M}_i$ of $\mathcal{M}$ consists in all the checks associated to the edges that connect type-i plaquettes, hence in the periodic cycle of measurements there will be: type-0 edges, type-1 edges and then type-2 edges. We will denote the elements of $\mathcal{M}_i$ as $m_i^j = m(e_i^j)$, where e_i^j is the j-th edge of type i and $m(\cdot)$ maps the edge to its corresponding weight-2 Pauli operator. The last objects we need to introduce to define the ISG are the plaquette operators and how to compute the sign associated to them. Suppose we are at round t, and we measured type-i checks, hence at time $t-1$ we measured type-j checks with $j = (i-1) \mod 3$. We now define the signed plaquette operator $\mathcal{P}_k^r$ associated to the k-th type-r plaquette P_k^r as:

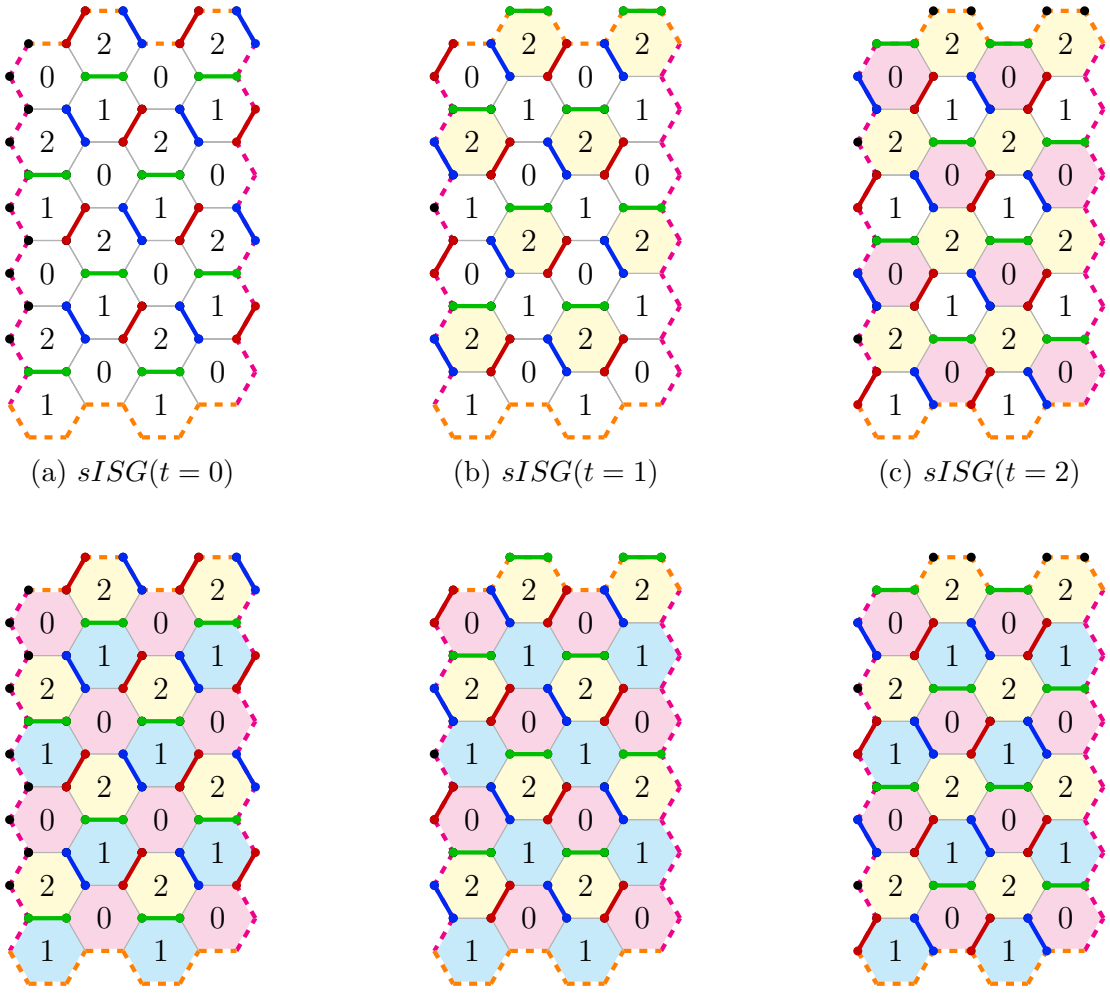
$$\mathcal{P}_k^r = \prod_{e_i^\ell \in \mathcal{M}_i \cap \partial_2 P_k^r} \prod_{e_j^q \in \mathcal{M}_j \cap \partial_2 P_k^r} b(m(e_i^\ell), t)\, b(m(e_j^q), t-1) m(e_i^\ell) m(e_j^q) \tag{1.7}$$

With $r = (i-2) \mod 3$. Note that the plaquette operator is not indicated as a time dependent quantity: that is because every plaquette operator commutes with all the weight-2 checks measured during the cycles of the honeycomb code, hence their associated sign remain constant. With these quantities we can now define the ISG as:

Definition 1.5.1. *Let the manifold $\mathcal{T}$ be a 2-torus and apply to it an hexagonal tessellation H. Consider as the defining sets of the 3-periodic measurement cycle of the honeycomb code the operators associated to the elements of $\mathcal{M} = \{\mathcal{M}_i\}_{i\in\{0,1,2\}}$ where $\mathcal{M}_i$ contains all the edges that connects type-i plaquettes. Then the* ***ISG of the honeycomb code*** *is defined by the following procedure:*

1. *Set:* $ISG(t=-1) = \{\emptyset\}$
2. *First update:* $ISG(t=0) = \boldsymbol{\beta}(\mathcal{M}_{t=0}, t=0)\cdot\mathcal{M}_0$
3. *Second update:* $ISG(t=1) = \boldsymbol{\beta}(\mathcal{M}_{t=1}, t=1)\cdot\mathcal{M}_1 \cup \{\mathcal{P}_k^2\}_{k\in\{1,\dots,\frac{n_p}{3}\}}$
4. *Third update:* $ISG(t=2) = \boldsymbol{\beta}(\mathcal{M}_{t=2}, t=2)\cdot\mathcal{M}_2 \cup \{\mathcal{P}_k^r\}_{k\in\{1,\dots,\frac{n_p}{3}\},r\in\{0,2\}}$
5. *Following updates:* $ISG(t\geq 3) = \{\boldsymbol{\beta}(\mathcal{M}_{t\bmod 3}, t)\cdot\mathcal{M}_{t\bmod 3}\} \cup \{\mathcal{P}_k^r\}_{k\in\{1,\dots,\frac{n_p}{3}\},r\in\{0,1,2\}}$

From the stabilizer group we defined it is immediate that after $t=3$ we get to a steady state condition for the ISG up to signs or, graphically:



(a) $sISG(t=0)$ (b) $sISG(t=1)$ (c) $sISG(t=2)$

(d) $sISG(t=0\bmod 3\wedge\geq 3)$ (e) $sISG(t=1\bmod 3\wedge\geq 3)$ (f) $sISG(t=2\bmod 3\wedge\geq 3)$

Figure 1.10: The evolution of the sign-less ISG (sISG) is displayed. Coloured edges and plaquettes belong to the ISG. Edges are coloured based on their operator type and plaquettes following their index: $\{\mathcal{P}_k^0\}$ are magenta, $\{\mathcal{P}_k^1\}$s are cyan and $\{\mathcal{P}_k^2\}$ are yellow.

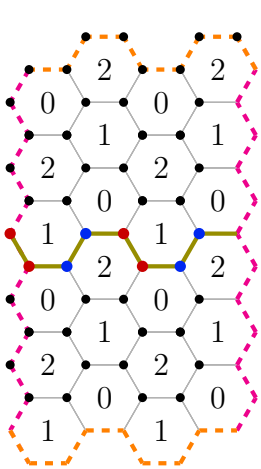
The last informations we need to figure out is: how many logical qubits in the "steady state condition" can this code represent? And how can we construct the logical operators associated to these logical qubits?

The number of qubits this code can represent can be found computing the dimension of the ISG for $t \geq 3$. The ISG in this time region is generated by $n_p - 1$ plaquettes and $n_p - 1$ edges: one plaquette can be discarded because the product of all the plaquettes is the identity and one edge can be discarded too because if we compute the product between all the type-i plaquettes and the type-i edges we obtain the identity, hence $\dim(ISG(t \geq 3)) = 2n_p - 2$. This result implies that we are fixing $2n_p - 2$ degrees of freedom out of the $2n_p$ available degrees of freedom of our system (we have $2n_p$ vertices, hence $2n_p$ physical qubits). The number of logical qubits of the honeycomb code is then 2.

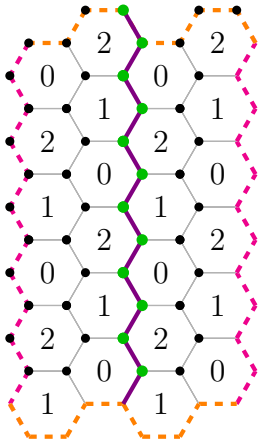
Obtaining these logical operators is the hardest part: they have to commute with all the possible elements of the ISG at a fixed time and they have to be paired in conjugate couples. It turns out that we can divide them in two categories named inner logical operators and outer logical operators. These names will be clarified in the next paragraph: take them as just a nomenclature fact for now. Inner logical operators turn out to be static operators while outer logical operators are dynamical and they need to be updated after every measurement round. Another peculiarity of outer logical operators is that (up to a sign) they are periodic with period-6 instead of what we could intuitively guessed to be periodic with period-3. For the definition of logical operators we will consider one of the possible definitions as follow:

Definition 1.5.2. *Let the manifold $\mathcal{T}$ be a 2-torus and apply to it an hexagonal tessellation H. Consider $\{h_0, h_1\}$ as a set of generators of the first homology group H_1 of the H-tessellated 2-torus $\mathcal{T}$. The* ***inner logical operators of the honeycomb code*** *$\mathcal{I}_0$ and $\mathcal{I}_1$ are respectively defined as the product of the operators associated to the edges in the 1-chains h_0 and h_1.*

Graphycally, we can see them represented in Fig. 1.11 (choosing h_0 and h_1 as a meridian and a longitude respectively) where green vertices indicate a Z operator, red vertices indicate an X operator and blue vertices indicate a Y operator.



(a) Inner logical operator $\mathcal{I}_0$.



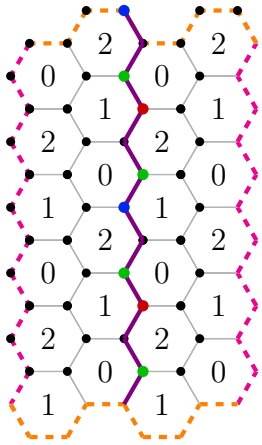
(b) Inner logical operator $\mathcal{I}_1$.

Figure 1.11: A graphical construction of the inner logical operators of the honeycomb code.

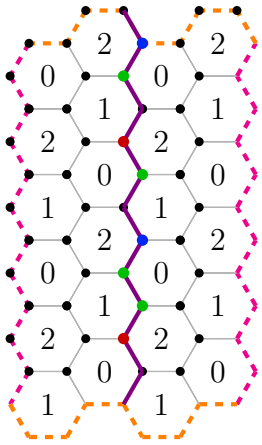
Regarding outer logical operators, a way to find them is to consider the non-trivial 1-chains used to define the inner logical operators and choose properly the operators to apply on the chain of edges. Keep in mind that this is not the only way to obtain them and the choice of mantaining the same 1-chains used for the inner logical operators is purely a convention we are adopting. There are for sure more general manners to express this definition but we are sticking to the following graphical method to deliever an operative definition to implement the honeycomb code with a programming language. The procedure starts from the operators that lie in the selected non-trivial chain in Figs. 1.12a and 1.12g that commute with $ISG(t = 0 \bmod 3)$ and anticommutes with the inner logical operator associated to the other 1-chain, then multiply it by the checks measured in the current round whose defining edge is on the 1-chain we are building the operator on and by the sign of their respective syndrome just like we did for plaquette operators.

$$\mathcal{O}_k(t \geq 3, h_{\bar{k}}) = \mathcal{O}_k(t = 3, h_{\bar{k}}) \prod_{r \in \{3,\dots,t-1\}} \prod_{e^j_{r \bmod 3} \in \mathcal{M}_{r \bmod 3} \cap h_{\bar{k}}} b(m(e^j_{r \bmod 3}), r) m(e^j_{r \bmod 3}) \tag{1.8}$$

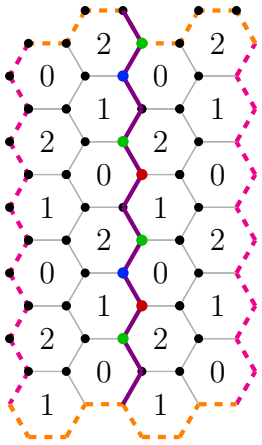
where $\mathcal{O}_k(t = 3, h_{\bar{k}})$ is the starting set of operators. We obtain then the following 6-rounds period (without counting the signs) for two possible **outer logical operators** named respectively $\mathcal{O}_0$ and $\mathcal{O}_1$ such that $\{\mathcal{O}_j, \mathcal{I}_j\} = 0$, $[\mathcal{O}_j, \mathcal{O}_k] = [\mathcal{I}_j, \mathcal{I}_k] = [\mathcal{O}_j, \mathcal{I}_{\bar{j}}] = 0$. In Fig. 1.12 you can find a graphical representation.



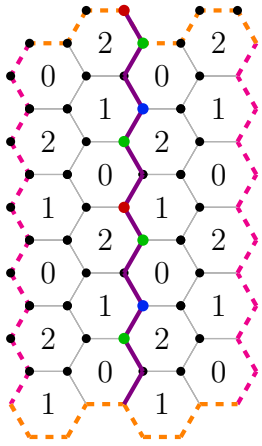
(a) $\mathcal{O}_0(t = 0)$



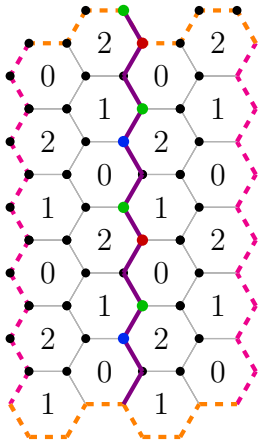
(b) $\mathcal{O}_0(t = 1)$



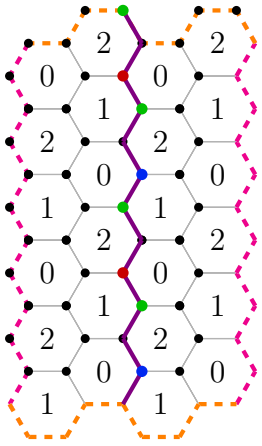
(c) $\mathcal{O}_0(t = 2)$



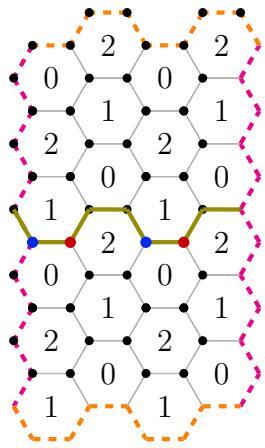
(d) $\mathcal{O}_0(t = 0 \bmod 3 \wedge \geq 3)$



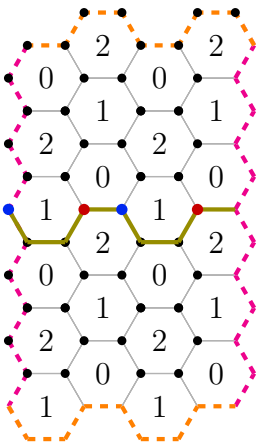
(e) $\mathcal{O}_0(t = 1 \bmod 3 \wedge \geq 3)$



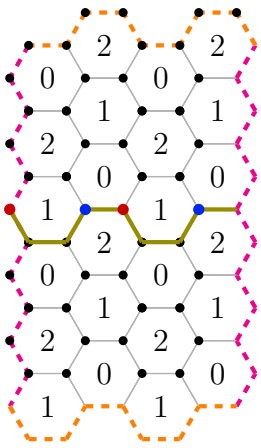
(f) $\mathcal{O}_0(t = 2 \bmod 3 \wedge \geq 3)$



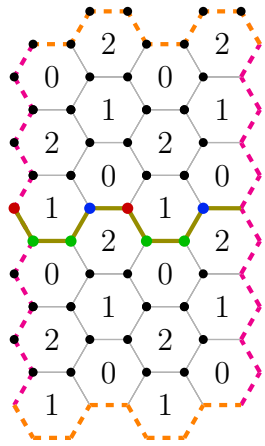
(g) $\mathcal{O}_1(t=0)$



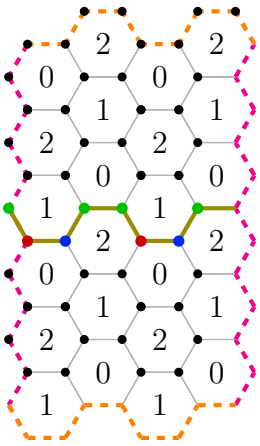
(h) $\mathcal{O}_1(t=1)$



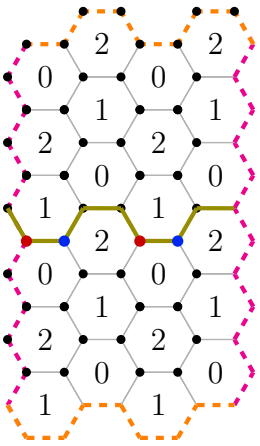
(i) $\mathcal{O}_1(t=2)$



(j) $O_1(t=\mathcal{O} \bmod 3 \wedge \geq 3)$



(k) $\mathcal{O}_1(t=1 \bmod 3 \wedge \geq 3)$



(l) $\mathcal{O}_1(t=2 \bmod 3 \wedge \geq 3)$

Figure 1.12: From (a) to (f) and (g) to (l) are represented respectively the evolution of $\mathcal{O}_0$ and $\mathcal{O}_1$. These operators can be extended to bigger tessellation repeating the patterns.

The honeycomb code as a subsystem code

The operators that appear in the measurement cycles of the honeycomb code can also be used to define a subsystem code. If we define the gauge group $\mathcal{G}$ as the subgroup of the Pauli group generated by the checks of type-0, type-1 and type-2, then we know that $\dim(\mathcal{G}) = 3n_p - 1$ because the product of all the checks gives the identity.

Given this definition of the gauge group, his internal stabilizer group can be generated by $n_p - 1$ plaquettes and two independent generators of the first homology group H_1, hence $\dim \mathcal{S} = n_p + 1$. This implies that the number of gauge qubits of the code is $n_{\mathcal{G}_q} = (\dim(\mathcal{G}) - \dim(\mathcal{S}))/2 = n_p - 1$. These dimensions tell us that the code has zero logical qubits, because $n_p + 1$ degrees of freedom are fixed by the stabilizer group and $n_p - 1$ degrees of freedom are fixed by the gauge qubits, hence our system of $2n_p$ physical qubits has zero logical qubits in it if we encode the information with this subsystem code.

We can now explain why in the previous paragraph the fixed logical operators are called inner logical operators: they are in the stabilizer group of the associated subsystem code, while outer logical operators are not.

Ladder code

The ladder code, also proposed by Haah and Hastings in [13], is a toy model that represents a single dynamical logical qubit. The code lives in a $2 \times L$ ladder with periodic boundary conditions on the rungs (the short side). We associate to vertical rungs a ZZ check and to horizontal edges associated to the same plaquette either XX or YY checks.

The measurement cycle is the following:

1. Measure vertical ZZ checks, hence $\mathcal{M}_0 = \{(ZZ)_{i=1,\dots,L}\}$.
2. Measure horizontal XX checks, hence $\mathcal{M}_1 = \{(XX)_{i=1,\dots,L}\}$.
3. Measure vertical ZZ checks, hence $\mathcal{M}_2 = \{(ZZ)_{i=1,\dots,L}\}$.
4. Measure horizontal YY checks, hence $\mathcal{M}_3 = \{(YY)_{i=1,\dots,L}\}$.

where the i index is referred to the L possible weight-2 operators of the chosen Pauli type.

Here the steady state sign-less stabilizer group is generated by the plaquettes built with the checks and the last measured set of checks; exactly like in the honeycomb code.

Regarding the number of logical qubits, we have $2L$ physical qubits in our system and we have to understand how many degrees of freedom are absorbed by the stabilizer group in every round. During ZZ-measurement rounds the stabilizer group is generated by $2L-1$ elements because the product of $XXXX$ plaquettes and the ZZ checks gives the product of the $YYYY$ faces, hence multiplying everything together we obtain the identity, while during XX (YY)-measurement rounds the product of $XXXX$ ($YYYY$) plaquettes and XX (YY) checks gives the identity, hence the stabilizer group is always generated by $2L-1$ elements. This proves the existence of a logical qubit because $2L-1$ out of $2L$ degrees of freedom are fixed.

The sign-less logical operators for this qubit are then defined as follow:

- The inner logical operator is non-trivial string along a rail of the ladder made of Z operators.
- The outer logical operator is a vertical check and his dynamics during the rounds is the following:
    1. t=0 mod 4: An XX or YY check over a vertical rung
    2. t=1 mod 4: An XX check over a vertical rung
    3. t=2 mod 4: An XX or YY check over a vertical rung
    4. t=3 mod 4: A YY check over a vertical rung

Chapter 2

Circuit Quantum Electrodynamics

In 1962, a macroscopical quantum effect has been predicted by Brian Josephson [18] via a Cooper pair tunneling mechanism in a junction, later called Josephson junction, made of three layers: superconductor - oxide - superconductor. This particular phenomenon granted Josephson the 1973 Nobel Prize in Physics "for his theoretical predictions of the properties of a supercurrent through a tunnel barrier, in particular those phenomena which are generally known as the Josephson effects" and was later revealed to be a key mechanism for the design of superconducting qubits; this drove to the "creation" of an entire new branch of research in physics: circuit quantum electrodynamics (cQED). Recently, another Nobel Prize related to cQED has been awarded: John Clarke, Michel H. Devoret and John M. Martinis won the 2025 Nobel Prize in Physics "for the discovery of macroscopic quantum mechanical tunnelling and energy quantisation in an electric circuit": this second award highlights even more how groundbreaking this branch of physics has been from both the theoretical aspect and the technological point of view.

