Repetition Cat Qubits

Jérémie Guillaud

PSL
UNIVERSITÉ PARIS

THÈSE DE DOCTORAT
DE L'UNIVERSITÉ PSL

Préparée à l'École Normale Supérieure

Repetition Cat Qubits

Soutenue par
Jérémie Guillaud
Le 13 avril 2021

École doctorale n° 564
Physique en Île de France

Spécialité
Physique Quantique

Composition du jury :

Florian, MARQUARDT Scientific Director, Max Planck Institute	*Rapporteur*
William, OLIVER Professor, Massachussetts Institute of Technology	*Rapporteur*
Yvonne, GAO Assistant Professor, National University of Singapore	*Examinatrice*
Philippe, GRANGIER Professeur, Institut d'Optique	*Président du Jury*
Benjamin, HUARD Professeur, École Normale Supérieure de Lyon	*Examinateur*
Mazyar, MIRRAHIMI Directeur de recherche, Inria	*Directeur de thèse*

ENS | PSL

Inria

ÉCOLE NORMALE SUPÉRIEURE

DOCTORAL THESIS

Repetition Cat Qubits

Author:
Jérémie GUILLAUD

Supervisor:
Dr. Mazyar MIRRAHIMI

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

in the

QUANTIC Team
INRIA Paris

Abstract

The construction of a quantum computer is an extremely challenging task, because the states of the quantum system used to carry out the computation are typically far too fragile. A necessary condition to build such a computer is to design a system in which a large number of quantum bits are protected from the devastating effect of their environment to withstand the quantum information for a sufficiently long time. At the same time, performing a computation supposes the ability to control the quantum states to process the information they encode. The theory of quantum error correction opens the path towards the realization of macroscopically large quantum systems with, in theory, arbitrary good protection against the noise induced by the environment. The bottleneck of the implementation of quantum error correction is twofold. First, it requires to build quantum systems for which the levels of noise are below a constant value called the accuracy threshold. Second, the quantum error correcting code, and the processing of the encoded information, result in a large physical hardware overhead.

In this thesis, we propose and analyze a scheme based on repetition cat qubits to perform large scale quantum computation. The protection against the environment induced noise is achieved in two steps. First, the two-photon pumped cat qubits are arbitrarily well protected against bit-flip errors with the average number of photons in the cat state. Second, a repetition code protecting against phase-flips is implemented using cat qubits, thus producing a "repetition cat qubit" with very low logical error rate. In order to perform quantum computation with this protected qubit, we design a universal set of logical gates acting on the repetition cat qubit, that is compatible with the structure of the scheme. More precisely, the construction is achieved in two steps: first, the design of physical operations acting on cat qubits. These operations must be bias-preserving in order to preserve the natural protection against bit-flip errors. A particular attention has been devoted to proposing operations that could be experimentally realized in the next few years, within the framework of circuit quantum electrodynamics. Then, from the set of bias-preserving physical operations, we construct a universal set of logical operations on the repetition cat qubits.

We hope that the resulting scheme, and the ideas developed for its construction, will prove useful for the construction of a large-scale quantum computer.

Résumé

La construction d'un ordinateur quantique est un défi technologique extrêmement difficile à cause de la fragilité des états quantiques qui servent de support de calcul. La réalisation d'une telle machine nécessite de construire un grand nombre de systèmes quantiques suffisamment protégés du bruit inévitable induit par l'environnement, afin que la durée de vie de l'information quantique encodée dans ces systèmes soit suffisamment grande devant le temps d'exécution typique d'un algorithme quantique. Paradoxalement, l'implémentation d'algorithmes suppose aussi la capacité de manipuler l'information. La théorie de la correction d'erreur quantique établit qu'il est possible de construire des systèmes quantiques de taille macroscopique, et pourtant arbitrairement bien protégés contre le bruit induit par l'environnement. En pratique, deux obstacles majeurs s'opposent à la réalisation physique de la correction d'erreur quantique. Le premier obstacle à surmonter est de parvenir à construire un système quantique "de base" pour lequel le niveau du bruit est déjà suffisamment bas, un résultat connu sous le nom de "théorème du seuil". Le second obstacle concerne la taille de l'ordinateur quantique: en effet, aussi bien le code correcteur d'erreur que la capacité à manipuler l'information logique sont responsables d'un important surcoût en matériel.