This chapter contains some of the basics of cQED and it is based on the knowledge acquired during the course "Dispositivi, circuiti ed algoritmi per il calcolo quantistico M" held at the University of Bologna, the review by Blais et al. [1] and the review by Krantz et al. [23] whenever another reference is not cited specifically. The reader can also consult the lecture notes by Ciani et al. [6] for a more pedagogical approach.

2.1 Quantization of the LC resonator

Analysing an LC resonator starting from the point of view of Lagrangian mechanics and quantizing the canonical variables into their corresponding operators, we can map the quantum LC resonator problem into a quantum linear harmonic oscillator (also referred to as the simple harmonic oscillator). We suggest the reader to check [29] for an explanation on the quantum harmonic oscillator with the second quantization formalism before starting this section. from this theoretical procedure we can then predict energy quantization in electrical circuits.

Consider now an LC resonator with a capacitor C_r and inductor L_r:
Characterized by the following equations:

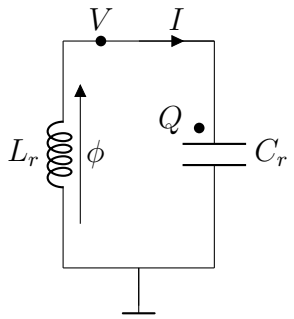


Figure 2.1: Schematic of the LC resonator

$$\begin{cases} V = \dot{\phi} \\ I = \dot{Q} \\ \phi = -L_r I \\ Q = C_r V \end{cases} \tag{2.1}$$

Our goal is to get the Hamiltonian function of this circuit; to obtain it we have to start from the Lagrangian function, which I remark to be defined as:

$$\mathcal{L}(q, \dot{q}) = T(\dot{q}) - U(q) \tag{2.2}$$

where q are the generalized coordinates, T is the kinetic energy and U is the potential energy. By arbitrary choice we associate the kinetic term to the energy stored in the electric field of the capacitor and the potential energy to the energy stored in the magnetic field of the inductor. Applying this particular choice together with the fundamental equation of the LC resonator we obtain the following flux dependent equations:

$$\begin{cases} T \to \frac{1}{2} C_r V^2 = \frac{1}{2} C_r \dot{\phi}^2 \\ U \to \frac{1}{2} L_r I^2 = \frac{1}{2} \frac{\phi^2}{L_r} \end{cases} \tag{2.3}$$

Hence, choosing ϕ as our generalized coordinate we can write the Lagrangian of the system:

$$\mathcal{L}(\phi, \dot{\phi}) = \frac{1}{2} C_r \dot{\phi}^2 - \frac{1}{2} \frac{\phi^2}{L_r} \,. \tag{2.4}$$

From the Lagrangian function we can obtain the conjugate momentum of the generalized coordinate ϕ. It turns out that it can be identified as the stored charge in the capacitor:

$$p = \frac{\partial \mathcal{L}}{\partial \dot{q}} \to \frac{\partial \mathcal{L}}{\partial \dot{\phi}} = C_r \dot{\phi} = C_r V = Q \,. \tag{2.5}$$

Applying now the Legendre transformation we can obtain the Hamiltonian function as:

$$H(q, p) = p\dot{q} - \mathcal{L}(q, \dot{q}) \to H(\phi, Q) = Q\dot{\phi} - \mathcal{L}(\phi, \dot{\phi}) = \frac{1}{2} C_r \dot{\phi}^2 + \frac{1}{2} \frac{\phi^2}{L_r} = \frac{1}{2} \frac{Q^2}{C_r} + \frac{1}{2} \frac{\phi^2}{L_r} \,. \tag{2.6}$$

We can then quantize the Hamiltonian and introduce the canonical commutation relation such that:

$$\begin{cases} \hat{H} = \frac{1}{2}\frac{\hat{Q}^2}{C_r} + \frac{1}{2}\frac{\hat{\phi}^2}{L_r} \\ [\hat{\phi}, \hat{Q}] = i\hbar \end{cases} \tag{2.7}$$

From now on we will drop the hats on the operators to have a simpler notation. From the Hamiltonian operator of equation 2.7, we can do a mapping to the Hamiltonian of the simple Harmonic oscillator:

$$H = \frac{p^2}{2m} + \frac{m\omega q^2}{2} \tag{2.8}$$

such that:

$$q \to \phi, \quad p \to Q \quad m \to C_r, \quad \omega = \frac{1}{\sqrt{L_r C_r}} \equiv \omega_r$$

From this mapping we can also define our Hamiltonian in second quantization formalism redefining the canonical couple of operators with creation and annihilation operators $a^\dagger$ and a:

$$\begin{cases} \phi = \sqrt{\frac{\hbar}{2\omega_r C_r}}(a + a^\dagger) \\ Q = -i\sqrt{\frac{\hbar\omega_r C_r}{2}}(a - a^\dagger) \end{cases} \tag{2.9}$$

such that $[a, a^\dagger] = 1$.
The second quantized Hamiltonian is then:

$$H = \hbar\omega_r \left(a^\dagger a + \frac{1}{2}\right) . \tag{2.10}$$

The eigenvalue equation of the Schrödinger problem associated to this Hamiltonian is trivially solved. That's because $n = a^\dagger a$ is a well known object called number operator. It is possible to prove that its eigenvalues are only non-negative integers $Z_+ \cup \{0\}$, hence we can index its eigenstates as follows:

$$n\,|m\rangle = m\,|m\rangle\,, \qquad m \in Z_+ \cup \{0\}\,. \tag{2.11}$$

The eigenvalues corresponding to the energies of the quantum LC resonator are then:

$$E_m = \hbar\omega_r \left(m + \frac{1}{2}\right), \qquad m \in Z_+ \cup \{0\} \tag{2.12}$$

where the spacing between two consecutive energy levels is constantly equal to $\hbar\omega_r$.

2.2 Superconducting qubits

To build a qubit where the state is encoded via energy levels, we do not want the energy splitting between two levels to be constant all over the spectrum, hence we do not want a simple harmonic oscillator. This necessity is due to the fact that, encoding the qubit states in the first two energy levels, it would be very hard to control the system and isolate the first two states from the higher energy states. To obtain the desired spectrum we need to introduce a non-linearity and a possible solution for this problem can be obtained using a Josephson junction.

2.2.1 The Josephson junction

A Josephson junction is a superconductor-oxide-superconductor structure where an interaction via tunneling current between the Cooper pair ground states of the two superconducting sides can happen.
The expressions for the tunneling current and the voltage difference read:

$$\begin{cases} I = I_c \sin(\varphi(t)) \\ \dot{\phi}(t) = V, \qquad \varphi(t) = \frac{2\pi}{\phi_0}\phi(t) \mod 2\pi \end{cases} \tag{2.13}$$

where $\varphi(t)$ is the difference between the phases of the order parameters of the Cooper pair ground states, $\phi_0 = h/2e$ is the superconducting flux quantum and I_c is the maximum supercurrent that the junction can sustain. Taking now the time derivative of the current equation and isolating the time derivative of φ, we obtain the following equation:

$$\dot{\varphi} = \frac{1}{I_c \cos(\varphi)} \dot{I} \tag{2.14}$$

that can be plugged into the voltage equation to obtain the following relation:

$$V = \frac{\phi_0}{2\pi I_c} \frac{1}{\sqrt{1-(I/I_c)^2}} \dot{I}\,. \tag{2.15}$$

It is clear that this equation resembles the equation of an inductor $V = L\dot{I}$ (where the current have been taken with the convention of Fig. 2.2), but the voltage is not a linear function of the current. This last note is due to the fact that the Josephson junction introduces a non-linearity; this helpful contribution will help us to create more reliable qubits. This non-linear inductance is often referred to as the Josephson inductance:

$$L_j = \frac{\phi_0}{2\pi I_c} \frac{1}{\sqrt{1-(I/I_c)^2}} \tag{2.16}$$

and can be considered as a linear inductor only for small currents with associated value

$$\lim_{I\to 0} L_j(I) \to \frac{\phi_0}{2\pi I_c}\,.$$

Substituting now the inductor in the LC resonator with a Josephson junction we obtain the circuit in Fig. 2.2.

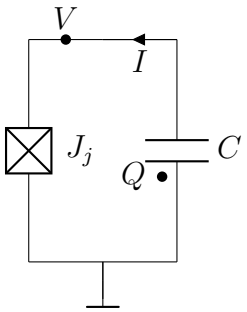


Figure 2.2: Schematic of the Josephson junction – indicated with the square box with the cross inside – with a parallel capacitance.

Compared to equation 2.7, we need to substitute the energy associated to the magnetic field of the inductor with the one associated to the Josephson junction. To obtain this expression we integrate the tunneling current multiplied by the voltage difference between the two superconducting sectors such that:

$$\begin{aligned} U_J = \int I(t)V(t)dt = \int I_c \sin\left(\frac{2\pi}{\phi_o}\phi\right)\frac{d\phi}{dt}dt = \\ \int_{\phi_1}^{\phi_2} I_c \sin\left(\frac{2\pi}{\phi_o}\phi\right)d\phi = -\frac{I_c\phi_0}{2\pi}\cos\left(\frac{2\pi}{\phi_0}\phi_1\right) \end{aligned} \tag{2.17}$$

where we chose ϕ_2 such that $\cos\left(\frac{2\pi}{\phi_0}\phi_2\right) = 0$. We can now define the Josephson energy as:

$$E_J = \frac{I_c\phi_0}{2\pi} \tag{2.18}$$

and obtain the standard form of the energy associated to the Josephson junction:

$$U_J(\phi) = -E_J\cos\left(\frac{2\pi}{\phi_0}\phi\right). \tag{2.19}$$

Substituting now this expression to the magnetic part of the Hamiltonian 2.7, we obtain the Hamiltonian associated to a generic capacitor-Josephson junction parallel:

$$H_J = \frac{1}{2}\frac{Q^2}{C} - E_J\cos\left(\frac{2\pi}{\phi_0}\phi\right). \tag{2.20}$$

From equation 2.20, we can observe that, near the minimum of the cosine, the expression is approximately a quantum linear harmonic oscillator while, when the energy goes up, we drift into the case of the free particle, hence the higher the energy, the smaller is the energy splitting.

2.2.2 The Cooper pair box

The Cooper pair box is one of the simplest circuits to use the non-linearity of a Josephson junction to obtain a reliable physical system to represent a qubit and it is represented in Fig. 2.3.

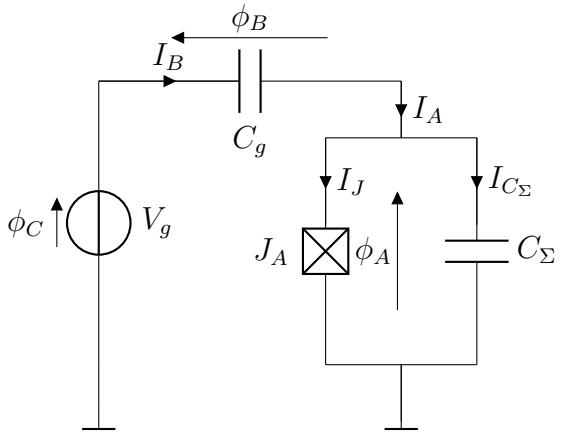


Figure 2.3: Schematic of a Cooper pair box with gate voltage V_g and gate capacitance C_g

To analyse this quantum circuit it is useful to associate a flux ϕ_i to each branch of the circuit such that $\dot{\phi}_i = V_i$. Applying now the Kirchhoff's laws we can obtain the following equations:

$$\begin{cases} \dot{\phi}_A = \dot{\phi}_C - \dot{\phi}_B \\ I_A = I_{C_\Sigma} + I_J = I_B \end{cases} \tag{2.21}$$

And the relations of the components in Fig. 2.3 are the following:

$$\begin{cases} I_{C_\Sigma} = C_\Sigma \ddot{\phi}_A \\ I_J = I_c \sin(\varphi_A) \\ I_B = C_g \ddot{\phi}_B \\ \dot{\phi}_C = V_g \end{cases} \tag{2.22}$$

Now it is possible to write the Lagrangian of the system in terms of energy stored in the capacitors C_Σ and C_g together with the Josephson energy associated to the Junction J_A such that:

$$\mathcal{L} = \frac{1}{2} C_\Sigma \dot{\phi}_A^2 + \frac{1}{2} C_g (\dot{\phi}_A - \dot{\phi}_C)^2 + E_j \cos(\varphi_A) \tag{2.23}$$

where the generalized coordinate that defines the dynamics of the system is ϕ_A. We can now obtain the conjugate momentum Q_A as

$$Q_A = \frac{\partial \mathcal{L}}{\partial \dot{\phi}_A} = (C_\Sigma + C_g)\dot{\phi}_A - C_g \phi_C \tag{2.24}$$

to finally obtain the Hamiltonian of the system

$$\begin{aligned} H = Q_A \dot{\phi}_A - \mathcal{L} = \frac{1}{2} C_\Sigma \dot{\phi}_A^2 + \frac{1}{2} C_g \dot{\phi}_A^2 - \frac{1}{2} C_g \dot{\phi}_C^2 - E_j \cos(\varphi_A) \\ \equiv \frac{1}{2} C_\Sigma \dot{\phi}_A^2 + \frac{1}{2} C_g \dot{\phi}_A^2 - E_j \cos(\varphi_A) = \frac{1}{2} \frac{(Q_A + C_g V_g)^2}{C_\Sigma + C_g} - E_j \cos(\varphi_A) \end{aligned} \tag{2.25}$$

where the term in $\dot{\phi}_C$ has been omitted because it does not contribute to the dynamic of the system. In the literature the Cooper pair box Hamiltonian is often written in another form introducing the normalized operators and quantities

$$\begin{cases} Q_g = C_g V_g \\ n = \frac{Q_A}{2e} \\ n_g = \frac{Q_g}{2e} \\ E_C = \frac{e^2}{2(C_\Sigma + C_g)} \end{cases} \tag{2.26}$$

where n is the operator associated to the number of Cooper pairs, n_g is a noise that causes decoherence and E_C is the energy needed to charge an electron in the total capacitance in parallel to the Josephson junction. With those definitions the Hamiltonian becomes:

$$H = 4E_c (n - n_g)^2 - E_j \cos(\varphi_A)\,. \tag{2.27}$$

2.2.3 The Transmon qubit

Transmon qubits [21] are just a straightforward application of the Cooper pair box in a particular regime, named transmon regime. Consider now the Hamiltonian of the Cooper pair box:

$$H = 4E_c (n - n_g)^2 - E_j \cos(\varphi_A)\,.$$

This equation can be studied varying the ratio E_J/E_C. If $E_J/E_C << 1$, then $H \approx 4E_c(n-n_g)^2$, hence the eigenstates will be extremely de-localized when projected in the φ_A space and, thanks to the Fourier transformation between canonical variables, very localized in the n space. On the other side, when $E_J/E_C >> 1$ we enter the so-called transmon regime where the effect is the opposite: the wave function in the φ_A space is localized while in the n space it is very de-localized. This last observation implies that the presence of n_g does not influence the dynamics of the system because the wave function is extremely delocalized in the n space, hence we will assume this approximation in the analysis of the Transmon qubit neglecting the n_g term. This leads obviously to the transmon Hamiltonian:

$$H_t = 4E_c n^2 - E_j \cos(\varphi_A)\,. \tag{2.28}$$

From equation 2.28 we can sum and subtract $\frac{1}{2}E_J\varphi_A^2$ to split the equation in the following two terms where the first one is harmonic and the second is anharmonic.

$$H_t = \left(4E_C n^2 + \frac{1}{2}E_j\varphi_A^2\right) - E_j\left(\cos(\varphi_A) + \frac{1}{2}\varphi_A^2\right). \tag{2.29}$$

We can now expand via Taylor series the nonlinear part related to the Josephson junction up to the fourth order such that

$$\begin{gathered} H_t \approx \left(4E_C n^2 + \frac{1}{2}E_j\varphi_A^2\right) - E_J\left(1 - \frac{\varphi_A^2}{2} + \frac{\varphi_A^4}{4!} + \frac{\varphi_A^2}{2}\right) = \\ \left(4E_C n^2 + \frac{1}{2}E_j\varphi_A^2\right) - E_J\left(1 + \frac{\varphi_A^4}{4!}\right) \equiv \left(4E_C n^2 + \frac{1}{2}E_j\varphi_A^2\right) - E_J\frac{\varphi_A^4}{4!} \end{gathered} \tag{2.30}$$

where constant terms have been omitted as before. From this expression of the Hamiltonian we can obtain its second quantized form rewriting the normalized canonical couple φ_A and n as functions of the creation and annihilation operators for the transmon qubit as follow:

$$\begin{cases} \varphi = \sqrt[4]{\frac{2E_C}{E_J}}(b + b^\dagger) \\ n = \frac{-i}{2}\sqrt[4]{\frac{E_J}{2E_c}}(b - b^\dagger) \end{cases} \tag{2.31}$$

where $[b, b^\dagger] = 1$.

Plugging now these definitions in the harmonic part of the Hamiltonian 2.30 we can arrive at the form:

$$H_t = \hbar\omega_p b^\dagger b - E_c\frac{(b+b^\dagger)^4}{12}, \qquad \hbar\omega_p = \sqrt{8E_C E_J}. \tag{2.32}$$

To obtain a simpler expression for the Hamiltonian we choose to apply the rotating wave approximation (RWA). This approximation consists in neglecting all the products $b^{\dagger m}b^\ell$ arising from the fourth power term such that $m \neq \ell$, hence the final second quantized expression of the Hamiltonian is:

$$H_t^{RWA} = \hbar\omega_q b^\dagger b - \frac{E_C}{2}b^\dagger b^\dagger b b, \qquad \hbar\omega_q = \sqrt{8E_C E_J} - E_C\,. \tag{2.33}$$

From equation 2.33 we can directly prove that the eigenstates of the transmon qubit, assuming the RWA to be valid, are the eigenstates of the number operator $n = b^\dagger b$. It

can be proved through the algebra of the creation and annihilation operators that let us rewrite the RWA Hamiltonian as:

$$H_t^{RWA} = \hbar \left(\omega_q - \frac{E_c}{2\hbar}(b^\dagger b - 1) \right) b^\dagger b \tag{2.34}$$

and this also shows that the eigenvalues of the Schrödinger problem associated to H_t^{RWA} are:

$$E_m = \left(\hbar\omega_q - \frac{E_C}{2}(m-1) \right) m, \qquad m \in Z_+ \cup \{0\}\,. \tag{2.35}$$

These eigenvalues are not linear in m as we expected from the introduction of a non-linearity. To capture the effect of the non-linearity intruduced by the Josephson junction we define the so-called anharmonicity α as the difference between the second energy transition and the first energy transition between levels:

$$\alpha = (E_2 - E_1) - (E_1 - E_0) = -E_C < 0\,. \tag{2.36}$$

In the end a transmon qubit is completely defined by two parameters: $\omega_q/(2\pi) \in [4, 10] GHz/s$ and $\alpha/(2\pi) \in [100, 400] MHz$.

2.3 Coupling in cQED

During the execution of unitary gates and measurements we want our qubits to interact. To make these interactions happen we need to utilize some kind of physical connection between our qubits. This is one of the bottlenecks of the superconducting platform for quantum computing considering that to have an interaction between qubits a physical connection must be built.

2.3.1 LC-LC capacitive coupling

Consider the scheme in Fig. 2.4 made of two resonators and a capacitance to couple them.

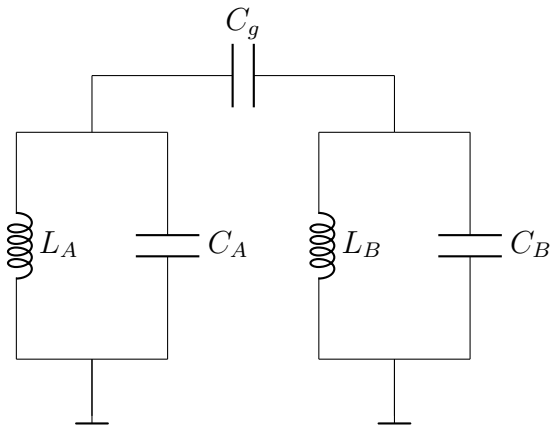


Figure 2.4: Schematic of an LC-LC circuit with capacitive coupling.

The capacitance matrix, the inductance matrix and the flux vector associated to the circuit in Fig. 2.4 read:

$$\mathbf{C} = \begin{pmatrix} C_A + C_g & -C_g \\ -C_g & C_B + C_g \end{pmatrix}, \qquad \tilde{\mathbf{L}} = \begin{pmatrix} 1/L_A & 0 \\ 0 & 1/L_B \end{pmatrix}, \qquad \boldsymbol{\phi} = \begin{pmatrix} \phi_A \\ \phi_B \end{pmatrix}. \tag{2.37}$$

The Lagrangian of the system built as the difference between the energy of the electric fields associated to the capacitors and the energy of the magnetic fields associated to the inductors can be expressed as

$$\mathcal{L}(\boldsymbol{\phi}, \dot{\boldsymbol{\phi}}) = \frac{1}{2}\dot{\boldsymbol{\phi}}^t \mathbf{C}\dot{\boldsymbol{\phi}} - \frac{1}{2}\boldsymbol{\phi}^t \tilde{\mathbf{L}}^{-1}\boldsymbol{\phi}\,. \tag{2.38}$$

From equation 2.38 we can compute the conjugate momentum of $\boldsymbol{\phi}$

$$\boldsymbol{Q} = \begin{pmatrix} Q_A \\ Q_B \end{pmatrix} \tag{2.39}$$

whose components are

$$\begin{cases} Q_A = \frac{\partial \mathcal{L}}{\partial \dot{\phi}_A} = C_A\dot{\phi}_A + C_g(\dot{\phi}_A - \dot{\phi}_B) \\ Q_B = \frac{\partial \mathcal{L}}{\partial \dot{\phi}_B} = C_B\dot{\phi}_B + C_g(\dot{\phi}_B - \dot{\phi}_A) \end{cases} \tag{2.40}$$

From the conjugate momentum expression, it is immediate to notice that

$$\boldsymbol{Q} = \mathbf{C}\dot{\boldsymbol{\phi}} \tag{2.41}$$

hence, knowing that $\mathbf{C}$ is symmetric we can also state that

$$\dot{\boldsymbol{\phi}} = \mathbf{C}^{-1}\boldsymbol{Q}, \qquad \dot{\boldsymbol{\phi}}^t = \boldsymbol{Q}^t\mathbf{C}^{-1} \tag{2.42}$$

where

$$\mathbf{C}^{-1} = \frac{1}{\bar{C}^2}\begin{pmatrix} C_B + C_g & C_g \\ C_g & C_A + C_g \end{pmatrix}, \qquad \bar{C}^2 = \det(\mathbf{C}). \tag{2.43}$$

These relations leads then straightforwardly to the Hamiltonian

$$H = \dot{\boldsymbol{\phi}}^t\boldsymbol{Q} - \mathcal{L}(\boldsymbol{\phi}, \dot{\boldsymbol{\phi}}) = \frac{1}{2}\boldsymbol{Q}^t\mathbf{C}^{-1}\boldsymbol{Q} + \frac{1}{2}\boldsymbol{\phi}^t\mathbf{L}^{-1}\boldsymbol{\phi} \tag{2.44}$$

or, in its expanded version:

$$H = \frac{(C_B + C_g)Q_A^2}{2\bar{C}^2} + \frac{(C_A + C_g)Q_B^2}{2\bar{C}^2} + \frac{C_g}{\bar{C}^2}Q_AQ_B + \frac{\phi_A^2}{2L_A} + \frac{\phi_B^2}{2L_B}\,. \tag{2.45}$$

From this form of the Hamiltonian we can identify the interaction term in the product $(C_g/\bar{C}^2)Q_AQ_B$ and, most of all, if we impose an open circuit considering the limit $C_g \to 0$ we obtain the non-interacting Hamiltonian as the sum of two LC resonator Hamiltonians.

2.3.2 LC-Transmon capacitive coupling

In quantum computing applications we are interested in how an LC resonator interacts with a transmon qubit, hence we consider the schematic in Fig. 2.5. To obtain the Hamiltonian of this schematic we can just substitute the term associated to the energy of the magnetic field of the substituted inductor and obtain the following equation:

$$H = \left(\frac{(C_A + C_g)Q_B^2}{2\bar{C}^2} - E_j\cos(\varphi_B)\right) + \frac{C_g}{\bar{C}^2}Q_AQ_B + \left(\frac{(C_B + C_g)Q_A^2}{2\bar{C}^2} + \frac{\phi_A^2}{2L_A}\right). \tag{2.46}$$

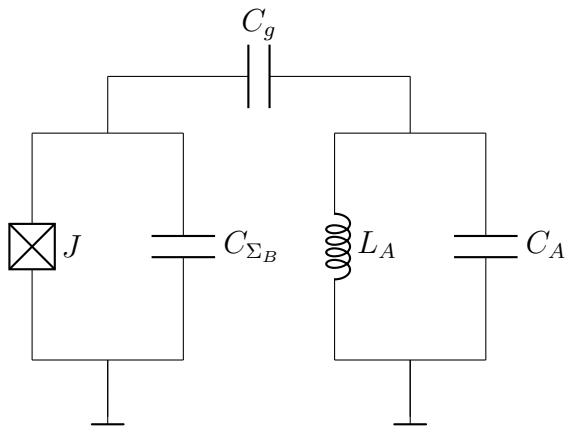


Figure 2.5: Schematic of a transmon-LC circuit with capacitive coupling

Now, assuming $C_{\Sigma_B}, C_A >> C_g$, we can approximate the Hamiltonian in the following way:

$$H \approx \left(\frac{Q_B^2}{2C_{\Sigma_B}} - E_j \cos(\varphi_B)\right) + \frac{C_g}{C_{\Sigma_B} C_A} Q_A Q_B + \left(\frac{Q_A^2}{2C_A} + \frac{\phi_A^2}{2L_A}\right). \tag{2.47}$$

This last equation is divided in three parts on purpose: in the first set of brackets there is the non-interacting Hamiltonian of a Cooper pair box, in the second set of brackets there is the non-interacting Hamiltonian of an LC resonator and in the middle there is the interaction term between the two systems due to the presence of the capacitance C_g. Assuming now to be in transmon regime and the validity of the RWA, using the definition introduced for the transmon qubit and the LC resonator in the previous sections and omitting the terms that do not contribute to the dynamics of the system, we can obtain the following approximate form of the Hamiltonian:

$$H \approx \left(\hbar\omega_{q_B} b^\dagger b - E_{C_B} b^\dagger b^\dagger\right) - \hbar g (b^\dagger - b)(a^\dagger - a) + \left(\hbar\omega_{R_A} a^\dagger a\right) \tag{2.48}$$

where $b^\dagger, b$ and $a^\dagger, a$ are respectively the creation and annihilation operators of the transmon qubit, eq. 2.31, and of the quantum linear harmonic oscillator, eq. 2.9, while ω_{q_B} is the transition frequency of the transmon, ω_{R_A} is the transition frequency of the resonator and E_{C_B} is the energy needed to put an electron in the total capacitance in parallel to the Josephson junction. Regarding the term g, its expression reads

$$g = \frac{C_g}{\sqrt{C_{\Sigma_B} C_A}} \frac{\sqrt{\omega_{R_A} \omega_{p_B}}}{2} \tag{2.49}$$

where ω_{p_B} is the plasma frequency of the transmon qubit.

There is another expression that is quite used in literature and can be obtained by applying the RWA to 2.48 such that:

$$H \approx \left(\hbar\omega_{q_B} b^\dagger b - E_{C_B} b^\dagger b^\dagger\right) - \hbar g (b^\dagger a - b a^\dagger) + \left(\hbar\omega_{r_A} a^\dagger a\right). \tag{2.50}$$

This last expression is a valid formula for $|g| << \omega_{R_A}, \omega_{q_B}$. From this very last form of the Hamiltonian it is possible to derive the so-called Jaynes-Cummings model which is widely used in cQED. This model can be obtained from equation 2.50 considering our transmon as a perfect 2-level system, hence doing the mappings $b^\dagger \to \sigma_+$ and $b \to \sigma_-$

where $\sigma_+ = |1\rangle\langle 0|$ and $\sigma_- = |0\rangle\langle 1|$. Following this procedure we obtain the so-called Jaynes-Cummings Hamiltonian:

$$H = \frac{\hbar\omega_{qB}}{2} Z + \hbar g(a^\dagger \sigma_- + a\sigma_+) + \hbar\omega_{r_A} a^\dagger a \,. \tag{2.51}$$

The Jaynes-Cummings model can be solved quite straightforwardly if we construct the solutions of the interacting system, called dressed states, from the solution of the non-interacting model obtaining setting $g = 0$.

Consider now the non-interacting Hamiltonian:

$$H_{\mathrm{JC}} = \frac{\hbar\omega_{qB}}{2} Z + \hbar\omega_{r_A} a^\dagger a \,. \tag{2.52}$$

This model contains the resonator Hamiltonian and the transmon Hamiltonian limited to the two-dimensional Hilbert space of the first two energy levels; these two operators commute and so the eigenstates of the total non-interacting Hamiltonian are $|\sigma, n\rangle$ with $\sigma \in \{0, 1\}$ and $n \in Z_+ \cup \{0\}$. The states $|\sigma, n\rangle$ are named as bare states. If we apply now the coupling term contained in the Jaynes-Cummings model to the bare states at equal number of excitations, the result for $n \geq 1$ excitations is:

$$\begin{cases} (a^\dagger \sigma_- + a\sigma_+)\,|0, n\rangle = \sqrt{n}\,|1, n-1\rangle \\ (a^\dagger \sigma_- + a\sigma_+)\,|1, n-1\rangle = \sqrt{n}\,|0, n\rangle \end{cases} \tag{2.53}$$

hence, the interaction only couples bare states with the same number of excitations in the system. This leads to a block-diagonal Hamiltonian matrix that can be diagonalized block by block. The generic block at constant number of interactions can be written in the basis $\{|0, n\rangle, |1, n-1\rangle\}$ as:

$$\mathbf{H}_n = \begin{pmatrix} -\dfrac{\hbar\omega_{qB}}{2} + \hbar(a^\dagger a - \dfrac{1}{2}) & \hbar g\sqrt{n} \\ \hbar g\sqrt{n} & \dfrac{\hbar\omega_{qB}}{2} + \hbar(a^\dagger a - \dfrac{1}{2}) \end{pmatrix} \tag{2.54}$$

The eigenvalues of this matrix are:

$$E_n^{\pm} = \hbar\omega_{rA} \pm \frac{\hbar}{2}\sqrt{4g^2 n + \Delta^2} \,. \tag{2.55}$$

Solving now back the eigenvalue equation to obtain the eigenvectors we obtain the dressed states as:

$$\begin{cases} \overline{|0, n\rangle} = \cos\left(\frac{\theta_n}{2}\right)|0, n\rangle - \sin\left(\frac{\theta_n}{2}\right)|1, n-1\rangle \\ \overline{|1, n-1\rangle} = \sin\left(\frac{\theta_n}{2}\right)|0, n\rangle + \cos\left(\frac{\theta_n}{2}\right)|1, n-1\rangle \end{cases} \tag{2.56}$$

where $\theta_n = \arctan\left(\frac{2g\sqrt{n}}{\omega_{q_B} - \omega_{r_A}}\right)$.

For the case with $n = 0$ excitations, if we apply the interaction we obtain the null vector, hence the ground state and the ground state energy remain the same such that $\overline{|0, 0\rangle} = |0, 0\rangle$ and $E_0 = -\frac{\hbar\omega_{q_B}}{2} + \frac{\hbar\omega_{r_A}}{2}$.

2.4 Dispersive regime

We are interested to study the so-called dispersive regime, which is defined by $|\frac{g}{\Delta}| << 1$ with $\Delta = \omega_{q_B} - \omega_{r_a}$. In this regime the Jaynes-Cummings Hamiltonian can be approximated to

$$H_{\text{JC}} \approx \frac{\hbar\omega'_{qB}}{2} Z + \hbar(\omega_{r_A} + \frac{g^2}{\Delta} Z) a^\dagger a, \qquad \omega'_{q_B} = \omega_{q_B} + \frac{g^2}{\Delta}\,. \tag{2.57}$$

This expression tells us that the we can look at this equation as if it was the non-interacting Hamiltonian 2.52 with a slightly modified qubit frequency, called Lamb-shifted qubit frequency, and a qubit-state dependent resonator frequency. The only way this behaviour is possible is thanks to the non-linearity introduced by the transmon qubit. The quantity $\chi = \frac{g^2}{\Delta}$ is often referred to as dispersive shift and so we can write the Hamiltonian as follows:

$$H_{\text{JC}} \approx \frac{\hbar\omega'_{qB}}{2} Z + \hbar(\omega_{r_A} + \chi Z) a^\dagger a\,. \tag{2.58}$$

This last result is an example of how the dispersive regime approximation can simplify the expressions in cQED; from now on we will always assume this regime for our purposes.

2.4.1 Qubit-LC-qubit coupling in dispersive regime

In this section we will consider a structure made of two transmons connected to a resonator in the middle like in the following figure:

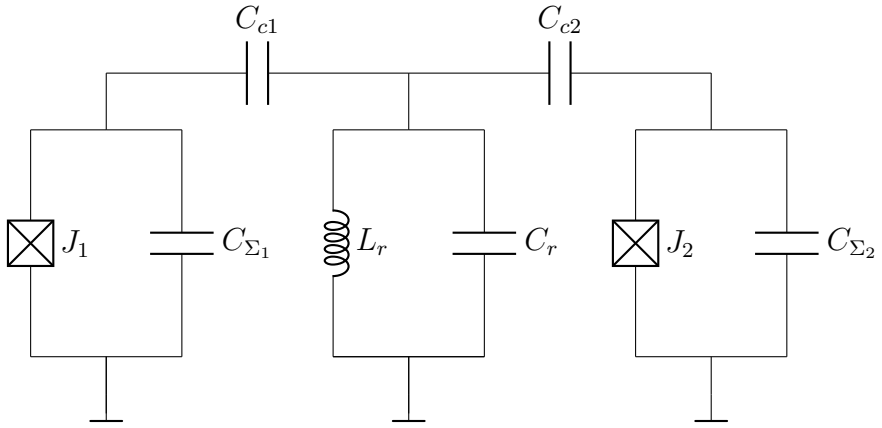


Figure 2.6: This figure represents a capacitive coupled system of two transmons and an LC resonator

Following now what is reported by Lalumière et al. in [24] if we approximate the transmons to qubits and apply a measurement drive of frequency ω_m and envelope $\epsilon_m(t)$ to the resonator, we can write in the dispersive regime, hence $\Delta_j = |\omega_{q_j} - \omega_r| >> |g_j|$ where g_j is the coupling strength of the j-th qubit with $j \in \{1, 2\}$, the following Hamiltonian:

$$\begin{aligned} H = \left(\omega_r + \sum_{j\in\{1,2\}} \chi^j Z_j\right) a^\dagger a + \sum_{j\in\{1,2\}} \frac{\omega'_{q_j}}{2} Z_j + J_q(\sigma_-^1\sigma_+^2 + \sigma_-^2\sigma_+^1) + \\ +\epsilon_m(t)(a^\dagger e^{-i\omega_m t} + h.c.) \end{aligned} \tag{2.59}$$

where $J_q = g_1 g_2(1/\Delta_1 + 1/\Delta_2)$ is the strength of the qubit-qubit coupling mediated by a photon. The frequency of the drive applied to the cavity, ω_m, must close to the resonator

frequency to obtain a good readout. Dropping now the J_q dependent term in order to focus only on the entanglement generated only by the measurement, and defining the signed dispersive parameter $\lambda_j = g_j/\Delta_j$, it is possible to obtain the master equation reported in [24] that reads

$$\begin{aligned}\dot{\rho} \approx -i[\sum_{j\in\{1,2\}} \frac{\omega'_{q_j}}{2} Z_j, \rho] + \sum_{j\in\{1,2\}} \gamma_{1_j}\mathcal{D}[\sigma_-^j]\rho + \sum_{j\in\{1,2\}} \frac{\gamma_{\phi_j}}{2}\mathcal{D}[Z_j]\rho \\ +\kappa\mathcal{D}[\sum_{j\in\{1,2\}} \lambda_j\sigma_-^j]\rho + \sum_{x,y\in\mathbb{F}_2^2} (\Gamma_d^{xy} - iA_c^{xy})\Pi_x\rho\Pi_y \equiv \mathcal{L}\rho\end{aligned} \tag{2.60}$$

where $\mathcal{D}[f](\cdot) = f(\cdot)f^\dagger - \{f^\dagger f, (\cdot)\}$ is the dissipator, Π_x is the projector into the subspace defined by the state $|x\rangle$, κ is the photon decay rate and represents the coupling between the resonator and the transmission line needed to perform the readout, γ_{1_j} is the relaxation rate of the j-th qubit and γ_{ϕ_j} is the pure dephasing rate of the j-th qubit. The fourth term represents the Purcell damping at rate $\lambda_j^2\kappa$ and the fifth term contains the measurement induced dephasing Γ_d^{xy} and the ac-Stark shift A_c^{xy}. Equation 2.60 is a good approximation under the hypothesis $\kappa/2 >> \gamma_{1_j}$, which is easily satisfiable.

2.5 Resonator states during the readout

Another important aspect to keep in mind for our study is that if we apply a drive at frequency ω_m to a resonator then under the RWA it will always stay in a coherent state, which is defined by the displacement operator $\hat{D}$ and a complex number α applied to the vacuum state of the resonator $|0\rangle$ as follows:

$$|\alpha\rangle = \hat{D}(\alpha)|0\rangle = e^{-|\alpha|^2}\sum_{n=0}^{+\infty}\frac{\alpha^n}{\sqrt{n!}}|n\rangle\,, \qquad \hat{D}(\alpha) = e^{\alpha a^\dagger - \alpha^* a}\,. \tag{2.61}$$

If we connect some qubits, in our case two, to the resonator satisfying the hypothesis of the dispersive regime, then α will be a quantity dependent from the qubit states, hence the general final state $|\Phi\rangle$ will be something of the following form:

$$|\Phi\rangle = \sum_{i,j\in\mathbb{F}_2} x_{ij}|ij,\alpha_{ij}\rangle\,. \tag{2.62}$$

Following Lalumière et al.[24] the coherent state parameter α_x relative to the qubit state $|x\rangle$ can be computed solving the following differential equation:

$$\dot{\alpha}_x = -i(\omega_r + \chi_x)\alpha_x - i\epsilon_m(t)e^{-i\omega_m t} - \frac{\kappa}{2}\alpha_x, \qquad x\in\mathbb{F}_2^2 \tag{2.63}$$

where $\chi_x = \langle x|\sum_j \chi^j Z_j|x\rangle$ will be named qubit-state-dependent dispersive shift. The coherent state parameter is very important for our purposes because it appears in the formula for the computation of the measurement induced dephasing and in the ac-Stark shift according to the following relations

$$\Gamma_d^{xy} = (\chi_x - \chi_y)\Im[\alpha_x\alpha_y^*] \tag{2.64}$$

$$A_c^{xy} = (\chi_x - \chi_y)\Re[\alpha_x\alpha_y^*] \tag{2.65}$$

where $\Im$ and $\Re$ are respectively the imaginary part and the real part of their argument.

Chapter 3

Error models for direct dispersive measurements

In every physical realization of quantum computers the biggest problems are always stochastic and/or systematic errors. There are plenty of error mechanism that could be considered; in our compact error model we are going to include single-qubit amplitude damping, single qubit pure dephasing and the two-qubit dephasing within the zero parity subspace of the two-qubit system defined by span$\{|00\rangle, |11\rangle\}$. We will show in the following chapter the methods used and implemented.

3.1 Pauli transfer matrix formalism

The Pauli transfer matrix (PTM) formalism [12] consists in a bijection between generic (bounded) operators acting on an n-qubits Hilbert space $\mathcal{H}$ and a set of coefficients. Essentially it is just a decomposition on a (normalized) basis of operators. It is very useful because it can map equations into linear algebra problems, leading to easier methods to solve the equations of motion for the density matrices of a quantum system.

We now remind the symplectic definition of the n-qubit Pauli operators:

$$P(\mathbf{r}) = i^{\mathbf{x}^t\mathbf{z}} X(\mathbf{x}) Z(\mathbf{z}), \quad \mathbf{r} = \begin{pmatrix} \mathbf{x} \\ \mathbf{z} \end{pmatrix} \in \mathbb{F}_2^{2n} \tag{3.1}$$

and their normalized version: $\mathbf{P}(\mathbf{r}) = P(\mathbf{r})/\sqrt{2^n}$. The set $\{\mathbf{P}(\mathbf{r})\}_{\mathbf{r}\in\mathbb{F}_2^{2n}}$ is an orthonormal basis for the (bounded) n-qubit operators under the Hilbert-Schmidt inner product $\langle A, B\rangle_{HS} = \mathrm{Tr}[A^\dagger B]$, hence $\forall A : \mathcal{H} \to \mathcal{H}$ we can write:

$$A = \sum_{\mathbf{r}\in\mathbb{F}_2^{2n}} \chi_A(\mathbf{r})\mathbf{P}(\mathbf{r}), \quad \chi_A(\mathbf{r}) = \langle \mathbf{P}(\mathbf{r}), A\rangle_{HS} \ . \tag{3.2}$$

This decomposition induces a bijection between the space of (bounded) n-qubit operators and the $\chi_A(\mathbf{r})$. The vector of the $\chi_A(\mathbf{r})$ is often denoted in the literature with the notation $|A\rangle\rangle$.

Consider now a generic linear map $\mathcal{M}$ that maps (bounded) operators to (bounded) operators (what in physics we usually call as superoperator). Applying $\mathcal{M}$ to a (bounded)

operator A we can apply the Pauli decomposition in the codomain to get the following result:

$$\mathcal{M}(A) = \sum_{\mathbf{r}\in\mathbb{F}_2^{2n}} \chi_{\mathcal{M}(A)}(\mathbf{r})\mathbf{P}(\mathbf{r})$$

and then use the same transformation into the definition of χ_A to obtain the following equation:

$$\chi_{\mathcal{M}(A)}(\mathbf{r}) = \sum_{r'\in\mathbb{F}_2^{2n}} \mathrm{Tr}[\mathbf{P}(\mathbf{r})\mathcal{M}(\mathbf{P}(\mathbf{r}'))]\chi_A(\mathbf{r}')\,.$$

From this result it is clear we obtained a matrix equation that connects the Pauli basis representation of the operator A with the Pauli basis representation of the operator $\mathcal{M}(A)$ such that:

$$|\mathcal{M}(A)\rangle\rangle = \hat{\mathcal{M}}|A\rangle\rangle, \quad \hat{\mathcal{M}}_{r,r'} = \langle\mathbf{P}(\mathbf{r})\mathcal{M}(\mathbf{P}(\mathbf{r}'))\rangle_{HS}\,. \tag{3.3}$$

The matrix $\hat{\mathcal{M}}$ is often referred to as *Pauli Transfer Matrix* of the operator $\mathcal{M}(\cdot)$. In our framework it will be very useful to consider $|\mathcal{M}(\rho)\rangle\rangle = |\dot{\rho}\rangle\rangle$ as the time evolution of the density matrix of the system and associate the PTM to the Lindbladian operator such that $\hat{\mathcal{M}} = \hat{\mathcal{L}}$.

3.2 Pauli Twirling

Pauli twirling [12] is a method to obtain a PTM − diagonal representation of a (bounded) superoperator acting on an n-qubits Hilbert space. From a mathematical point of view it can be seen as a consequence of Shur's lemma [5][34]. For further readings we also recommend the appendix of [30].