L'objet de cette thèse est la construction et l'analyse d'un schéma particulier pour la réalisation d'un ordinateur quantique basé sur les qubits de chat répétés. La protection contre les erreurs est assurée en deux temps. D'abord, les qubits de chats utilisés sont arbitrairement bien protégés contre les erreurs dites de "bit-flip", lorsque le nombre moyen de photons dans les états de chat utilisés est suffisamment grand. Ensuite, un code de répétition contre les erreurs dites de "phase-flip" est construit à partir de ces qubits de chat. Le qubit logique résultant de cette construction est appelé le qubit de chat répété. Afin de réaliser du calcul quantique avec ce qubit, un ensemble universel de portes logiques pour les qubits de chat répétés est proposé dans cette thèse. La construction de ces portes logiques respecte la structure de la protection, afin de la préserver: les portes physiques construites au niveau des qubits de chat ont la propriété d'être "bias-preserving", c'est-à-dire qu'elles préservent la protection naturelle contre les erreurs de "bit-flip". Les propositions d'implémentation de ces portes ont été adaptées au maximum à la réalité des expériences contemporaines, afin que le schéma proposé puisse être raisonnablement implémenté d'ici quelques années. Enfin, à partir de ces portes physiques, un ensemble universel de portes logiques est construit au niveau du qubit de chat répété.

Nous espérons que le schéma ainsi proposé, et les idées développées pour sa construction, seront utiles dans la réalisation d'un ordinateur quantique.

Acknowledgements

I would like to express my deepest gratitude feelings to Mazyar Mirrahimi for the outstanding and consistent guidance he has provided throughout the years of this PhD work. His research intuitions, together with the patience and esteem he shows towards his students, lay out the most fertile ground to grow as a scientist. I feel blessed I was given the opportunity to work under his direction, and I look forward to further collaborating with him in the future.

Second, I would like to thank all the permanent members of the Quantic team for making this team such a rich scientific environment to work in. Most especially, I would like to thank Alain Sarlette for the invaluable help he has given me from the first to the last day of my PhD, and for keeping his door always open to students seeking guidance no matter how busy he may be. The Quantic team would not be what it is without Pierre Rouchon, whose sense of humour is matched only by his mathematical rigor. I would also like to thank Zaki Leghtas and Philippe Campagne-Ibarcq. No one really knows whether they are theorists disguised in experimentalists or the other way around, but having them around is a constant reminder that theoretical research will ultimately be applied in the lab and discussing with them always gives material to think about.

In the few years I spent within the team, I have had the chance to meet many outstanding persons. First, I would like to thank Rémi, Gerardo and Lucas who gave me a warm welcome in 2017 when I first joined the team, Paolo and Donatas for the nice work atmosphere, and Martine and Derya for constantly helping making my administrative life easier. I remember fondly the original TLS we shared at ENS with the Quantic team and would like to thank Benjamin, Quentin, Raphaël, Théau, Danijela, Sébastien and the rest of the Benjamin Huard's group.

Then, I would like to thank the dream team François-Marie, Ronan, Lev-Arcady, Michiel, Vincent and Christian for the exceptional work environment they have turned the team into. It has been a privilege to share so many good moments with you all, and I will miss a lot our science discussions, our famous FLCs, our (clandestine) sherry chouffes, the countless running sessions, stealing tea from you but also finding my own teabox empty after spending a few weeks away, etc. I wish you all the best of luck with your PhDs and look forward to keep working with some of you.