Consider now the single-qubit case and let $\mathcal{E}$ be a completely positive trace preserving (CPTP) map acting on single-qubit (bounded) operators with associated PTM $\hat{\mathcal{E}}$. It can be proven that, applying the elements of the Pauli basis before and after the channel $\mathcal{E}$, we obtain a new map $\mathcal{E}_{twirl}$ whose PTM representation $\hat{\mathcal{E}}_{twirl}$ is a diagonal matrix and can be obtained as follows:

$$\hat{\mathcal{E}}_{twirl} = \frac{1}{4}\left(\hat{\mathbb{1}}\hat{\mathcal{E}}\hat{\mathbb{1}} + \hat{X}\hat{\mathcal{E}}\hat{X} + \hat{Z}\hat{\mathcal{E}}\hat{Z} + \hat{Y}\hat{\mathcal{E}}\hat{Y}\right) \tag{3.4}$$

where $\hat{\mathcal{E}}$ is the PTM of the channel and $\hat{\mathbb{1}}$, $\hat{X}$, $\hat{Z}$ and $\hat{Y}$ are the PTMs associated to the four Kraus operators defined by the errors $\mathbb{1}$, X, Z and Y respectively:

$$\hat{\mathbb{1}} = \begin{pmatrix} +1 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 \\ 0 & 0 & +1 & 0 \\ 0 & 0 & 0 & +1 \end{pmatrix}, \quad \hat{Z} = \begin{pmatrix} +1 & 0 & 0 & 0 \\ 0 & +1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix},$$
$$\hat{X} = \begin{pmatrix} +1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & +1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \quad \hat{Y} = \begin{pmatrix} +1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & +1 \end{pmatrix} \tag{3.5}$$

The same holds for the n-qubits case where, considering $\mathcal{E}$ as a CPTP map acting on n-qubits (bounded) operators and representing the Paulis with the symplectic notation the following equation holds:

$$\hat{\mathcal{E}}_{twirl} = \frac{1}{2^{2n}} \sum_{\mathbf{r}\in\mathbb{F}_2^{2n}} \hat{P}(\mathbf{r})\hat{\mathcal{E}}\hat{P}(\mathbf{r})\,. \tag{3.6}$$

This technique from a theoretical perspective is actually very powerful but once we connect it to real applications it is an approximation because we are not able to realize perfect elements of the Pauli basis. In this project we are going to use the Pauli Twirling which is a de-facto standard in modern QEC papers.

3.2.1 Why do we need a diagonal PTM?

To appreciate the power of having a diagonal PTM, consider now a generic stochastic Pauli channel (SPC), or stochastic Pauli noise, [12] applied to an operator A:

$$\mathcal{E}_{Pauli}(A) = \left(1 - \sum_{\mathbf{r}\in\mathbf{F}_2^{2n}\setminus\mathbf{0}} p_{\mathbf{r}}\right) A + \sum_{\mathbf{r}\in\mathbf{F}_2^{2n}\setminus\mathbf{0}} p_{\mathbf{r}} P(\mathbf{r}) A P(\mathbf{r})\,. \tag{3.7}$$

The PTM of this channel is diagonal and so we can obtain an SPC from a generic diagonal PTM. To prove this property we just have to compute straightforwardly the PTM of the SPC using the Pauli basis definition and obtain the following result:

$$\hat{\mathcal{E}}_{Pauli}\left[\mathbf{r}_i, \mathbf{r}_j\right] = \left(1 - \sum_{\mathbf{r}\in\mathbf{F}_2^{2n}\setminus\mathbf{0}} \left(1 - (-1)^{\langle \mathbf{r}_i, \mathbf{r}\rangle_\Omega}\right) p_{\mathbf{r}}\right) \delta_{\mathbf{r}_i,\mathbf{r}_j} \tag{3.8}$$

where $\langle \mathbf{r}_i, \mathbf{r}\rangle_\Omega$ is the symplectic inner product defined in Appendix C. Equation 3.8 proves the claim and leads to the single qubit case:

$$\hat{\mathcal{E}}_{Pauli}^{n=1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1-2(p_{1|0}+p_{1|1}) & 0 & 0 \\ 0 & 0 & 1-2(p_{0|1}+p_{1|1}) & 0 \\ 0 & 0 & 0 & 1-2(p_{0|1}+p_{1|0}) \end{pmatrix}. \tag{3.9}$$

It is important to state that the previous matrix have been written within the Pauli basis ordered by counting the decimal value associated to the symplectic vector $\mathbf{r}$, hence the order is the following $\mathbb{1}, Z, X, Y$. From now on the all the PTM will be written in the Pauli basis ordered with this procedure that can be generalized to n-qubits; for example for $n = 2$ it will start the following way: $\mathbb{1}\otimes\mathbb{1}, \mathbb{1}\otimes Z, Z\otimes\mathbb{1}, Z\otimes Z, \mathbb{1}\otimes X, \mathbb{1}\otimes XZ$ etc.

For completeness, we report also the diagonal elements of the 2-qubit case. For simplicity the symplectic representation of Pauli operators $x_2x_1|z_2z_1$ has been converted to their corresponding decimal representation.

$$\begin{cases} \hat{\mathcal{E}}^{n=2}_{Pauli}(0) = 1 \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(1) = 1 - 2(p_4 + p_5 + p_6 + p_7 + p_{12} + p_{13} + p_{14} + p_{15}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(2) = 1 - 2(p_8 + p_9 + p_{10} + p_{11} + p_{12} + p_{13} + p_{14} + p_{15}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(3) = 1 - 2(p_4 + p_5 + p_6 + p_7 + p_8 + p_9 + p_{10} + p_{11}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(4) = 1 - 2(p_1 + p_3 + p_5 + p_7 + p_9 + p_{11} + p_{13} + p_{15}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(5) = 1 - 2(p_1 + p_3 + p_4 + p_6 + p_9 + p_{11} + p_{12} + p_{14}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(6) = 1 - 2(p_1 + p_3 + p_5 + p_7 + p_8 + p_{10} + p_{12} + p_{14}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(7) = 1 - 2(p_1 + p_3 + p_4 + p_6 + p_8 + p_{10} + p_{13} + p_{15}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(8) = 1 - 2(p_2 + p_3 + p_6 + p_7 + p_{10} + p_{11} + p_{14} + p_{15}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(9) = 1 - 2(p_2 + p_3 + p_4 + p_5 + p_{10} + p_{11} + p_{12} + p_{13}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(10) = 1 - 2(p_2 + p_3 + p_6 + p_7 + p_8 + p_9 + p_{12} + p_{13}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(11) = 1 - 2(p_2 + p_3 + p_4 + p_5 + p_8 + p_9 + p_{14} + p_{15}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(12) = 1 - 2(p_1 + p_2 + p_5 + p_6 + p_9 + p_{10} + p_{13} + p_{14}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(13) = 1 - 2(p_1 + p_2 + p_4 + p_7 + p_9 + p_{10} + p_{12} + p_{15}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(14) = 1 - 2(p_1 + p_2 + p_5 + p_6 + p_8 + p_{11} + p_{12} + p_{15}) \\ \hat{\mathcal{E}}^{n=2}_{Pauli}(15) = 1 - 2(p_1 + p_2 + p_4 + p_7 + p_8 + p_{11} + p_{13} + p_{14}) \end{cases} \tag{3.10}$$

3.3 Solutions of the Lindblad master equation via PTM

The quantum systems we study constantly interact with the environment and the total Hilbert space of the universe can be described as $\mathcal{H} = \mathcal{H}_S \otimes \mathcal{H}_E$. The evolution of the complete density operator ρ is completely described by the total Hamiltonian H and the Liouville-Von Neumann equation:

$$\dot{\rho} = -i\,[H, \rho]\,. \tag{3.11}$$

This equation can be derived straightforwardly from the Schrödinger equation like in [4]. Tracing out the environment via partial trace and assuming Markovian evolution one can obtain the Lindblad master equation:

$$\dot{\rho} = -i[H_{\text{eff}}, \rho] + \sum_k \mathcal{D}[L_k](\rho), \quad \mathcal{D}[L_k](\rho) = L_k L_k^\dagger - \frac{1}{2}\left\{L_k^\dagger L_k, \rho\right\}. \tag{3.12}$$

where $\mathcal{D}$ is usually referred to as the dissipator, the $\{L_k\}$ are the jump operators and H_{eff} is an effective Hamiltonian.
In the literature it is usually expressed in a more compact way defining the Lindbladian operator such that

$$\dot{\rho} = \mathcal{L}(\rho). \tag{3.13}$$

We are interested in studying amplitude damping and pure dephasing processes, so we do not care about the coherent evolution given by the commutator. For this reason we are going to study the following master equation:

$$\dot{\rho} = \sum_k \mathcal{D}[L_k](\rho). \tag{3.14}$$

If we assume time independent jump operators, equation 3.14 can be seen as equation 3.12 in a reference frame that is rotating coherently with the dynamics associated to H_{eff}. Other possibilities could be a special case where the commutator is zero or simply an

approximation.

Thanks to the PTM formalism we can map this equation to linear algebra problems and so it is possible to obtain a simpler method to solve the equation of motion. To achieve this mapping we can apply expansion 3.2 to the density matrix and thanks to the linearity of Lindblad's equation we get the following equality:

$$\sum_{\mathbf{r}\in\mathbb{F}_2^{2n}} \mathbf{P}(\mathbf{r})\dot{\chi}_\rho(\mathbf{r}) = \sum_{\mathbf{r}\in\mathbb{F}_2^{2n}} \left(L_k \mathbf{P}(\mathbf{r}) L_k^\dagger - \frac{1}{2}\left\{ L_k^\dagger L_k, \mathbf{P}(\mathbf{r}) \right\} \right) \chi_\rho(\mathbf{r}) \,.$$

Once we obtained this form of the master equation we can project it on $\mathbf{P}(\mathbf{r})$ via the Hilbert-Schmidt inner product, hence we get the PTM version of the Lindblad master equation:

$$\dot{\chi}_\rho(\mathbf{r}_i) = \sum_{\mathbf{r}_j\in\mathbb{F}_2^{2n}} \left\{ \sum_k \left(\mathrm{Tr}\left[\mathbf{P}(\mathbf{r}_i) L_k \mathbf{P}(\mathbf{r}_j) L_k^\dagger \right] - \frac{1}{2}\mathrm{Tr}\left[\mathbf{P}(\mathbf{r}_i)\left\{ L_k^\dagger L_k, \mathbf{P}(\mathbf{r}_j) \right\} \right] \right) \chi_\rho(\mathbf{r}_j) \right\} . \tag{3.15}$$

Defining the PTM matrix associated to the Lindbladian as

$$\mathbf{L}_{ij} = \sum_k \left(\mathrm{Tr}\left[\mathbf{P}(\mathbf{r}_i) L_k \mathbf{P}(\mathbf{r_j}) L_k^\dagger \right] - \frac{1}{2}\mathrm{Tr}\left[\mathbf{P}(\mathbf{r}_i)\left\{ L_k^\dagger L_k, \mathbf{P}(\mathbf{r}_j) \right\} \right] \right) \tag{3.16}$$

we can write the general result that will be useful in all our calculations: the Lindblad master equation written in the PTM formalism:

$$|\rho\rangle\rangle = \mathbf{L}|\rho\rangle\rangle \,. \tag{3.17}$$

This equation can then be integrated to obtain the following formal solution

$$|\rho(t)\rangle\rangle = e^{\mathbf{L}(t-t_0)}|\rho(t_0)\rangle\rangle \tag{3.18}$$

where $e^{\mathbf{L}(t-t_0)}$ is the evolution operator associated to the quantum channel. Now we have all the tools to solve the Lindblad equation using the PTM representation.

3.3.1 Single-qubit amplitude damping

Consider now a single qubit whose Markovian dynamics is fully described by a single jump operator $L = \sqrt{\gamma_1}\sigma_-$, where $\gamma_1 = 1/T_1$ is the inverse of the relaxation time and $\sigma_- = |0\rangle\langle 1|$. This choice for the definition fixes the ground state to $|0\rangle$ and the excited state to $|1\rangle$ but it is a completely arbitrary choice. We invite the reader to consult [26] for a non-PTM approach to obtain the solution.

With the highlighted definitions, the master equation for the qubit reads:

$$\dot{\rho} = \gamma_1 \left(\sigma_- \rho \sigma_+ - \frac{1}{2}\{\sigma_+\sigma_-, \rho\} \right) . \tag{3.19}$$

From equation 3.16 the PTM matrix of the Lindbladian takes the form:

$$\begin{aligned} \mathbf{L}_{ij} &= \gamma_1 \left(\mathrm{Tr}\left[\mathbf{P}(\mathbf{r}_i)\sigma_- \mathbf{P}(\mathbf{r}_j)\sigma_+ \right] - \frac{1}{2}\mathrm{Tr}\left[\mathbf{P}(\mathbf{r}_i)\{\sigma_+\sigma_-, \mathbf{P}(\mathbf{r}_j)\} \right] \right) = \\ &= \frac{\gamma_1 i^{x_i z_i + x_j z_j}}{2} \left(\mathrm{Tr}\left[X^{x_i} Z^{z_i} \sigma_- X^{x_j} Z^{z_j} \sigma_+ \right] - \frac{1}{2}\mathrm{Tr}\left[X^{x_i} Z^{z_i} \{\sigma_+\sigma_-, X^{x_j} Z^{z_j}\} \right] \right) \end{aligned} \tag{3.20}$$

We can now observe that $\sigma_\pm = \frac{1}{2}(X \mp iY)$ and using the algebra of the Pauli matrices, after some tedious calculations, we can obtain the following general form:

$$\begin{aligned}\mathbf{L}_{ij} = \frac{\gamma_1 i^{x_i z_i + x_j z_j}}{8}\delta_{x_i,x_j}\{2(-1)^{x_j z_i}\delta_{z_i,z_j}\left((-1)^{z_j}(1+(-1)^{x_j})-2\right)+ \\ +2\delta_{z_i,\bar{z}_j}\left((-1)^{z_j+x_j z_i}-(-1)^{z_i+x_i z_j}+(-1)^{x_j(1+z_i)}\left(1+(-1)^{x_j}\right)\right)\}.\end{aligned} \tag{3.21}$$

If the basis is ordered counting in binary arithmetic the symplectic vector, then 3.21 corresponds to the following 4×4 matrix:

$$\mathbf{L} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \gamma_1 & -\gamma_1 & 0 & 0 \\ 0 & 0 & -\gamma_1/2 & 0 \\ 0 & 0 & 0 & -\gamma_1/2 \end{pmatrix} \tag{3.22}$$

and we obtain the following four equations:

$$\begin{cases} \dot{\chi}_\rho(0|0,t) = 0 \\ \dot{\chi}_\rho(0|1,t) = \gamma_1\chi_\rho(0|0,t) - \gamma_1\chi_\rho(0|1,t) \\ \dot{\chi}_\rho(1|0,t) = -\frac{\gamma_1}{2}\chi_\rho(1|0,t) \\ \dot{\chi}_\rho(1|1,t) = -\frac{\gamma_1}{2}\chi_\rho(1|1,t) \end{cases} \tag{3.23}$$

Assuming to know χ_ρ at $t=0$, this set of equations can be easily solved by separating variable and integrating. With this procedure the solutions are:

$$\begin{cases} \chi_\rho(0|0,t) = \chi_\rho(0|0,0) \\ \chi_\rho(0|1,t) = \chi_\rho(0|0,0)\left(1-e^{-\gamma_1 t}\right) + \chi_\rho(0|1,0)e^{-\gamma_1 t} \\ \chi_\rho(1|0,t) = \chi_\rho(1|0,0)e^{-\frac{\gamma_1}{2}t} \\ \chi_\rho(1|1,t) = \chi_\rho(1|1,0)e^{-\frac{\gamma_1}{2}t} \end{cases} \tag{3.24}$$

At this point we can build the evolution matrix for the PTM-represented density operator. Its expression reads as follows:

$$e^{\mathbf{L}t} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \Gamma(t) & 1-\Gamma(t) & 0 & 0 \\ 0 & 0 & \sqrt{1-\Gamma(t)} & 0 \\ 0 & 0 & 0 & \sqrt{1-\Gamma(t)} \end{pmatrix}, \quad \Gamma(t) = 1-e^{-\gamma_1 t} \tag{3.25}$$

It is clear that there is no SPC associated to this matrix since it is not in diagonal form. To obtain a compact model our goal is to obtain an SPC, so the next step is to twirl the channel according to equation 3.4 with $\hat{\mathcal{E}} = e^{\mathbf{L}t}$ obtaining the following twirled channel:

$$\hat{\mathcal{E}}_{twirl} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1-\Gamma(t) & 0 & 0 \\ 0 & 0 & \sqrt{1-\Gamma(t)} & 0 \\ 0 & 0 & 0 & \sqrt{1-\Gamma(t)} \end{pmatrix} \tag{3.26}$$

Comparing equation 3.26 with equation 3.9 we can obtain the values of the probabilities associated to the Pauli errors solving the linear system and obtain the following mapping rules:

$$\begin{cases} p_{0|1} = \frac{1}{2}\left(1-\sqrt{1-\Gamma(t)}-\frac{\Gamma(t)}{2}\right) \\ p_{1|0} = p_{1|1} = \frac{\Gamma(t)}{4} \end{cases} \tag{3.27}$$

This last result contains a very straightforward physical hint: the x-axis and the y-axis are identical by simmetry with respect to a process whose only possible event is a σ_- transformation. Another observation is that if t goes to ∞, then $\Gamma(t \to \infty) = 1$ and so $p_{\mathbf{r}} = 1/4 \; \forall \mathbf{r} \in \mathbb{F}_2^2$.

3.3.2 Single-qubit pure dephasing

Consider now a single qubit whose Markovian dynamics is fully described by the jump operator $L = \sqrt{\gamma_\phi/2}Z$, where γ_ϕ is the dephasing rate. With this definition of the Lindbladian, the master equation for the qubit reads:

$$\dot{\rho} = \frac{\gamma_\phi}{2}\left(Z\rho Z - \rho\right). \tag{3.28}$$

As for the amplitude damping case, we want the PTM representation of the Lindbladian. To get it we use equation 3.16 applying the Pauli matrices' properties to obtain:

$$\mathbf{L}_{ij} = \frac{\gamma_\phi}{2}\Big(\mathrm{Tr}\big[\mathbf{P}(\mathbf{r}_i)Z\,\mathbf{P}(\mathbf{r}_j)Z\big] - \mathrm{Tr}\big[\mathbf{P}(\mathbf{r}_i)\mathbf{P}(\mathbf{r}_j)\big]\Big) = \frac{\gamma_\phi}{2}\big((-1)^{x_i} - 1\big)\delta_{x_i,x_j}\delta_{z_i,z_j} \tag{3.29}$$

or, in matrix form using the Pauli basis ordered by increasing binary digits:

$$\mathbf{L} = -\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma_\phi & 0 \\ 0 & 0 & 0 & \gamma_\phi \end{pmatrix} \tag{3.30}$$

This time we are lucky because the PTM Lindbladian is diagonal. This implies that the evolution operator $e^{\mathbf{L}t}$ can be trivially computed via Taylor expansion as:

$$\begin{aligned} e^{\mathbf{L}t} &= \sum_{n=0}^{+\infty}\frac{(\mathbf{L}t)^n}{n!} = \sum_{n=0}^{+\infty}\sum_{k=0}^{3}\frac{(\mathbf{L}_{kk}\left|\mathbf{r}_k\right\rangle\left\langle\mathbf{r}_k\right|t)^n}{n!} = \\ &= \sum_{k=0}^{3} e^{\mathbf{L}_{kk}t}\left|\mathbf{r}_k\right\rangle\left\langle\mathbf{r}_k\right| = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e^{-\gamma_\phi t} & 0 \\ 0 & 0 & 0 & e^{-\gamma_\phi t} \end{pmatrix} \end{aligned} \tag{3.31}$$

where $|\mathbf{r}_k\rangle$ are the elements of the canonical basis of $\mathbb{F}_2^2$ written in symplectic notation. Equation 3.31 leads to the solution for the four components of the PTM representation of the density matrix:

$$\begin{cases} \chi_\rho(0|0,t) = \chi_\rho(0|0,0) \\ \chi_\rho(0|1,t) = \chi_\rho(0|1,0) \\ \chi_\rho(1|0,t) = \chi_\rho(1|0,0)e^{-\gamma_\phi t} \\ \chi_\rho(1|1,t) = \chi_\rho(1|1,0)e^{-\gamma_\phi t} \end{cases} \tag{3.32}$$

From 3.31 we see the PTM associated to the evolution of the state in this channel is diagonal; hence a bijection between it and a SPC exists and it can be checked it is defined by the following probabilities to plug in 3.7:

$$\begin{cases} p_{0|1} = \left(1 - e^{-\gamma_\phi t}\right)/2 \\ p_{1|0} = p_{1|1} = 0 \end{cases} \tag{3.33}$$

3.3.3 Two-qubits even-parity subspace pure dephasing

Considering now cQED platforms for quantum computing, in the dispersive parity measurement of weight-2 parity check operators there is a measurement induced dephasing within the even-parity subspace $\mathcal{H}_{even} = \text{span}\left(|00\rangle, |11\rangle\right)$ as can be deduced from [24]. The Markovian dynamics of this system can be described by a Lindblad equation defined by the jump operator:

$$L = \sqrt{\frac{\gamma_{even}}{2}} Z_{even}, \quad Z_{even} = |00\rangle\langle 00| - |11\rangle\langle 11|. \tag{3.34}$$

For the sake of simplicity in the following calculation we are going to shorten the notation in the following way $even \to e$.
We can now write the master equation as:

$$\dot{\rho} = \frac{\gamma_e}{2}\left(Z_e \rho Z_e - \frac{1}{2}\{\Pi_e, \rho\}\right), \quad \Pi_e = Z_e^2 = |00\rangle\langle 00| + |11\rangle\langle 11|. \tag{3.35}$$

Note that Π_e is the projector onto the even-parity subspace.
Using now equation 3.16 and Pauli matrices' properties we can get the following equations for the elements of the PTM of the Lindladian:

$$\mathbf{L}_{ij} = \frac{\gamma_e}{2}\left(\text{Tr}\left[\mathbf{P}(\mathbf{r}_i) Z_e \mathbf{P}(\mathbf{r}_j) Z_e\right] - \frac{1}{2}\text{Tr}\left[\mathbf{P}(\mathbf{r}_i)\{\Pi_e, \mathbf{P}(\mathbf{r}_j)\}\right]\right) =$$
$$\frac{\gamma_e}{8} i^{\mathbf{x}_i^t \mathbf{z}_i + \mathbf{x}_j^t \mathbf{z}_j}\left(\text{Tr}\left[X(\mathbf{x}_i) Z(\mathbf{z}_i) Z_e X(\mathbf{x}_j) Z(\mathbf{z}_j) Z_e\right] - \frac{1}{2}\text{Tr}\left[X(\mathbf{x}_i) Z(\mathbf{z}_i)\{\Pi_e, X(\mathbf{x}_j) Z(\mathbf{z}_j)\}\right]\right). \tag{3.36}$$

Now we can obtain the matrix elements of the Lindbladian PTM matrix and, via a Python-sympy code, we discovered that $\mathbf{L}$ matrix is the following block-diagonal matrix:

$$\mathbf{L} = \frac{\gamma_e}{4}\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 0 & 2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & -2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & -2
\end{pmatrix} \tag{3.37}$$

To obtain now the channel matrix $e^{\mathbf{L}t}$, we need to solve all the sixteen differential equations defined by the Lindblad equation in PTM form: $|\dot{\rho}\rangle\rangle = \mathbf{L}|\rho\rangle\rangle$. Luckily, the matrix is block-diagonal. The first 12×12 block – from equation 0 to equation 11 – can be solved just like in the single qubit pure dephasing case writing the exponential with its Taylor expansion; from this procedure we get a constant solution for the first four equations and a decaying exponential for the following eight equations. The last 4×4 block can be solved summing and subtracting the coupled equations; here below the method for equation 12 and equation 15 is explained.

Consider the coupled equations:

$$\begin{cases} \dot{\chi}_\rho(\mathbf{r},t) = \frac{\gamma_e}{2}\chi_\rho(\tilde{\mathbf{r}},t) - \frac{\gamma_e}{2}\chi_\rho(\mathbf{r},t) \\ \dot{\chi}_\rho(\tilde{\mathbf{r}},t) = \frac{\gamma_e}{2}\chi_\rho(\mathbf{r},t) - \frac{\gamma_e}{2}\chi_\rho(\tilde{\mathbf{r}},t) \end{cases} \tag{3.38}$$

Now sum and subtract the two equations to obtain the following equivalent system of equations:

$$\begin{cases} \dot{\chi}_\rho(\mathbf{r},t) + \dot{\chi}_\rho(\tilde{\mathbf{r}},t) = 0 \\ \dot{\chi}_\rho(\mathbf{r},t) - \dot{\chi}_\rho(\tilde{\mathbf{r}},t) = \gamma_e\chi_\rho(\tilde{\mathbf{r}},t) - \gamma_e\chi_\rho(\mathbf{r},t) \end{cases} \tag{3.39}$$

Integrating in dt the first equation and applying the conditions at $t = 0$, we obtain the equation:

$$\chi_\rho(\mathbf{r},t) + \chi_\rho(\tilde{\mathbf{r}},t) = c_+, \qquad c_+ = \chi_\rho(\mathbf{r},0) + \chi_\rho(\tilde{\mathbf{r}},0)\,.$$

Regarding the second equation, it is practical to define the following auxiliary function $\chi_\Delta(t) = \chi_\rho(\mathbf{r},t) - \chi_\rho(\tilde{\mathbf{r}},t)$, such that the equation to solve can be written as:

$$\dot{\chi}_\Delta(t) = \gamma_e\chi_\Delta(t)$$

hence, the solution is just a decaying exponential:

$$\chi_\Delta = c_\Delta e^{-\gamma_e t}, \qquad c_\Delta = \chi_\rho(\mathbf{r},0) - \chi_\rho(\tilde{\mathbf{r}},0)\,.$$

From these simple steps we obtained the following algebraic system to solve:

$$\begin{cases} \chi_\rho(\mathbf{r},t) + \chi_\rho(\tilde{\mathbf{r}},t) = \chi_\rho(\mathbf{r},0) + \chi_\rho(\tilde{\mathbf{r}},0) \\ \chi_\rho(\mathbf{r},t) - \chi_\rho(\tilde{\mathbf{r}},t) = (\chi_\rho(\mathbf{r},0) - \chi_\rho(\tilde{\mathbf{r}},0))e^{-\gamma_e t} \end{cases} \tag{3.40}$$

Now summing and subtracting again the two equations we can straightforwardly get the solution of the problem:

$$\begin{cases} \chi_\rho(\mathbf{r},t) = \frac{\chi_\rho(\mathbf{r},0)+\chi_\rho(\tilde{\mathbf{r}},0)}{2} + \frac{(\chi_\rho(\mathbf{r},0)-\chi_\rho(\tilde{\mathbf{r}},0))}{2}e^{-\gamma_e t} \\ \chi_\rho(\tilde{\mathbf{r}},t) = \frac{\chi_\rho(\mathbf{r},0)+\chi_\rho(\tilde{\mathbf{r}},0)}{2} - \frac{(\chi_\rho(\mathbf{r},0)-\chi_\rho(\tilde{\mathbf{r}},0))}{2}e^{-\gamma_e t} \end{cases} \tag{3.41}$$

For equation 13 and 14, the method used is exactly the same; the only difference is a sign to carry along the calculations. We will just report the starting system of equations:

$$\begin{cases} \dot{\chi}_\rho(\breve{\mathbf{r}},t) = -\frac{\gamma_e}{2}\chi_\rho(\hat{\mathbf{r}},t) - \frac{\gamma_e}{2}\chi_\rho(\breve{\mathbf{r}},t) \\ \dot{\chi}_\rho(\hat{\mathbf{r}},t) = -\frac{\gamma_e}{2}\chi_\rho(\breve{\mathbf{r}},t) - \frac{\gamma_e}{2}\chi_\rho(\hat{\mathbf{r}},t) \end{cases} \tag{3.42}$$

and the final solution:

$$\begin{cases} \chi_\rho(\breve{\mathbf{r}},t) = \frac{\chi_\rho(\breve{\mathbf{r}},0)-\chi_\rho(\hat{\mathbf{r}},0)}{2} + \frac{(\chi_\rho(\breve{\mathbf{r}},0)+\chi_\rho(\hat{\mathbf{r}},0))}{2}e^{-\gamma_e t} \\ \chi_\rho(\hat{\mathbf{r}},t) = -\frac{\chi_\rho(\breve{\mathbf{r}},0)-\chi_\rho(\hat{\mathbf{r}},0)}{2} + \frac{(\chi_\rho(\breve{\mathbf{r}},0)+\chi_\rho(\hat{\mathbf{r}},0))}{2}e^{-\gamma_e t} \end{cases} \tag{3.43}$$

From these results we can appreciate that both systems are stable and they converge to the mean of the starting conditions or to the signed semi-difference between them.

To insert these results in the channel evolution matrix $e^{\mathbf{L}t}$ we need to properly order the terms; isolating the terms in $\chi_\rho(\mathbf{r},t)$ and $\chi_\rho(\tilde{\mathbf{r}},t)$ we obtain, for equations 12 and 15:

$$\begin{cases} \chi_\rho(\mathbf{r},t) = \chi_\rho(\mathbf{r},0)\left(\frac{1}{2}\left(1+e^{-\gamma_e t}\right)\right) + \chi_\rho(\tilde{\mathbf{r}},0)\left(\frac{1}{2}\left(1-e^{-\gamma_e t}\right)\right) \\ \chi_\rho(\tilde{\mathbf{r}},t) = \chi_\rho(\mathbf{r},0)\left(\frac{1}{2}\left(1-e^{-\gamma_e t}\right)\right) + \chi_\rho(\tilde{\mathbf{r}},0)\left(\frac{1}{2}\left(1+e^{-\gamma_e t}\right)\right) \end{cases} \tag{3.44}$$

while for equations 13 and 14:

$$\begin{cases} \chi_\rho(\breve{\mathbf{r}},t) = \chi_\rho(\breve{\mathbf{r}},0)\left(\frac{1}{2}\left(1+e^{-\gamma_e t}\right)\right) - \chi_\rho(\hat{\mathbf{r}},0)\left(\frac{1}{2}\left(1-e^{-\gamma_e t}\right)\right) \\ \chi_\rho(\hat{\mathbf{r}},t) = -\chi_\rho(\breve{\mathbf{r}},0)\left(\frac{1}{2}\left(1-e^{-\gamma_e t}\right)\right) + \chi_\rho(\hat{\mathbf{r}},0)\left(\frac{1}{2}\left(1+e^{-\gamma_e t}\right)\right) \end{cases} \tag{3.45}$$

Defining now the quantities

$$\Gamma_e = e^{-\frac{\gamma_e t}{4}}, \qquad \Gamma_e^+ = \frac{1+e^{-\gamma_e t}}{2}, \qquad \Gamma_e^- = \frac{1-e^{-\gamma_e t}}{2} \tag{3.46}$$

we can finally write the channel matrix made of the four constant solutions, the eight decaying solutions and the two couples of coupled equations to satisfy $|\rho(t)\rangle\rangle = e^{\mathbf{L}t}|\rho(0)\rangle\rangle$ as:

$$e^{\mathbf{L}t} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e^+ & 0 & 0 & \Gamma_e^- \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e^+ & -\Gamma_e^- & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\Gamma_e^- & \Gamma_e^+ & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e^- & 0 & 0 & \Gamma_e^+ \end{pmatrix} \tag{3.47}$$

In our compact model, as we did for the amplitude damping, we assume the validity of Pauli twirling to remove the off-diagonal terms and obtain the final SPC of the compact model as:

$$e^{\mathbf{L}t}_{twirl} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e^+ & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e^+ & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e^+ & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \Gamma_e^+ \end{pmatrix} \tag{3.48}$$

This matrix has also been checked thanks to a Python-sympy code.

Comparing now the generic expression for a 2-qubit SPC 3.10 with the twirled channel matrix 3.48 and solving the linear system for the fifteen probabilities via Python-sympy we obtain the following result:

$$\begin{cases} p(1) = \frac{1}{8}(1-e^{-\gamma_e t}) \\ p(2) = \frac{1}{8}(1-e^{-\gamma_e t}) \\ p(3) = \frac{1}{8}(3+e^{-\gamma_e t} - 4e^{-\frac{\gamma_e t}{4}}) \\ p(i \in \{4,5,...,14,15\}) = 0 \end{cases} \tag{3.49}$$

Chapter 4

Study of the compact model

The goal of the thesis is to set the basics for the development of a compact model to be implemented in error correcting cycles simulations or to be a starting point to study new error-aware error correcting codes. For this purpose, the model has to be simple yet somehow realistic catching a sweet spot between complexity and precision. In this chapter most of the ideas developed will be highlighted and explained referencing to results obtained or reported in previous chapters. Some hints on future developments will be also highlighted during the discussion.

4.1 Study of the coherent state parameter

As explained in section 2.5, to obtain the coherent state parameter α_x for the two-qubit direct readout we need to solve equation 2.63 which we also report here:

$$\dot{\alpha}_x = -i(\omega_r + \chi_x)\alpha_x - i\epsilon_m(t)e^{-i\omega_m t} - \frac{\kappa}{2}\alpha_x$$

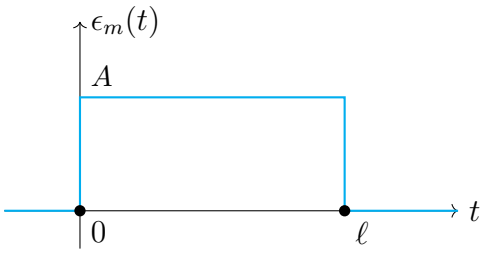
This equation can be analysed both analytically 4.1.1 and numerically 4.1.2. It is possible to obtain an analytical solution of the equation for a resonator drive envelope defined by a boxcar function. The solution will then be verified numerically and, thanks to the generality of the Python code written, the equation will also be solved for an envelope defined by a flat-top Gaussian function. This type of envelope is more similar to the ones used for real readout experiments and it is enhancing the slope to get to its regime condition. It will be also interesting to observe the difference in the dynamics between the ideal envelope and the more realistic one.

4.1.1 An analytical solution for the α_x equation

Consider now the coherent state parameter equation 2.63, and let $\epsilon_m(t)$ be defined by the boxcar function

$$\epsilon_m(t) = A\chi_\ell\left(t - \frac{\ell}{2}\right) = \begin{cases} 0, & t < 0 \\ A, & t \in [0, \ell] \\ 0, & t > \ell \end{cases} \tag{4.1}$$

or, graphically



To have cleaner calculations we now define the variable $\eta_x = \frac{k}{2} + i(\omega_r + \chi_x)$, hence the coherent state equation becomes

$$\dot{\alpha}_x(t) = -\eta_x\, \alpha_x(t) - i\epsilon_m(t)e^{-i\omega_m t}\,.$$

We now impose the boundary condition $\alpha_x(t=0) = 0$. This last condition is set because we suppose that at the beginning of the process there are no photons in the resonator, hence we are doing a small approximation compared to the real case. It can be noticed that, even if we had an initially non-zero number of photons in the cavity, the solution would be the same up to a translation due to the coherent state parameter at $t = 0$. To solve the equation the Laplace transform can be used so we remind the reader its definition and the inversion formula:

$$Y(s) = \mathcal{L}\{y\}(s) = \int_0^{+\infty} y(t)e^{-st}dt$$

$$y(t) = \mathcal{L}^{-1}\{Y\}(t) = \frac{1}{2\pi i}\lim_{T\to+\infty}\int_{\gamma-iT}^{\gamma+iT} Y(s)e^{st}dt$$

where the straight line that defines the integration domain of the inversion formula must be in the region of convergence of the Laplace transform.

Now we can apply the Laplace transform to the coherent state parameter equation and, reminding that $\mathcal{L}\{\dot{y}\}(s) = sY(s) - y(t=0)$, we obtain the following algebraic equation in the Laplace domain:

$$s\tilde{\alpha}_x(s) + \eta_x\,\tilde{\alpha}_x(s) = -i\mathcal{L}\{\epsilon_m(t)e^{-i\omega_m t}\}(s)\,. \tag{4.2}$$

The Laplace transform of the drive impulse can be computed applying the definition of the box car function and standard integration rules for exponentials; the result obtained is the following:

$$\mathcal{L}\{\epsilon_m(t)e^{-i\omega_m t}\}(s) = \frac{A(1-e^{-(i\omega_m+s)\ell})}{i\omega_m + s}$$

hence we can write the expression for $\tilde{\alpha}_x(s)$ as follows

$$\tilde{\alpha}_x(s) = \frac{-iA(1-e^{-(i\omega_m+s)\ell})}{(i\omega_m+s)(\eta_x+s)}\,. \tag{4.3}$$

Looking at the poles of this equation one could think that there is some kind of metastability for the pole at $\Re = 0$, but we can see that this pole gets cancelled by the numerator that goes to zero in that exact point, hence our system is stable and we will see it in the

plots of the solution. To proceed with the solution we now want to write the denominator in its partial fraction decomposition:

$$\frac{1}{(i\omega_m + s)(\eta_x + s)} = \frac{\bar{A}}{(i\omega_m + s)} + \frac{\bar{B}}{(\eta_x + s)}\,.$$

The values of $\bar{A}$ and $\bar{B}$ can be computed with Heaviside's cover-up method:

$$\bar{A} = (i\omega_m + s)\frac{1}{(i\omega_m + s)(\eta_x + s)}\bigg|_{s=-i\omega_m} = \frac{1}{\eta_x - i\omega_m}$$

$$\bar{B} = (\eta_x + s)\frac{1}{(i\omega_m + s)(\eta_x + s)}\bigg|_{s=-\eta_x} = \frac{-1}{\eta_x - i\omega_m}$$

hence the solution for the coherent state parameter in the Laplace domain takes the form:

$$\tilde{\alpha}_x(s) = \frac{-iA}{\eta_x - i\omega_m}\left[\frac{1 - e^{-(i\omega_m + s)\ell}}{i\omega_m + s} + \frac{e^{-(i\omega_m + s)\ell} - 1}{\eta_x + s}\right]. \tag{4.4}$$

Now we have to anti-transform this expression to obtain the time-domain representation. This can be easily done by using the linearity of the anti-transform into four easier functions to anti-transform. Defining now $u(t)$ as the Heaviside step function, its Laplace transform is $\mathcal{L}\{u\}(s) = 1/s$ and we can apply the following properties:

$$e^{\beta t} f(t) \xleftrightarrow{\mathcal{L}} F(s - \beta) \Rightarrow \mathcal{L}^{-1}\left\{\frac{1}{s - \beta}\right\} = e^{-\beta t}u(t)$$

$$f(t-\beta)u(t-\beta) \xleftrightarrow{\mathcal{L}} e^{-\beta s}F(s) \Rightarrow \mathcal{L}^{-1}\left\{e^{-\ell s}\frac{1}{\gamma + s}\right\} = e^{-\gamma(t-\ell)}u(t-\ell)u(t-\ell) = e^{-\gamma(t-\ell)}u(t-\ell)$$

and then obtain the final solution for the differential equation

$$\alpha_x(t) = \frac{-iA}{\eta_x - i\omega_m}\left\{u(t)\left[e^{-i\omega_m t} - e^{-\eta_x t}\right] + u(t-\ell)\left[e^{-i\omega_m t} - e^{-i\omega_m \ell}e^{-\eta_x(t-\ell)}\right]\right\}. \tag{4.5}$$

It is interesting to observe the physical drift of this equation because

$$\lim_{t \to +\infty} \alpha_x(t) \to 0$$

hence after dropping down the driving signal the resonator state tends to its vacuum state.

To check the validity of equation 4.5 it has been plugged in 2.63 and it has been verified that, in the sense of distributions, the solution is valid. In the next chapter we will also show a numerical double check for the solution with the boxcar function.

4.1.2 Numerical solutions for the α_x equation

To double check the analytical solution and obtain a solution for the coherent state parameter for an arbitrary envelope, a numerical simulation code has been developed. The code solves the equation in a rotating frame which is synchronized with the drive frequency ω_m, hence we defined the rotating frame $\breve{\alpha}_x$ as follows

$$\breve{\alpha}_x(t) = \alpha_x(t)e^{i\omega_m t} \tag{4.6}$$

hence the coherent state equation in this rotating frame becomes:

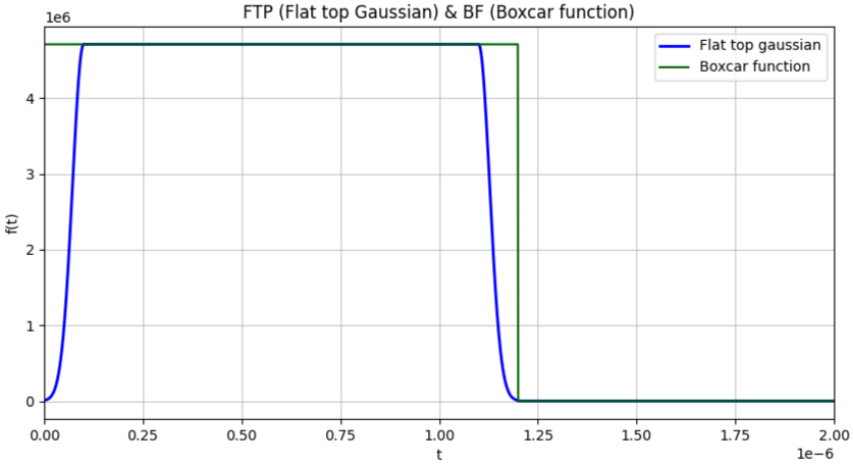
$$\dot{\breve{\alpha}}_x(t) = (i\omega_m - \eta_x)\breve{\alpha}_x(t) - i\epsilon_m(t) \tag{4.7}$$

Discretizing now the equation we obtain the following update formula:

$$\breve{\alpha}_x[n+1] = \breve{\alpha}_x[n] + \Delta t\,\{(i\omega_m - \eta_x)\breve{\alpha}_x[n] - i\epsilon_m[n]\} \tag{4.8}$$

where $\breve{\alpha}_x[n] = \breve{\alpha}_x(n\Delta t)$.