Finally, I would like to give my deepest thanks to my friends and family for their constant support throughout my entire student life and most especially during these last years. Having all of you around definitely helped to get through the difficult moments of a PhD journey.

Contents

Chapter 1

Introduction

1.1 Promise of quantum computing

The imperious need to *understand* is deeply rooted in human nature. From the child overwhelming his parents with "why?"s about the world surrounding him to the most educated philosopher or scientist investigating the origin of life, the need to *find answers* inhabits each and everyone of us. Science is perhaps the most extraordinary collective attempt at satisfying this need. Through careful observation of Nature, patterns are identified and hypothesis formulated, eventually leading to new theories. A theory that can accurately predict what is actually *observed* is considered a valid answer and is accepted as the truth, growing our collective knowledge about the world. The validity of a theory thus crucially depends on our ability to observe Nature and its effects, and what is believed to be true can become wrong upon a closer inspection.

In physics, the observation of Nature naturally began at the time, space and energy scales available to us. By studying the relationship between a body and the forces acting upon it, Isaac Newton established the three Laws of Motion in his *Philosophiæ Naturalis Principia Mathematica* first published in 1687, laying the foundations for classical mechanics. The continuous development of scientific equipment

allowed scientists to observe Nature ever more precisely, over space and time scales far beyond what is conceivable for a human mind, which eventually led to observations that could not be explained by classical physics. The accurate observation of black-body radiation spectra could not be predicted by any existing theory, motivating Max Planck to formulate in 1900 the hypothesis that an electrically charged oscillator in a cavity containing a black-body could only absorb discrete amounts of energy proportional to the frequency of its associated electromagnetic wave. The observation of the photoelectric effect by Heinrich Hertz in 1887 was incompatible with the hypothesis in classical electromagnetic theory that a beam of light is a continuous wave propagating in space. In 1905, Albert Einstein explained this effect (and other light-related phenomena including the black-body radiation) by suggesting that a beam of light is a collection of discrete wave packets that he called the *light quantum* (das Lichtquant), now called *photons*, whose energy is proportional to the light frequency $E = \hbar\nu$. These two important observations and the ideas developed as an attempt to explain them were foundational in the development of quantum mechanics during the first half of the twentieth century. Quantum mechanics is the only theory that can predict accurately the behaviour of physical systems at the atomic scale, yet it also successfully explains all the predictions of classical physics theories in the limit where the size of the system becomes large. In this limit, called the *classical* or *correspondence* limit, some features specific to quantum mechanics "disappear", such as the wave-particle duality, the wave function collapse, quantum entanglement, or the principle of uncertainty.

Up to this day, there has been no observation of Nature that quantum mechanics failed to predict. Yet, even though the theory is successful in predicting what we are able to observe, the interpretation to give to the mathematical formalism of quantum mechanics still remains a challenge for physicists. Because we only experience the effects of quantum mechanics at the macroscopic scale, where some of its features do not longer exist, it is very challenging for a human mind to have an intuition of purely quantum mechanical effects.

In the second half of the twentieth century, the development of modern computers opened a new era for science. The physical laws governing the evolution of a given system are at the core of the *model of computation* used by the programmers to develop large computational systems. Physicists and computer scientists naturally began to wonder whether the theory of quantum mechanics could produce a novel model of computation based on quantum systems. In 1980, Paul Benioff developed a quantum mechanical model of Turing machines, establishing for the first time that reversible quantum computing was possible, at least in theory. Building on this work, Richard Feynman, David Deutsch and other physicists suggested in the 1980s that quantum computation had the potential to achieve things classical computation could not. More specifically, quantum computation does not provide any additional advantage in terms of computability, in the sense that a quantum computer can solve any computational problem that a classical computer can, and

reciprocally. Rather, Feynman suggested that using a quantum model of computation would allow to solve certain problems *faster* than what is possible with any classical model of computation. This intuition, initially based on the observation that the simulation of a quantum mechanical system is computationally challenging for a classical computer, but is natural for a quantum mechanical computer, culminated in 1994 with the publication by Peter Shor of a quantum factoring algorithm with an exponential speedup over the best known classical factoring algorithm. Other "quantum algorithms", i.e algorithms that can only be run on a quantum computer, exhibiting a speedup over their best known classical counterparts include Grover's algorithm to conduct searches in large database or the Harrow, Hassidim and Lloyd (HHL) algorithm to solve linear systems of equations.