To report an example of a solution for this equation we set the following parameters: $\kappa = 2\pi \cdot 1.5\,MHz$, $g_2 = g_1 = 15\kappa$, $\chi^1 = \chi^2 = 3\kappa$ and $\omega_r = 2\pi \cdot 6\,GHz$. Considering now the envelopes $\epsilon_m(t)$ with amplitude $A = \kappa/2$ reported in the next figure



we obtain the following plots for the coherent state parameters in the rotating frame $\breve{\alpha}_x(t)$

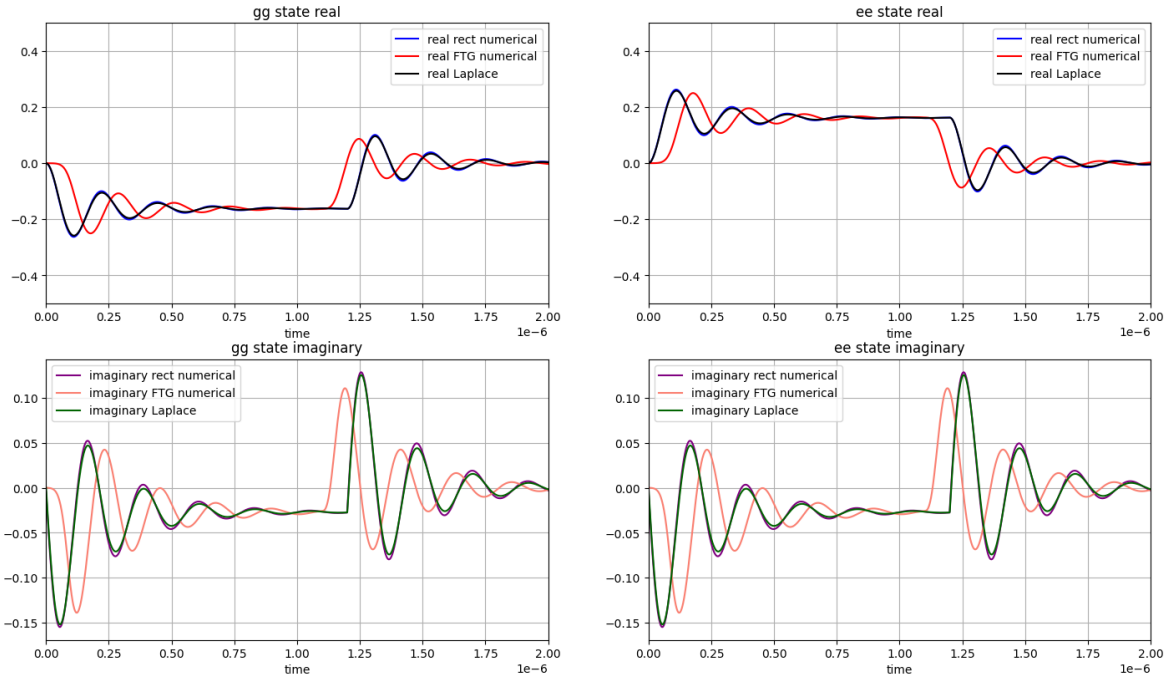


Figure 4.1: Real and imaginary part of α_x with $x \in \{00, 11\}$

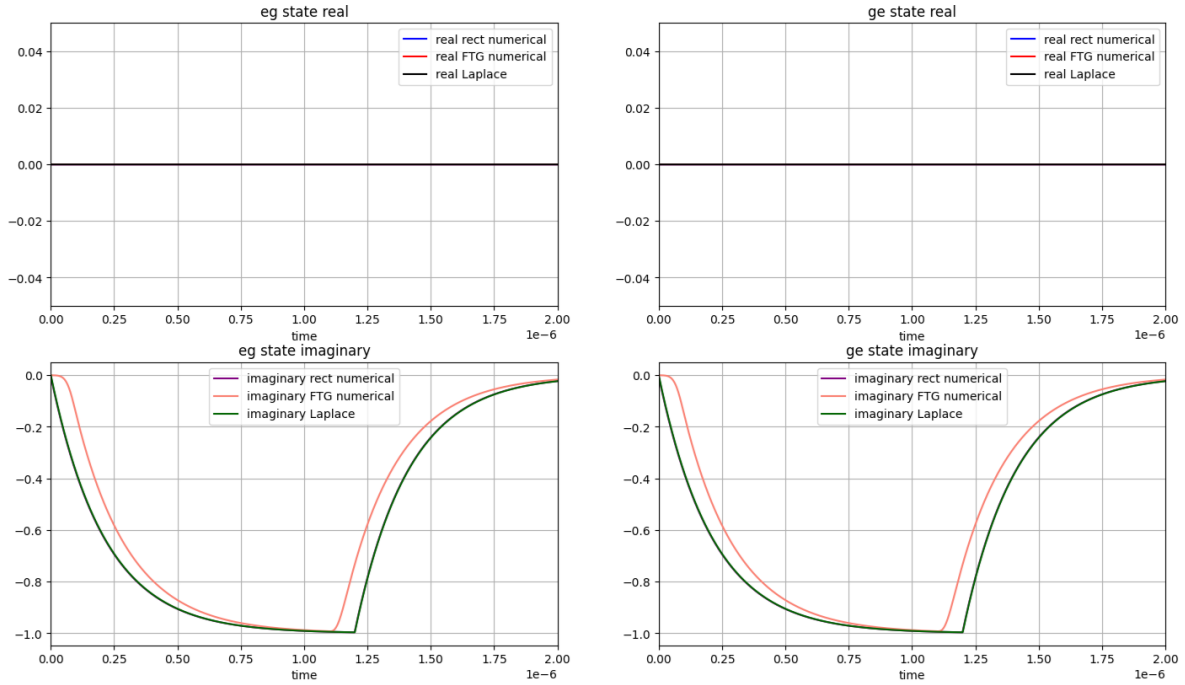


Figure 4.2: Real and imaginary part of α_x with $x \in \{10, 01\}$

From these pictures we can see that our solutions are compatible with what is reported by Lalumière et al. in [24]: states $|00\rangle$ and $|11\rangle$ have the same imaginary part and opposite real part while $|01\rangle$ and $|10\rangle$ have the same coherent state parameter. We can clearly observe that the solution of the equation with a flat top Gaussian impulse has a way slower slope compared to the boxcar solution and this introduces a sort of delay when the envelope is turned on together with a lead when the envelope is turned off. The difference in the starting slope was easily predictable considering that the differential equation is linear and the flat top Gaussian has a much more localized Fourier spectrum compared to the cardinal sine of the boxcar function. The delay and the lead were predictable because at the beginning of the measurement and when the envelope is turned off the FTG is weaker than the boxcar and so it is like if there was an equivalent effect of delay at the beginning and lead in the end of the process.

4.2 Analysis of the master equation

The master equation reported by Lalumière et al. in [24] that represents the effective evolution of the density matrix of the system we are analysing during the readout process is

$$\dot{\rho} \approx -i[\sum_{j\in\{1,2\}} \frac{\omega'_{q_j}}{2} Z_j, \rho] + \sum_{j\in\{1,2\}} \gamma_{1_j} \mathcal{D}[\sigma_-^j]\rho + \sum_{j\in\{1,2\}} \frac{\gamma_{\phi_j}}{2} \mathcal{D}[Z_j]\rho$$
$$+\kappa\mathcal{D}[\sum_{j\in\{1,2\}} \lambda_j \sigma_-^j]\rho + \sum_{x,y\in\mathbb{F}_2^2} (\Gamma_d^{xy} - iA_c^{xy})\Pi_x\rho\Pi_y \equiv \mathcal{L}\rho$$

and considering that this is the starting point of the study we decided to simplify a lot the equation and consider a simpler model that can be refined in future developments. The methods presented here can be generalized to more complicated models and we will

explain in the end how to do it. First we decided to fix the single qubit pure dephasing and single qubit amplitude damping to constant reference values of their rates γ_{1_j} and γ_{ϕ_j}, then we decided to avoid to consider the Purcell damping and the ac-Stark shift. After these considerations we considered the following dissipative master equation

$$\dot{\rho} \equiv \sum_{x,y \in \mathbb{F}_2^2} \Gamma_d^{xy} \, \Pi_x \rho \Pi_y \, . \tag{4.9}$$

From equation 4.9 we decided to simplify even more to test the workflow on a simpler case. A more rigorous justification that can lead to 4.13 can be found in Appendix E, while here we will explain a more intuitive explanation that requires a stronger assumption.

Consider the elements of the master equation 4.9 arising only from the dephsing rate $\Gamma_d^{00|11} = \Gamma_d^{11|00}$. Observing now that

$$\begin{cases} \Pi_{00} = \frac{\Pi_e + Z_e}{2} \\ \Pi_{11} = \frac{\Pi_e - Z_e}{2} \end{cases} \tag{4.10}$$

we can rewrite equation 4.9 as

$$\dot{\rho} = -\frac{\Gamma_d^{00|11}}{2} (Z_e \rho Z_e - \Pi_e \rho \Pi_e) \, . \tag{4.11}$$

Assuming now our state to be in the subspace $\text{span}\{|00\rangle, |11\rangle\}$, the application of Π_e is like an identity, hence we get the following equation

$$\dot{\rho} = -\frac{\Gamma_d^{00|11}}{2} (Z_e \rho Z_e - \rho) \, . \tag{4.12}$$

This last equation cannot be cast into a standard Lindblad equation defined by jump operators, hence we decided to stick to a similar model defined by the jump operator Z_e obtaining the following master equation

$$\dot{\rho} = \frac{\gamma_e}{2} \left(Z_e \rho Z_e - \frac{1}{2} \{\Pi_e, \rho\} \right) \tag{4.13}$$

which is the equation we studied in 3.3.3. This equation represents only an even-parity subspace dephasing; the reason why we are not considering an odd-parity subspace dephasing is that we are assuming that $\chi^1 = \chi^2$ and this leads to $\Gamma_d^{01|10} = 0$. Applying now the Pauli twirling method, the equivalent SPC of the model results to be defined by the following three mutually exclusive errors: $Z \otimes \mathbb{1}$, $\mathbb{1} \otimes Z$ and $Z \otimes Z$ with the probabilities we derived in equation 3.49 that we remark:

$$\begin{cases} p(1) = p(Z \otimes \mathbb{1}) = p(2) = p(\mathbb{1} \otimes Z) = \frac{1}{8}(1 - e^{-\gamma_e t}) \\ p(3) = p(Z \otimes Z) = \frac{1}{8}(3 + e^{-\gamma_e t} - 4e^{-\frac{\gamma_e t}{4}}) \end{cases} \tag{4.14}$$

To compute these probabilities we assumed $\gamma_e = -\Gamma_d^{00|11}$ to have a first order estimation of the parameter. The sign is due to the fact that the master equation written in Lalumière's paper [24] is not written in Lindblad form (defined by a set of jump operators) and expanding the terms it can be seen that $\Gamma_d^{00|11}$ is negative, hence must be taken positive.

Estimation of the even-parity dephasing rate

From the solution of equation 2.63 it is clear that Γ_d^{xy} is time-dependent during the measurement. To obtain a first model we can just assume the measurement time is sufficiently large that for most of the time the coherent state parameter is in its regime value, hence from equation 4.5 we can approximate α_x to its steady value:

$$\alpha_x^{steady} = \frac{-iA}{\eta_x - i\omega_m} e^{i\xi} \tag{4.15}$$

where ξ is a time dependent phase. This implies that we can express γ_e as follows

$$\gamma_e \approx -(\chi_{00} - \chi_{11})\, \Im\left[\frac{A^2}{(\eta_{00} - i\omega_m)(\eta_{11}^* + i\omega_m)}\right]. \tag{4.16}$$

From the last equation we observe that

$$\Im\left[\frac{A^2}{(\eta_{00} - i\omega_m)(\eta_{11}^* + i\omega_m)}\right] = -\frac{A^2\Delta_\chi\kappa/2}{((k/2)^2 + \Delta_\omega^2 + \Delta_\omega\chi_\Sigma + \chi_\Pi)^2 + (\Delta_\chi\kappa/2)^2}$$

with $\Delta_\omega = \omega_r - \omega_m$, $\Delta_\chi = \chi_{00} - \chi_{11}$, $\chi_\Sigma = \chi_{00} + \chi_{11}$, $\chi_\Pi = \chi_{00}\chi_{11}$; this implies that:

$$\gamma_e = \frac{A^2\Delta_\chi^2\kappa/2}{((\kappa/2)^2 + \Delta_\omega^2 + \Delta_\omega\chi_\Sigma + \chi_\Pi)^2 + (\Delta_\chi\kappa/2)^2}. \tag{4.17}$$

In our specific case, we set $\omega_r = \omega_m$ and $\chi_{00} = -\chi_{11} \equiv \tilde{\chi} = \chi^1 + \chi^2 = 2\chi$ holds, hence:

$$\gamma_e = \frac{A^2 2\tilde{\chi}^2\kappa}{((\kappa/2)^2 - \tilde{\chi}^2)^2 + (\tilde{\chi}\kappa)^2} = \frac{8A^2\chi^2\kappa}{(\kappa^2/4 - 4\chi^2)^2 + (2\chi\kappa)^2}. \tag{4.18}$$

Another slightly more precise estimation could be done considering the average value of α_x during the whole measurement process, but from the plots we can see that the value will be more or less consistent with the steady state value. Hence we decided to proceed with the analytical result to help us to formulate closed form expressions.

We report the plot of γ_e setting $\kappa = \chi = 2\pi \cdot 3MHz$ sweeping A from zero to 1.4κ.

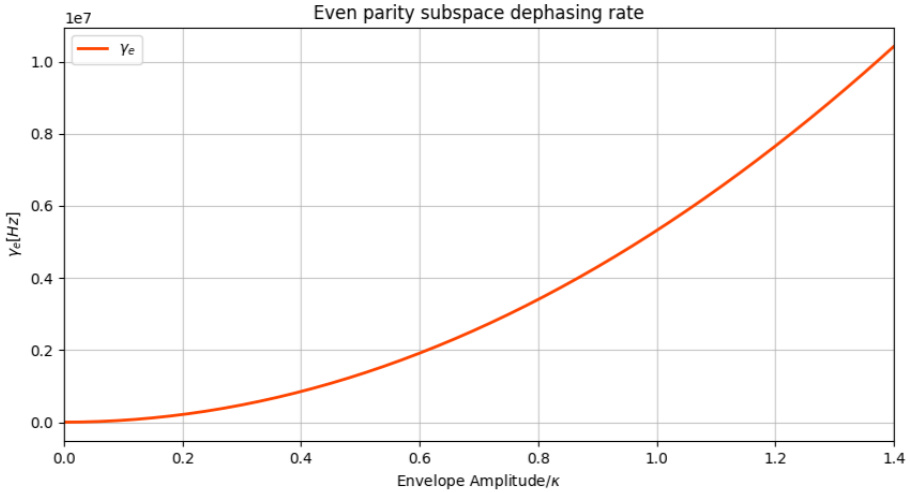


Figure 4.3: γ_e plots as a function of drive amplitude A normalized to κ.

Hints on how to generalize the result in future studies

A possible approach to go beyond the constant rate approximation could be to consider the PTM expression of the right-hand side of one equation between 3.35,4.9 or even 2.60 and then solve numerically the sixteen differential equations starting from the numerical or analytical solution of the coherent state equation. Once these equations are solved we could obtain the matrix elements of the channel matrix in PTM form as numerical functions, then we must trace out all the off-diagonal elements of the channel matrix via Pauli Twirling and obtain the desired SPC that defines the error model. For time constraints this analysis has not been done, but could be a possible way to obtain a more accurate model.

4.2.1 An estimation of the equivalent error model

To propose a first model we decided to fix the single qubit errors to reference rates that are common in cQED platforms. To do so we fixed the relaxation time $T_1 = 250\,\mu s$ and the pure dephasing rate $\gamma_\phi = 2\pi \cdot 3\,kHz$. Setting now the measurement time to $300\,ns$ and the time interval between one measurement and the next one to $100\,ns$ we obtain the following probabilities associated to single qubit error mechanisms for each measurement cycle:

$$\begin{cases} p_Z^{\gamma_\phi} \approx 0.0038 \\ p_Z^{\gamma_1} \approx 6.28 \cdot 10^{-6} \\ p_X^{\gamma_1} = p_Y^{\gamma_1} \approx 0.0025 \end{cases} \tag{4.19}$$

Regarding the measurement induced errors, we stick to the constant coherent state parameter approximation we explained in the previous paragraph and we set $\chi^j = \kappa$ and $g_j = 15\kappa$. Fixing then $\kappa = 2\pi \cdot 3MHz$ like we did for the γ_e plot in Fig. 4.3 we report in Fig. 4.4 the measurement induced probabilities plots as a function of the drive amplitude normalized to the photon decay rate κ following equations 4.14:

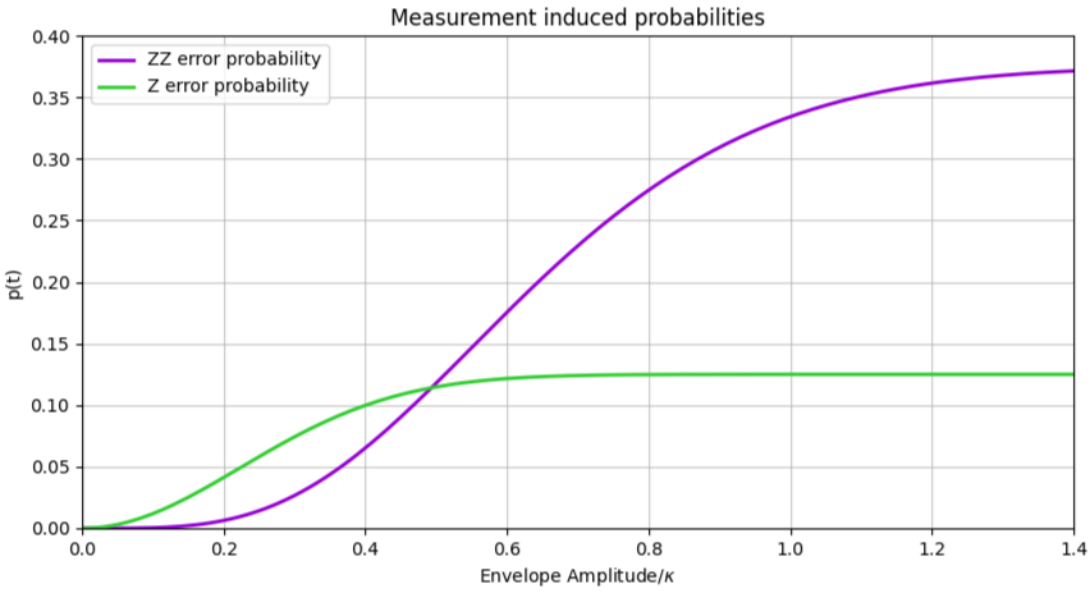


Figure 4.4: Measurement induced ZZ – the purple line – and Z – the green line – error probabilities as a function of the drive envelope amplitude normalized to κ.

From Fig. 4.4 we can clearly see that the higher the amplitude of the envelope the higher is the error rate. This was predictable because we are interacting a lot more with the system, hence we are contributing to error mechanisms. Another interesting thing to observe is that for low amplitudes the dominant error type is the Z error, while for $A \geq 0.5\kappa$ it saturates and the ZZ error takes the lead.

For completeness we also report the plot of the coherent state parameters α_{00} and α_{11} that would be needed to compute γ_e without assuming the steady state approximation.

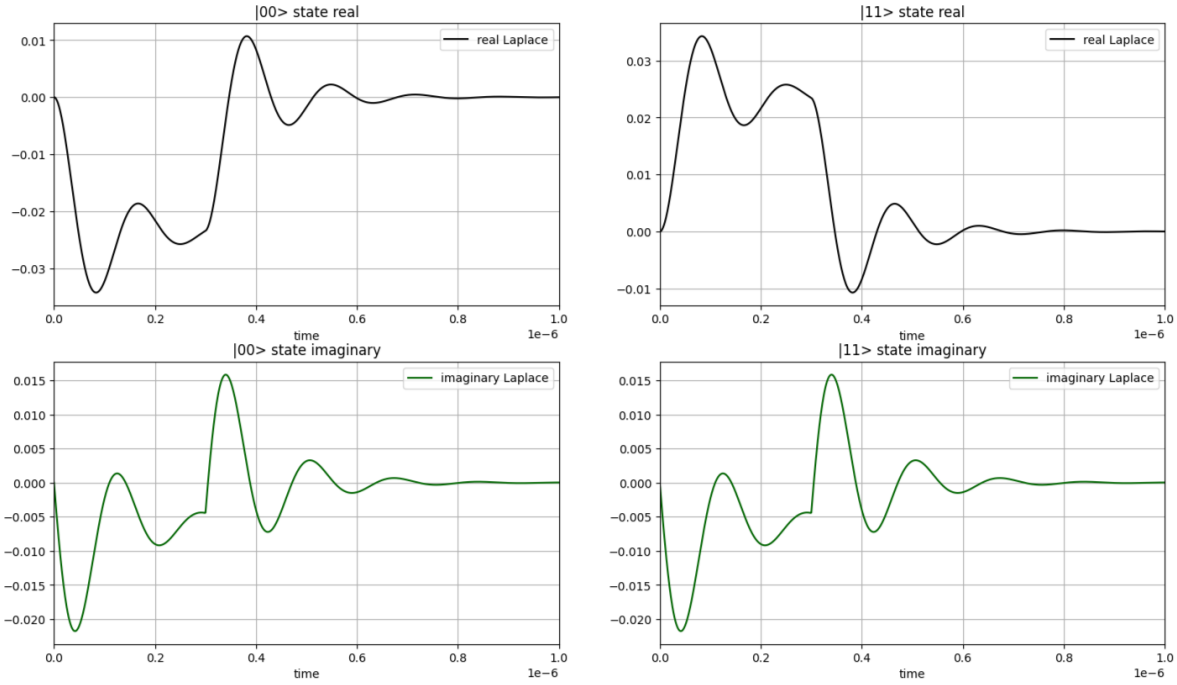


Figure 4.5: The plots on the left side represent the real and imaginary part of the coherent state parameter α_{00} while the plots on the right side represent the real and imaginary part of the coherent state parameter α_{11}. To obtain the plots we set $\omega_r = 2\pi \cdot 6\,GHz$.

4.2.2 Threshold simulations for the honeycomb code with the simplified compact model

To study how the honeycomb code can perform with the previously developed error models, we used a Python code obtained by introducing some modifications to the script by Craig Gidney and available on GitHub Gist [9]. This code is based on the library developed by the same author called Stim [10] and the decoding library developed by Oscar Higgott [15] and then by Craig Gidney too [16] called Pymatching. Pymatching implements the Minimum Weight Perfect Matching (MWPM) decoder, which is the stardard decoder used for surface-like codes. From this implementation we modified the decoding section, the error model applied to the circuit and the plots section. Regarding the decoding section we decided to stick with the same MWPM decoder used in Gidney's script and compared to his solution we changed the graph construction using an automatic method implemented in Pymatching and we enabled the decoding of correlated errors like the ZZ errors contained in the compact model of measurement induced errors.

In these threshold simulations we decided to set on the x-axis the drive amplitude normalized to κ because both the ZZ and Z type error probabilities increase with the envelope amplitude A. The results are obtained simulating eighteen noisy rounds with perfect measurements and collecting the statistics comparing the predicted result with the measured result of the vertical outer logical operator $\mathcal{O}_0$ 1.8 of the honeycomb code over the results of 10^4 executions of the error correcting cycles for four different scaling factors of the square honeycomb tessellation. In the threshold plot in Fig. 4.6 we report the results obtained with an error model containing both the single-qubit dissipative processes not associated to the measurement with probabilities fixed and highlighted in 4.19 together with the signal amplitude dependent SPC with probabilities plotted in 4.4 associated to the measurement.

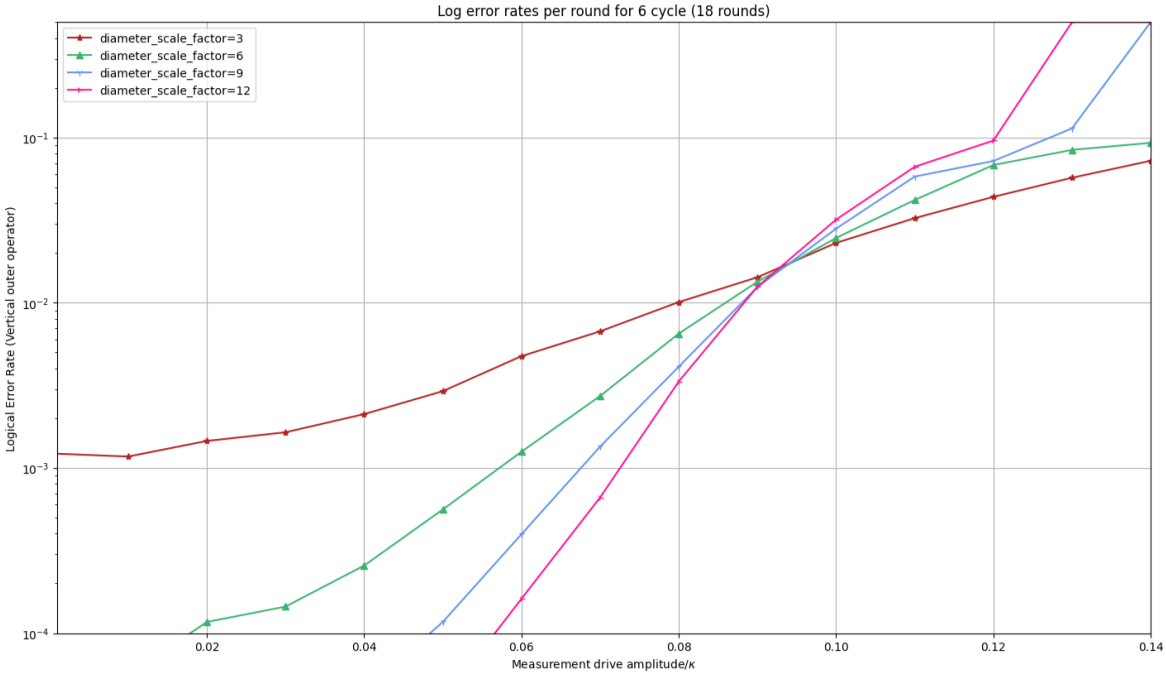


Figure 4.6: Results of the threshold simulation of the honeycomb code decoded with correlated MWPM with measurement induced even parity subspace dephasing and non-measurement related single qubit errors.

As we can in the previous figure from the crossing of the curves, the threshold is around $A = 0.095\kappa$ and above this value it is no more useful to increase the tessellation of qubits.

We also studied how the noise floor introduced by the single qubit non-measurement dependent errors impacted on the threshold and the result is Fig. 4.7.

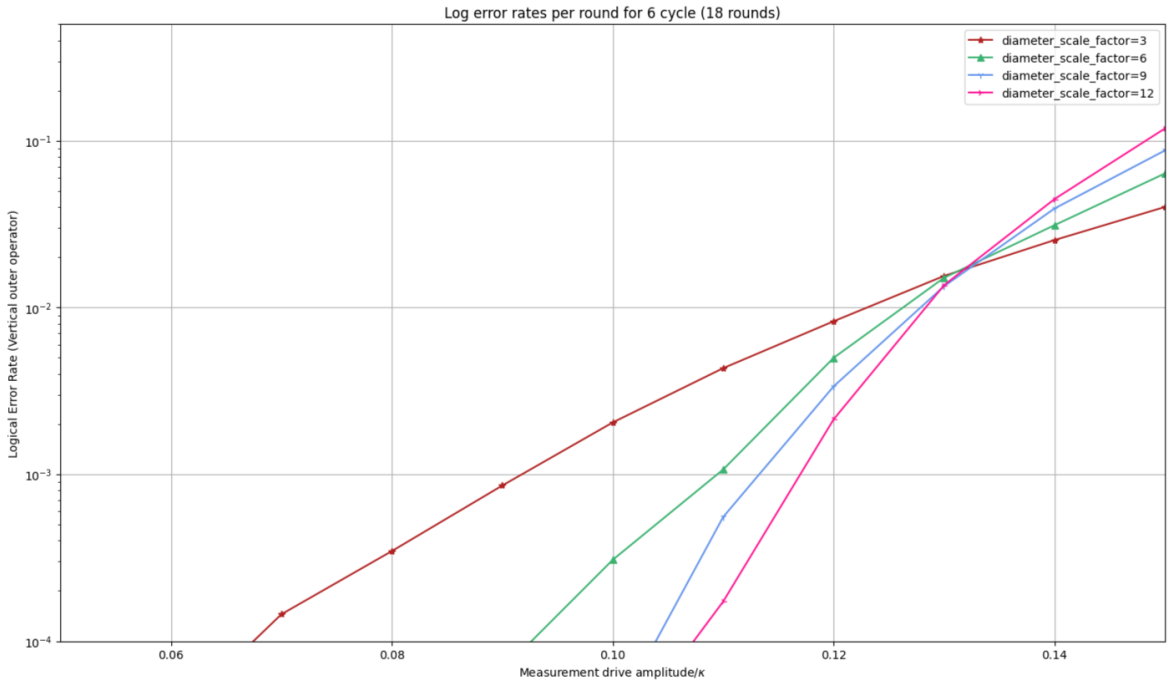


Figure 4.7: Results of the threshold simulation of the honeycomb code decoded with correlated MWPM with measurement induced even parity subspace dephasing.

It is clear from the previous figure that the noise we removed had somehow an impact because the threshold is around $A \approx 0.13\kappa$.

Conclusions

In this thesis we studied how to build error models to be implementable in error correcting cycles. The methods have been applied to a simplified measurement induced error in direct weight-2 parity measurements – an even-parity subspace dephasing mechanism – to extract an equivalent model that has been implemented in an error correcting cycle simulation of the honeycomb code obtaining a threshold plot that pairs the logical error rate with the drive amplitude. The expressions of the model for the even-parity subspace dephasing considered showed that the lower the drive amplitude the more dominant are the Z-type errors compared to the ZZ-type ones and the opposite for high values of the amplitude. From the simulation results we can see that the threshold is in the region where the single qubit Z errors dominate because the crossing of the logical error rates is at $A \approx 0.095\kappa$. An analysis on the various approximations has been made and some hints on how to continue the study have been given, for example how to consider time varying dephasing rates. The workflow we used is quite general and can be applied to a more general master equation to obtain an equivalent stochastic Pauli channel. Clearly the study can also be upgraded from the point of view of the Python implementation considering that the same models can be tested using other type of decoders like BPOSD or considering also measurement fidelity errors, i.e. when you measure a $|1\rangle$ instead of a $|0\rangle$ or vice versa. Another very interesting future development could also be to apply this workflow to a complete master equation to obtain an equivalent error model that captures the majority of the errors that happen in the system and then propose an error-aware error correcting code that can surpass the standard error correcting codes for this specific type of errors. Last but not least, it would be very interesting to do a proper comparison between the direct dispersive parity measurements we analysed with the standard measurement done with ancilla qubits.

Bibliography

[1] Alexandre Blais et al. "Circuit quantum electrodynamics". In: *Reviews of Modern Physics* 93.2 (May 2021). ISSN: 1539-0756. DOI: 10.1103/revmodphys.93.025005. URL: http://dx.doi.org/10.1103/RevModPhys.93.025005.

[2] H. Bombin and M. A. Martin-Delgado. "Homological error correction: Classical and quantum codes". In: *Journal of Mathematical Physics* 48.5 (May 2007). ISSN: 1089-7658. DOI: 10.1063/1.2731356. URL: http://dx.doi.org/10.1063/1.2731356.

[3] S. B. Bravyi and A. Yu. Kitaev. *Quantum codes on a lattice with boundary*. 1998. arXiv: quant-ph/9811052 [quant-ph]. URL: https://arxiv.org/abs/quant-ph/9811052.

[4] Heinz-Peter Breuer and Francesco Petruccione. *The Theory of Open Quantum Systems*. Oxford University Press, Jan. 2007. ISBN: 9780199213900. DOI: 10.1093/acprof:oso/9780199213900.001.0001. URL: https://doi.org/10.1093/acprof:oso/9780199213900.001.0001.

[5] Senrui Chen et al. "Robust Shadow Estimation". In: *PRX Quantum* 2 (Sept. 2021), p. 030348. DOI: 10.1103/PRXQuantum.2.030348. URL: https://link.aps.org/doi/10.1103/PRXQuantum.2.030348.

[6] Alessandro Ciani, David P. DiVincenzo, and Barbara M. Terhal. *Lecture Notes on Quantum Electrical Circuits*. TU Delft OPEN Publishing, Feb. 2024. ISBN: 9789463668156. DOI: 10.59490/tb.85. URL: http://dx.doi.org/10.59490/tb.85.

[7] Richard P. Feynman. "Simulating physics with computers". In: *International Journal of Theoretical Physics* 21.6 (June 1982), pp. 467–488. DOI: 10.1007/BF02650179. URL: https://doi.org/10.1007/BF02650179.

[8] Austin G. Fowler et al. "Surface codes: Towards practical large-scale quantum computation". In: *Physical Review A* 86.3 (Sept. 2012). ISSN: 1094-1622. DOI: 10.1103/physreva.86.032324. URL: http://dx.doi.org/10.1103/PhysRevA.86.032324.

[9] Craig Gidney. *Estimating the threshold of a new quantum code using stim and pymatching.* https://gist.github.com/Strilanc/a4a5f2f9410f84212f6b2c26d9e46e24. 2021.

[10] Craig Gidney. "Stim: a fast stabilizer circuit simulator". In: *Quantum* 5 (July 2021), p. 497. ISSN: 2521-327X. DOI: 10.22331/q-2021-07-06-497. URL: http://dx.doi.org/10.22331/q-2021-07-06-497.

[11] Lov K. Grover. *A fast quantum mechanical algorithm for database search*. 1996. arXiv: quant-ph/9605043 [quant-ph]. URL: https://arxiv.org/abs/quant-ph/9605043.

[12] Akel Hashim et al. "Practical Introduction to Benchmarking and Characterization of Quantum Computers". In: *PRX Quantum* 6.3 (Aug. 2025). ISSN: 2691-3399. DOI: 10.1103/prxquantum.6.030202. URL: http://dx.doi.org/10.1103/PRXQuantum.6.030202.

[13] Matthew B. Hastings and Jeongwan Haah. "Dynamically Generated Logical Qubits". In: *Quantum* 5 (Oct. 2021), p. 564. ISSN: 2521-327X. DOI: 10.22331/q-2021-10-19-564. URL: http://dx.doi.org/10.22331/q-2021-10-19-564.

[14] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002. ISBN: 9780521795401. URL: https://books.google.it/books?id=BjKs86kosqgC.

[15] Oscar Higgott. *PyMatching: A Python package for decoding quantum codes with minimum-weight perfect matching*. 2021. arXiv: 2105.13082 [quant-ph]. URL: https://arxiv.org/abs/2105.13082.

[16] Oscar Higgott and Craig Gidney. "Sparse Blossom: correcting a million errors per core second with minimum-weight matching". In: *Quantum* 9 (Jan. 2025), p. 1600. ISSN: 2521-327X. DOI: 10.22331/q-2025-01-20-1600. URL: http://dx.doi.org/10.22331/q-2025-01-20-1600.

[17] I.M. Isaacs. *Algebra: A Graduate Course*. Graduate studies in mathematics. American Mathematical Society, 2009. ISBN: 9780821847992. URL: https://books.google.it/books?id=5tKq0kbHuc4C.

[18] B.D. Josephson. "Possible new effects in superconductive tunnelling". In: *Physics Letters* 1.7 (1962), pp. 251–253. ISSN: 0031-9163. DOI: https://doi.org/10.1016/0031-9163(62)91369-0. URL: https://www.sciencedirect.com/science/article/pii/0031916362913690.

[19] A.Yu. Kitaev. "Fault-tolerant quantum computation by anyons". In: *Annals of Physics* 303.1 (Jan. 2003), pp. 2–30. ISSN: 0003-4916. DOI: 10.1016/s0003-4916(02)00018-0. URL: http://dx.doi.org/10.1016/S0003-4916(02)00018-0.

[20] Emanuel Knill and Raymond Laflamme. "Theory of quantum error-correcting codes". In: *Phys. Rev. A* 55 (Feb. 1997), pp. 900–911. DOI: 10.1103/PhysRevA.55.900. URL: https://link.aps.org/doi/10.1103/PhysRevA.55.900.

[21] Jens Koch et al. "Charge-insensitive qubit design derived from the Cooper pair box". In: *Physical Review A* 76.4 (Oct. 2007). ISSN: 1094-1622. DOI: 10.1103/physreva.76.042319. URL: http://dx.doi.org/10.1103/PhysRevA.76.042319.

[22] Jin Ming Koh et al. *Entangling logical qubits without physical operations*. 2026. arXiv: 2601.20927 [quant-ph]. URL: https://arxiv.org/abs/2601.20927.

[23] P. Krantz et al. "A quantum engineer's guide to superconducting qubits". In: *Applied Physics Reviews* 6.2 (June 2019), p. 021318. ISSN: 1931-9401. DOI: 10.1063/1.5089550. eprint: https://pubs.aip.org/aip/apr/article-pdf/doi/10.1063/1.5089550/20722375/021318_1_1.5089550.pdf. URL: https://doi.org/10.1063/1.5089550.

[24] Kevin Lalumière, J. M. Gambetta, and Alexandre Blais. "Tunable joint measurements in the dispersive regime of cavity QED". In: *Physical Review A* 81.4 (Apr. 2010). ISSN: 1094-1622. DOI: 10.1103/physreva.81.040301. URL: http://dx.doi.org/10.1103/PhysRevA.81.040301.

[25] D.A. Lidar and T.A. Brun. *Quantum Error Correction*. Cambridge University Press, 2013. ISBN: 9780521897877. URL: https://books.google.de/books?id=bM5KngEACAAJ.

[26] M.A. Nielsen and I.L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010. ISBN: 9781107002173. URL: https://books.google.de/books?id=j2ULnwEACAAJ.

[27] Anthony Ryan O'Rourke and Simon Devitt. *Compare the Pair: Rotated vs. Unrotated Surface Codes at Equal Logical Error Rates*. 2024. arXiv: 2409.14765 [quant-ph]. URL: https://arxiv.org/abs/2409.14765.

[28] David Poulin. "Stabilizer Formalism for Operator Quantum Error Correction". In: *Physical Review Letters* 95.23 (Dec. 2005). ISSN: 1079-7114. DOI: 10.1103/physrevlett.95.230504. URL: http://dx.doi.org/10.1103/PhysRevLett.95.230504.

[29] J. J. Sakurai and Jim Napolitano. *Modern Quantum Mechanics*. 3rd ed. Cambridge University Press, 2020. DOI: 10.1017/9781108587280.

[30] M Schumann, F K Wilhelm, and A Ciani. "Emergence of noise-induced barren plateaus in arbitrary layered noise models". In: *Quantum Science and Technology* 9.4 (Aug. 2024), p. 045019. ISSN: 2058-9565. DOI: 10.1088/2058-9565/ad6285. URL: http://dx.doi.org/10.1088/2058-9565/ad6285.

[31] Peter W. Shor. "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". In: *SIAM Journal on Computing* 26.5 (Oct. 1997), pp. 1484–1509. ISSN: 1095-7111. DOI: 10.1137/s0097539795293172. URL: http://dx.doi.org/10.1137/S0097539795293172.

[32] 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.

[33] Christophe Vuillot, Alessandro Ciani, and Barbara M. Terhal. "Homological Quantum Rotor Codes: Logical Qubits from Torsion". In: *Communications in Mathematical Physics* 405.2 (Feb. 2024). ISSN: 1432-0916. DOI: 10.1007/s00220-023-04905-4. URL: http://dx.doi.org/10.1007/s00220-023-04905-4.

[34] A. Zee. *Group Theory in a Nutshell for Physicists*. Princeton, NJ: Princeton Univeristy Press, 2016.

Appendices

Appendix A

Classical linear codes

The goal of this appendix is to compress all the needed knowledge about classical linear codes used chapter 1 and its contents are based on [26]. All the arithmetic done in this appendix will be mod2.

A linear code space $\mathcal{C}$ encodes k information bits in an n-bit code space; these codes are also referred as $[n, k]$ linear codes. A common formalism for this type of code is the so-called Parity check matrix H formalism that can be defined as follows

Let $H \in \mathbb{F}_2^{(n-k)\times n}$ be the Parity check matrix. The $[n, k]$ code space is defined as the kernel of H. To ensure the total representation of the 2^k codewords the condition $\text{rank}(H) = n - k$ must be verified. This condition is imposed by the rank-nullity theorem because it states that $\text{nullity}(H) = \dim(\mathbb{F}_2^n) - \text{rank}(H)$. Hence, to get $\text{nullity}(H) = k$, $\text{rank}(H)$ must be equal to $n - k$.

Appendix B

Kraus representation of operators

In this appendix we will derive the formula to obtain the Kraus operator sum representation of operators, or simply Kraus representation, from a generic Hamiltonian. The contents of this appendix are completely based on [25] and you can find there more details on the matter.

Consider the total Hilbert space of a quantum system $\mathcal{H}_{SB} = \mathcal{H}_S \otimes \mathcal{H}_B$ where $\mathcal{H}_S$ is the Hilbert space associated to the quantum system we are interested in and $\mathcal{H}_B$ is the Hilbert space associated to the environment, usually referred to as bath. Suppose now that the dynamics of the whole system is described by the Hamiltonian H_{SB}. This Hamiltonian can be decomposed as a sum of three terms such that

$$H_{SB} = H_S \otimes \mathbb{1}_B + \mathbb{1}_S \otimes H_B + H_I \tag{B.1}$$

where H_S and H_B are the non interacting Hamiltonians of the system and the bath and H_I is the interaction Hamiltonian. The interaction term can always be written as

$$H_I = \sum_\alpha S_\alpha \otimes B_\alpha \tag{B.2}$$

in which S_α and B_α are operators acting respectively on $\mathcal{H}_S$ and $\mathcal{H}_B$. From Hamiltonian B.1 it is possible to obtain the unitary time evolution operator, often referred to as the propagator, that we indicate with U such that we can express the evolution of the system-bath density operator from time $t = 0$ to a generic time t as follows:

$$\rho_{SB}(t) = U(t)\rho_{SB}(0)U^\dagger(t)\,. \tag{B.3}$$

To isolate from the total evolution the informations about the system S we are interested in and obtain the so-called reduced density operator ρ_S we need to apply the partial trace operator. Choosing now, for example, a basis of $\mathcal{H}_B$ denoted as $\{\mu_B\}_\mu$ we define the partial trace operator with respect to $\mathcal{H}_B$ as

$$\mathrm{Tr}_B[\cdot] = \sum_\mu \langle \mu_B | \ \cdot \ | \mu_B \rangle \tag{B.4}$$

hence the reduced density operator of the system S reads:

$$\rho_S(t) = \mathrm{Tr}_B[U(t)\rho_{SB}(0)U^\dagger(t)]\,. \tag{B.5}$$

Assuming now that for $t = 0$ the system is completely decoupled from the bath such that:

$$\rho_{SB}(0) = \rho_S(0) \otimes \rho_B(0) \tag{B.6}$$

where ρ_S and ρ_B are the reduced density operators obtained applying respectively the partial trace operators respect to $\mathcal{H}_B$ and $\mathcal{H}_S$ to ρ_{SB}, we can compute the reduced density operator of the system S at time t as follows:

$$\begin{gathered}\rho_S(t) = \sum_{\mu} \langle \mu_B | \, U(t) \, (\rho_S(0) \otimes \rho_B(0)) \, U^{\dagger}(t) \, | \mu_B \rangle = \\ \sum_{\mu} \langle \mu_B | \, U(t) \left(\rho_S(0) \otimes \sum_{\nu} \lambda_\nu \, | \nu_B \rangle \, \langle \nu_B | \right) U^{\dagger}(t) \, | \mu_B \rangle = \sum_{a=(\mu,\nu)} E_a \, \rho_S(0) \, E_a^{\dagger}\end{gathered} \tag{B.7}$$

where the reduced density matrix of the bath at $t = 0$ has been expanded via spectral decomposition as $\rho_B(0) = \sum_\nu \lambda_\nu \, | \nu_B \rangle \, \langle \nu_B |$ and the operators E_a which are defined by $E_a = \sqrt{\lambda_\nu} \, \langle \mu_B | \, U \, | \nu_B \rangle$ are denoted as Kraus operators. After these consideration we can state that once we know the Kraus operators E_a associated to a system-bath Hamiltonian, the associated quantum channel $\mathcal{E}$ is completely characterized.

Appendix C

Symplectic representation

In this appendix we will introduce the symplectic representation: this particular way of expressing Pauli operators is very useful because it maps the study of non-commutative matrices to simple operations between binary vectors. The contents of this chapter are inspired from [26], but we also suggest to the reader to check the appendix A in the recent work from Jin Ming Koh et al. [22].

Consider now the single qubit Pauli operators:

$$\begin{aligned} \mathbb{1} &= \begin{pmatrix} +1 & 0 \\ 0 & +1 \end{pmatrix}, \qquad Z = \begin{pmatrix} +1 & 0 \\ 0 & -1 \end{pmatrix}, \\ X &= \begin{pmatrix} 0 & +1 \\ +1 & 0 \end{pmatrix}, \qquad Y = \begin{pmatrix} 0 & -i \\ +i & 0 \end{pmatrix}. \end{aligned} \tag{C.1}$$

We define the single-qubit Pauli group $\mathcal{G}_1$ [26] as:

$$\mathcal{G}_1 = \{\pm\mathbb{1}, \pm i\mathbb{1}, \pm X, \pm iX, \pm Z, \pm iZ, \pm Y, \pm iY\} = \langle X, Z, Y\rangle \ . \tag{C.2}$$

Observing now that the relations $XYZ = i\mathbb{1}$ and $X^2 = Z^2 = Y^2 = \mathbb{1}$ hold, we can think about representing a Pauli operator with only two of the Four Pauli operators as a product of two independent Paulis multiplied by a phase. It turns out that, defining $\mathbf{r} = (x, z)^t \in \mathbb{F}_2^2$, we can express a single qubit Pauli operator as:

$$P(\mathbf{r}) = P(x|z) = i^{xz} X^x Z^z \tag{C.3}$$

this last equation maps the single qubit Pauli operators in a binary vector $\mathbf{r} \in \mathbb{F}_2^2$ which is called symplectic vector or symplectic representation of the single-qubit Pauli operators. This concept can be simply generalized to the n-qubit Pauli operators such that:

$$P(\mathbf{r}) = P(\mathbf{x}|\mathbf{z}) = i^{\mathbf{x}^t\mathbf{z}} X(\mathbf{x}) Z(\mathbf{z}), \quad \mathbf{r} = \begin{pmatrix} \mathbf{x} \\ \mathbf{z} \end{pmatrix} \in \mathbb{F}_2^{2n} . \tag{C.4}$$

In this last definition $X(\mathbf{x})$ and $Z(\mathbf{z})$ are constructed by the following expressions:

$$X(\mathbf{x}) = \bigotimes_{j\in\{1,\dots,n\}} X^{x_j}, \qquad Z(\mathbf{z}) = \bigotimes_{j\in\{1,\dots,n\}} Z^{z_j} \tag{C.5}$$

where x_j and z_j represent the j-th component of the x-part and the z-part of the n-qubit symplectic vector $\mathbf{r}$.