1.2 Circuit model of quantum computation

In the circuit model of quantum computation, the basic unit of quantum information is the *quantum bit*, or *qubit*. From a practical point of view, any quantum system that can exist in two different states can be used to store a quantum bit of information. It could be stored, for instance, in the states 'up' and 'down' of the spin of an electron, the 'vertical' and 'horizontal' polarization of a photon, the 'ground' and 'excited' states of an atom, etc. In this dissertation as well as in the quantum information literature, the term *qubit* refers to both the quantum bit of information and the physical two-level quantum system in which it is encoded. The actual physical system used as the recipient of the quantum bit of information can be abstracted away by simply denoting its two states $|0\rangle$ and $|1\rangle$, called the *computational states*, where the notation 0 and 1 facilitates the classical analogy and the quantum mechanical Dirac notation $|\cdot\rangle$ emphasizes the quantum behaviour of the system.

Mathematically speaking, the two states $|0\rangle$ and $|1\rangle$ are orthogonal unit vectors forming the computational basis of a complex separable Hilbert space, called the *state space* of the system. By the rules of quantum mechanics, any unit vector $|\psi\rangle$, called the *state vector*, of this associated Hilbert space

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

where α and β are complex numbers satisfying $|\alpha|^2 + |\beta|^2 = 1$, is a valid state of the system. Two state vectors that differ only by a global phase, $|\psi\rangle$ and $|\tilde{\psi}\rangle = e^{i\phi}|\psi\rangle$ actually describe the same physical state, and the complex number α can be assumed to be real without loss of generality. Writing $\alpha = \cos(\frac{\theta}{2})$ and $\beta = e^{i\varphi}\sin(\frac{\theta}{2})$, the vector states of a qubit can be conveniently visualized as the points of the surface of the Bloch sphere, as depicted in Figure 1.1, where the state is parametrized by the angles $(\theta, \varphi) \in [0, \pi] \times [0, 2\pi[$.

In order to learn the state of a qubit, one needs to measure it. A quantum measurement is fundamentally different than its classical counterpart. A classical bit of

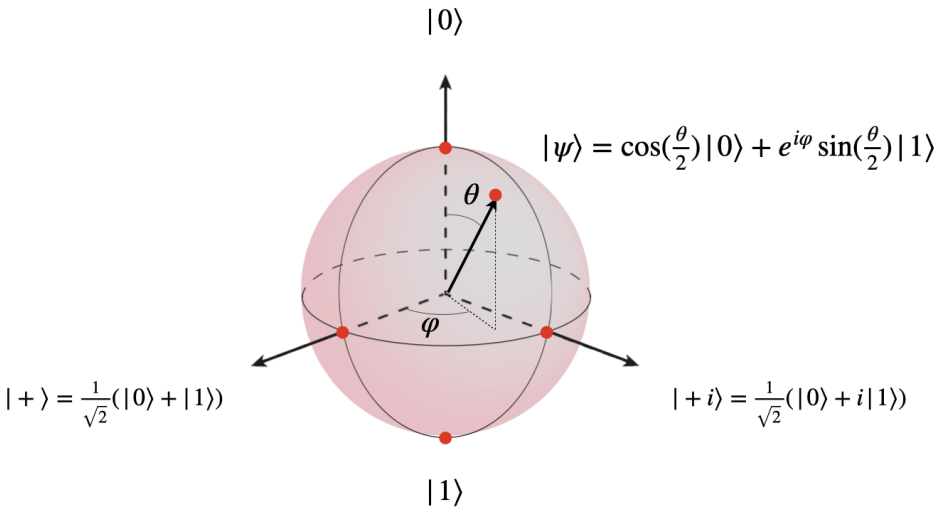


FIGURE 1.1: Bloch sphere representation of the state of a qubit.