The n-qubit Pauli group can then be cast in a symplectic fashion as follows:

$$\mathcal{G}_n = \{ i^k P(\mathbf{r}) \ : \ \mathbf{r} \in \mathbb{F}_2^{2n}, k \in \{0,1,2,3\} \} \, . \tag{C.6}$$

Within the symplectic representation we can also map the study of the commutativity of Pauli operators to a particular binary inner product called symplectic inner product. First of all we need to define the so-called symplectic matrix associated to the n-qubit Pauli group Ω_n as follows:

$$\Omega_n = \begin{pmatrix} \mathbf{0}_{n\times n} & \mathbb{1}_{n\times n} \\ \mathbb{1}_{n\times n} & \mathbf{0}_{n\times n} \end{pmatrix} \tag{C.7}$$

where $\mathbf{0}_{n\times n}$ is an $n \times n$ matrix full of zeros and $\mathbb{1}_{n\times n}$ is the $n \times n$ identity matrix.

Let now $\mathbf{r}$ and $\mathbf{s}$ be elements of $\mathbb{F}_2^{2n}$, we can define the symplectic inner product between them as follows:

$$\langle \mathbf{r}, \mathbf{s} \rangle_\Omega = \mathbf{r}^t \, \Omega_n \, \mathbf{s} \mod 2 \, . \tag{C.8}$$

Now suppose that $\mathbf{r}$ and $\mathbf{s}$ are the symplectic representations of two n-qubit Pauli operators; it is easy to verify that if $\langle \mathbf{r}, \mathbf{s} \rangle_\Omega = 0$ then $[P(\mathbf{r}), P(\mathbf{s})] = 0$ while if $\langle \mathbf{r}, \mathbf{s} \rangle_\Omega = 1$ then $\{P(\mathbf{r}), P(\mathbf{s})\} = 0$.

Appendix D

Useful notions for quantum error correction

The goal of this appendix is simply to report some important results used in chapter 1 for the derivation of other theorems or for doing some considerations regarding quantum error correcting codes. The concepts are based on [26], [25] and [17]; we suggest the reader to consult them for more details.

Definition D.0.1 ([17], [25])**.** *Consider a group A and a set B such that $B \subseteq A$, we define the* ***centralizer*** *of B in A the set:*

$$\mathcal{C}_A(B) = \{a \in A \,:\, \forall b \in B,\, ab = ba\}\,. \tag{D.1}$$

If $A = B$, then the centralizer $\mathcal{C}_A(A)$ is called ***center*** *and is denoted as $\mathcal{C}(A)$.*

Definition D.0.2 ([17], [25])**.** *Consider a group A and a set B such that $B \subseteq A$, we define the* ***normalizer*** *of B in A the set:*

$$\mathcal{N}_A(B) = \{a \in A \,:\, a^{-1} B\, a = a\}\,. \tag{D.2}$$

If $A = U(2^n)$, with $U(2^n)$ being the unitary group on n qubits, and $B = \mathcal{G}_n$ then the normalizer $\mathcal{N}_{U(2^n)}(\mathcal{G}_n)$ is called the ***Clifford group*** *and his elements are referred to as Clifford operators.*

Theorem D.0.1 (**Unitary freedom in Kraus representation** [26])**.** *Let $\mathcal{E}$ and $\mathcal{F}$ be quantum operations that can be written in Kraus representation respectively with the error operators $\{E_j\}_{j\in\{1,\dots,m\}}$ and $\{F_k\}_{k\in\{1,\dots,n\}}$ and to ensure $m = n$ append zeros to the shorter list. Then we can state that $\mathcal{E} = \mathcal{F}$ if and only if exists a unitary matrix $\mathcal{U}$ made of $u_{jk} \in \mathbb{C}$ such that $E_j = \sum_k u_{jk} F_k$.*

Theorem D.0.2 (**Polar decomposition** [26])**.** *Consider $\mathcal{A}$ to be a linear operator acting on a vector space $\mathcal{H}$, then it is possible to decompose it in the product of a unitary operator $\mathcal{U}$ and one of the unique operators $\mathcal{R} = \sqrt{\mathcal{A}^\dagger \mathcal{A}}$ or $\mathcal{L} = \sqrt{\mathcal{A}\mathcal{A}^\dagger}$ such that:*

$$\mathcal{A} = \mathcal{U}\mathcal{R} = \mathcal{L}\mathcal{U}\,. \tag{D.3}$$

We can also state that if $\mathcal{A}$ is an invertible operator then the unitary operator $\mathcal{U}$ is unique.

Lemma D.0.3 ([26])**.** *Let $\mathcal{S} = \langle s_1, \dots, s_m \rangle$ be a subgroup of the Pauli group $\mathcal{G}_n$ generated by the set of independent elements $\{s_j\}_{j\in\{1,\dots,m\}}$ such that $-\mathbb{1} \notin \mathcal{S}$. Then $\forall i \in \{1, \dots, m\} \exists g \in \mathcal{G}_n$ such that:*

$$g s_i g^\dagger = -s_i, \quad \wedge \quad g s_j g^\dagger = s_j \quad \forall j \neq i \tag{D.4}$$

Theorem D.0.4 (**Code dimension of a stabilizer code** [26]s)**.** *Let $\mathcal{S} = \langle s_1, ..., s_{n-k} \rangle$ be a stabilizer group generated the the set of $n-k$ independent generators $\{s_j\}_{j\in\{1,...,n-k\}}$. The dimension of the code space $\mathcal{L}(\mathcal{S})$ is 2^k, hence we can represent k logical qubits.*

Proof. Let $\mathbf{x} \in \mathbb{F}_2^{n-k}$ and define an associated projector:

$$\mathcal{P}_{\mathcal{S}}^{\mathbf{x}} = \frac{1}{2^{n-k}} \prod_{j=1}^{n-k} (\mathbb{1} + (-1)^{x_j} s_j)$$

where x_j is the j-th element of $\mathbf{x}$ and it is immediate to observe that setting $\mathbf{x} = \mathbf{0}$ we obtain the projector on the code space defined by the $+1$ eigenspaces of the generators of $\mathcal{S}$. Applying now lemma D.0.3 we can choose proper elements $g_{\mathbf{x}} \in \mathcal{G}_n$ to construct the projector associated to a particular element of $\mathbb{F}_2^{n-k}$ with the following expression:

$$\mathcal{P}_{\mathcal{S}}^{\mathbf{x}} = g_{\mathbf{x}} \mathcal{P}_{\mathcal{S}}^{\mathbf{0}} g_{\mathbf{x}}^{\dagger}$$

hence the dimension of the vector space where the $\mathcal{P}_{\mathcal{S}}^{\mathbf{x}}$ project the state is the same $\forall \mathbf{x}$. We can now state that these projectors are always orthogonal with the following consideration. Consider the expression

$$\mathcal{P}_{\mathcal{S}}^{\mathbf{x}} \mathcal{P}_{\mathcal{S}}^{\mathbf{x}'} = \frac{1}{2^{2(n-k)}} \prod_{j=1}^{n-k} \prod_{\ell=1}^{n-k} (\mathbb{1} + (-1)^{x_j} s_j)(\mathbb{1} + (-1)^{x'_\ell} s_\ell),$$

setting $j = \ell$ we can observe that if $x_j \neq x'_j$ then the product is zero, hence the projectors are orthogonal. To finally end the proof of the theorem consider the product of the already defined 2^{n-k} projectors such that:

$$\sum_{\mathbf{x}\in\mathbb{F}_2^{n-k}} \mathcal{P}_{\mathcal{S}}^{\mathbf{x}} = \frac{1}{2^{n-k}} \sum_{\mathbf{x}\in\mathbb{F}_2^{n-k}} \prod_{j=1}^{n-k} (\mathbb{1} + (-1)^{x_j} s_j) = \mathbb{1}$$

hence we got 2^{n-k} projectors that project on a vector space with the same number of dimensions of the code space $\mathcal{L}(\mathcal{S})$ and that form a resolution of the identity, hence $\dim(\mathcal{L}(\mathcal{S})) = \frac{2^n}{2^{n-k}} = 2^k$. □

Appendix E

A more rigorous error model

Throughout this appendix we will expose a rigorous calculation starting from equation 4.9 for the case of identical qubits with identical coupling capacitances with the redonator, hence the dispersive shifts are identical as well. The result of this appendix will justify why we are doing this preliminary study on the simpler model highlighted in 3.35. The final error probabilities of the SPC associated to this rigorous model have the same form of the ones obtained starting 3.35 and reported in 3.49. The workflow we are going to use now is the same used in chapter 3.

Consider equation 4.9 obtained isolating the two-body part of equation 2.60 neglecting the ac-Stark shift

$$\dot{\rho} = \sum_{x,y \in \mathbb{F}_2^2} \Gamma_d^{xy} \, \Pi_x \rho \Pi_y \equiv \tilde{\mathcal{L}} \rho$$

where $\tilde{\mathcal{L}}$ will be referred to as pseudo-Lindbladian because it is a truncation of the complete Lindbladian operator. We can write the element of the PTM form of the pseudo-Lindbladian as

$$\tilde{\mathbf{L}}_{ij} = \frac{i^{\mathbf{x}_i^t \mathbf{z_i} + \mathbf{x}_j^t \mathbf{z_j}}}{4} \sum_{x,y \in \mathbb{F}_2^2} \Gamma_d^{xy} \, \mathrm{Tr}[X(\mathbf{x}_i) Z(\mathbf{z}_i) \Pi_x X(\mathbf{x}_j) Z(\mathbf{z}_j) \Pi_y] \tag{E.1}$$

From 2.64 $\Gamma_d^{xy} = (\chi_x - \chi_y) \Im[\alpha_x \alpha_y^*] = \Gamma_d^{yx}$, hence if $x = y$ then $\Gamma_d^{xx} = 0$. With this consideration we can recast equation E.1 into

$$\begin{aligned} \tilde{\mathbf{L}}_{ij} = \frac{i^{\mathbf{x}_i^t \mathbf{z_i} + \mathbf{x}_j^t \mathbf{z_j}}}{4} \sum_{x \in \{0,1,2\},\, y \in \{x+1,\ldots,3\}} \Gamma_d^{xy} \, \{ & \mathrm{Tr}[X(\mathbf{x}_i) Z(\mathbf{z}_i) \Pi_x X(\mathbf{x}_j) Z(\mathbf{z}_j) \Pi_y] + \\ & \mathrm{Tr}[X(\mathbf{x}_i) Z(\mathbf{z}_i) \Pi_y X(\mathbf{x}_j) Z(\mathbf{z}_j) \Pi_x] \} \end{aligned} \tag{E.2}$$

where we moved from indexing the states with elements of $\mathbb{F}_2^2$ to their associated binary value to simplify the notation. This last equation has been analysed via symbolic calculations via a Python sympy script and the whole pseudo-Lindbladian matrix came out as a block-diagonal matrix

$$\tilde{\mathbf{L}} = \mathrm{diag}\{\tilde{\mathbf{L}}_0, \tilde{\mathbf{L}}_1, \tilde{\mathbf{L}}_2, \tilde{\mathbf{L}}_3\} \tag{E.3}$$

where the blocks are 4×4 matrices given by

$$\tilde{\mathbf{L}}_0 = \mathbf{0}_{4\times 4},$$

$$\tilde{\mathbf{L}}_1 = \frac{1}{2}\begin{pmatrix} \Gamma_d^{0|1}+\Gamma_d^{2|3} & 0 & \Gamma_d^{0|1}-\Gamma_d^{2|3} & 0 \\ 0 & \Gamma_d^{0|1}+\Gamma_d^{2|3} & 0 & \Gamma_d^{0|1}-\Gamma_d^{2|3} \\ \Gamma_d^{0|1}-\Gamma_d^{2|3} & 0 & \Gamma_d^{0|1}+\Gamma_d^{2|3} & 0 \\ 0 & \Gamma_d^{0|1}-\Gamma_d^{2|3} & 0 & \Gamma_d^{0|1}+\Gamma_d^{2|3} \end{pmatrix},$$

$$\tilde{\mathbf{L}}_2 = \frac{1}{2}\begin{pmatrix} \Gamma_d^{0|2}+\Gamma_d^{1|3} & \Gamma_d^{0|2}-\Gamma_d^{1|3} & 0 & 0 \\ \Gamma_d^{0|2}-\Gamma_d^{1|3} & \Gamma_d^{0|2}+\Gamma_d^{1|3} & 0 & 0 \\ 0 & 0 & \Gamma_d^{0|2}+\Gamma_d^{1|3} & \Gamma_d^{0|2}-\Gamma_d^{1|3} \\ 0 & 0 & \Gamma_d^{0|2}-\Gamma_d^{1|3} & \Gamma_d^{0|2}+\Gamma_d^{1|3} \end{pmatrix}, \tag{E.4}$$

$$\tilde{\mathbf{L}}_3 = \frac{1}{2}\begin{pmatrix} \Gamma_d^{0|3}+\Gamma_d^{1|2} & 0 & 0 & -\Gamma_d^{0|3}+\Gamma_d^{1|2} \\ 0 & \Gamma_d^{0|3}+\Gamma_d^{1|2} & \Gamma_d^{0|3}-\Gamma_d^{1|2} & 0 \\ 0 & \Gamma_d^{0|3}-\Gamma_d^{1|2} & \Gamma_d^{0|3}+\Gamma_d^{1|2} & 0 \\ -\Gamma_d^{0|3}+\Gamma_d^{1|2} & 0 & 0 & \Gamma_d^{0|3}+\Gamma_d^{1|2} \end{pmatrix}$$

If we now focus on our case study we know that $\chi^1 = \chi^2 = \chi$, hence $\chi_0 = -\chi_3 = 2\chi$ and $\chi_1 = \chi_2 = 0$. This implies straightforwardly that $\Gamma_d^{1|2} = 0$ and it is possible to find some conditions on the other dephasing rates. To obtain these conditions we start from the following identities that can be deduced from the solution of the coherent state parameter equation obtained in 4.1

$$\begin{cases} \alpha_0 = \Re[\alpha_0] + i\Im[\alpha_0] \\ \alpha_1 = \Re[\alpha_1] + i\Im[\alpha_1] = i\Im[\alpha_1] \\ \alpha_2 = \Re[\alpha_2] + i\Im[\alpha_2] = i\Im[\alpha_1] \\ \alpha_3 = \Re[\alpha_3] + i\Im[\alpha_3] = -\Re[\alpha_0] + i\Im[\alpha_0] \end{cases} \tag{E.5}$$

Combining these equations with the already highlighted observations on the dephasing rates for our case study, we obtain

$$\begin{cases} \Gamma_d^{i|i} = 0 \quad \forall i \in \{0,1,2,3\} \\ \Gamma_d^{1|2} = 0 \\ \Gamma_d^{1|3} = \Gamma_d^{2|3} = \Gamma_d^{0|2} = \Gamma_d^{0|1} = -\chi_{00}\Re[\alpha_0]\Im[\alpha_1] = -2\chi\Re[\alpha_0]\Im[\alpha_1] \equiv \Gamma_d^{e|o} \\ \Gamma_d^{0|3} = -4\chi_{00}\Re[\alpha_0]\Im[\alpha_0] = -8\chi\Re[\alpha_0]\Im[\alpha_0] \end{cases} \tag{E.6}$$

This result is important for our purposes because it tells us that the dephasing rates between mixed elements of the even parity and odd parity subspace is the same and there is no dephasing in the odd parity subspace. The blocks of $\tilde{\mathbf{L}}$ become then

$$\tilde{\mathbf{L}}_0 = \mathbf{0}_{4\times 4},$$

$$\tilde{\mathbf{L}}_1 = \begin{pmatrix} \Gamma_d^{e|o} & 0 & 0 & 0 \\ 0 & \Gamma_d^{e|o} & 0 & 0 \\ 0 & 0 & \Gamma_d^{e|o} & 0 \\ 0 & & 0 & \Gamma_d^{e|o} \end{pmatrix},$$

$$\tilde{\mathbf{L}}_2 = \begin{pmatrix} \Gamma_d^{e|o} & 0 & 0 & 0 \\ 0 & \Gamma_d^{e|o} & 0 & 0 \\ 0 & 0 & \Gamma_d^{e|o} & 0 \\ 0 & & 0 & \Gamma_d^{e|o} \end{pmatrix}, \tag{E.7}$$

$$\tilde{\mathbf{L}}_3 = \frac{1}{2}\begin{pmatrix} \Gamma_d^{0|3} & 0 & 0 & -\Gamma_d^{0|3} \\ 0 & \Gamma_d^{0|3} & \Gamma_d^{0|3} & 0 \\ 0 & \Gamma_d^{0|3} & \Gamma_d^{0|3} & 0 \\ -\Gamma_d^{0|3} & 0 & 0 & \Gamma_d^{0|3} \end{pmatrix}$$

Solving now the differential equations with the same methods exposed in subsection 3.3.3 assuming constant dephasing rates, we obtain the block-diagonal channel matrix $e^{\tilde{\mathbf{L}}t} = \mathrm{diag}\{e^{\tilde{\mathbf{L}}_0 t}, e^{\tilde{\mathbf{L}}_1 t}, e^{\tilde{\mathbf{L}}_2 t}, e^{\tilde{\mathbf{L}}_3 t}\}$ where the blocks are constructed as follows

$$e^{\tilde{\mathbf{L}}_0 t} = \mathbb{1}_{4\times 4},$$

$$e^{\tilde{\mathbf{L}}_1 t} = \begin{pmatrix} e^{\Gamma_d^{e|o} t} & 0 & 0 & 0 \\ 0 & e^{\Gamma_d^{e|o} t} & 0 & 0 \\ 0 & 0 & e^{\Gamma_d^{e|o} t} & 0 \\ 0 & & 0 & e^{\Gamma_d^{e|o} t} \end{pmatrix},$$

$$e^{\tilde{\mathbf{L}}_2 t} = \begin{pmatrix} e^{\Gamma_d^{e|o} t} & 0 & 0 & 0 \\ 0 & e^{\Gamma_d^{e|o} t} & 0 & 0 \\ 0 & 0 & e^{\Gamma_d^{e|o} t} & 0 \\ 0 & & 0 & e^{\Gamma_d^{e|o} t} \end{pmatrix}, \tag{E.8}$$

$$e^{\tilde{\mathbf{L}}_3 t} = \begin{pmatrix} (1+e^{\Gamma_d^{0|3} t})/2 & 0 & 0 & (1-e^{\Gamma_d^{0|3} t})/2 \\ 0 & (1+e^{\Gamma_d^{0|3} t})/2 & -(1-e^{\Gamma_d^{0|3} t})/2 & 0 \\ 0 & -(1-e^{\Gamma_d^{0|3} t})/2 & (1+e^{\Gamma_d^{0|3t}})/2 & 0 \\ (1-e^{\Gamma_d^{0|3} t})/2 & 0 & 0 & (1+e^{\Gamma_d^{0|3} t})/2 \end{pmatrix}$$

At this point applying the Pauli twirling we get $e^{\tilde{\mathbf{L}}t}_{twirl} = \text{diag}\{e^{\tilde{\mathbf{L}}_0 t}_{twirl}, e^{\tilde{\mathbf{L}}_1 t}_{twirl}, e^{\tilde{\mathbf{L}}_2 t}_{twirl}, e^{\tilde{\mathbf{L}}_3 t}_{twirl}\}$ where the blocks are defined as

$$e^{\tilde{\mathbf{L}}_0 t}_{twirl} = \mathbb{1}_{4\times 4},$$

$$e^{\tilde{\mathbf{L}}_1 t}_{twirl} = \begin{pmatrix} e^{\Gamma_d^{e|o} t} & 0 & 0 & 0 \\ 0 & e^{\Gamma_d^{e|o} t} & 0 & 0 \\ 0 & 0 & e^{\Gamma_d^{e|o} t} & 0 \\ 0 & & 0 & e^{\Gamma_d^{e|o} t} \end{pmatrix},$$

$$e^{\tilde{\mathbf{L}}_2 t}_{twirl} = \begin{pmatrix} e^{\Gamma_d^{e|o} t} & 0 & 0 & 0 \\ 0 & e^{\Gamma_d^{e|o} t} & 0 & 0 \\ 0 & 0 & e^{\Gamma_d^{e|o} t} & 0 \\ 0 & & 0 & e^{\Gamma_d^{e|o} t} \end{pmatrix}, \tag{E.9}$$

$$e^{\tilde{\mathbf{L}}_3 t}_{twirl} = \begin{pmatrix} (1+e^{\Gamma_d^{0|3} t})/2 & 0 & 0 & 0 \\ 0 & (1+e^{\Gamma_d^{0|3} t})/2 & 0 & 0 \\ 0 & 0 & (1+e^{\Gamma_d^{0|3} t})/2 & 0 \\ 0 & 0 & 0 & (1+e^{\Gamma_d^{0|3} t})/2 \end{pmatrix}$$

Comparing now this channel matrix with the SPC PTM we obtain probabilities similar to equations 3.49 but with different exponential coeffients. The probabilities of the associated SPC then read:

$$\begin{cases} p(1) = \frac{1}{8}(1 - e^{\Gamma_d^{0|3} t}) \\ p(2) = \frac{1}{8}(1 - e^{\Gamma_d^{0|3} t}) \\ p(3) = \frac{1}{8}(3 + e^{\Gamma_d^{0|3} t} - 4e^{\Gamma_d^{e|o} t}) \\ p(i \in \{4, 5, ..., 14, 15\}) = 0 \end{cases} \tag{E.10}$$

As we can see the only difference between the probabilities E.10 and 3.49 is the second exponential in the $Z \otimes Z$ probability. This justifies why we decided to do this preliminary study with a simpler model defined by a single dephasing rate and a single jump operator $Z_e = |00\rangle\langle 00| - |11\rangle\langle 11|$ as reported in 4.13.

A last observation on this master equation is that, within the $\chi^1 = \chi^2$ hypothesis, we can write the master equation in an even parity subspace part that defines the block $\tilde{\mathbf{L}}_3$ which is contained in the first brackets and another part that contains the interactions between even parity states and odd parity states and defines the blocks $\tilde{\mathbf{L}}_1$ and $\tilde{\mathbf{L}}_2$ which is contained in the second brackets

$$\dot{\rho} = \Gamma_d^{03}(\Pi_0 \rho \Pi_3 + \Pi_3 \rho \Pi_0) + \Gamma_d^{e|o}\{(\Pi_0 + \Pi_3)\rho(\Pi_1 + \Pi_2) + (\Pi_1 + \Pi_2)\rho(\Pi_0 + \Pi_3)\} \tag{E.11}$$