information is either in the state '0' or the state '1' and a perfect measurement of such a bit reveals the state of the system with unit probability. Since the most general state of a quantum bit is $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, one could optimistically assume that a perfect quantum measurement of the qubit would reveal the complex number α and β. Unfortunately, in quantum mechanics, one cannot directly measure the state of the qubit but rather, the value of physical *observables* corresponding to Hermitian operators acting on the state space. When the qubit is in the state $|\psi\rangle$, the measurement of the operator associated with the canonical states $|0\rangle$ and $|1\rangle$ produces a non-deterministic outcome: the measurement outcome is +1 (resp. -1), corresponding to the state $|0\rangle$ (resp. $|1\rangle$), is obtained with probability $|\alpha|^2$ (resp. $|\beta|^2$). Importantly, such a measurement has a *back-action* on the quantum state of the system. If, for example, the result of the measurement is +1, then the state of the qubit *after* the measurement is no longer $|\psi\rangle$ but $|0\rangle$, such that a second measurement just after the first one would now yield the same result with probability one. This quantum phenomenon is known as the *collapse* of the wave function, as the quantum superposition is destroyed and the information contained in the complex numbers α and β is lost.

The *pure* states of a qubit, described by state vectors, correspond to the states of *maximal knowledge* of the state of a qubit perfectly isolated from any other quantum system. However, they fail to describe the state of a qubit which shares some information with another quantum system. To illustrate this, consider the case of a composite quantum system made of two qubits. As in the classical case, there are

four computational states $|00\rangle, |01\rangle, |10\rangle$ and $|11\rangle$, forming the basis of a four dimensional Hilbert space. The quantum state

$$|\psi\rangle = \tfrac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

is said to be *entangled*, because it cannot be written as a separable product of two distinct quantum states

$$|\psi\rangle = |\psi_{\text{qubit 1}}\rangle \otimes |\psi_{\text{qubit 2}}\rangle.$$

For a qubit entangled to another quantum system, the full description of the quantum state of the whole system as a pure state necessarily involves both quantum systems. In this case, it is still possible to consider the state of the qubit only, disregarding the other quantum systems to which it is entangled, and the state is said to be a *mixed*. Mixed states are described by a *density matrix* denoted ρ, which is positive operator of trace 1 acting on the state space of the system. For a pure state $|\psi\rangle$, the density matrix reduces to the projector on this state $\rho = |\psi\rangle\langle\psi|$.

The state vector of a qubit can be modified by the application of a quantum operator acting on the state space. Quantum operators are linear operators and because they map unitary vectors to unitary vectors, they are necessarily unitary operators. In the context of quantum information, the usual basis of U(2) are the four 2×2 complex *Pauli matrices*

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

In the quantum circuit model of computation, a computation is carried out by a quantum circuit composed of quantum operators, called the *quantum gates*, acting on a register of n quantum bits. Any unitary operator acting on the state of n qubits can be decomposed on the n-qubit Pauli group, obtained by forming all the possible tensor products of the Pauli matrices

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

and the four overall phases $\{\pm 1, \pm i\}$ ensure the group structure.

Postponing the measurement of the state of the quantum register of qubits to the end of the execution of a quantum algorithm, a quantum algorithm performed on a register of n qubits is a unitary operation $U \in \mathrm{U}(2^n)$. There are infinitely many such operations, so that is it not reasonable to expect to be able to implement any of these unitaries physically on the register. Two key theoretical concepts are crucial for the realization of a quantum computer.

First, an arbitrary quantum operator can be arbitrarily well approximated by a *finite* set of operators. A finite set of operators $\mathcal{S}$ that can actually approximate *any*

unitary operator is said to be *universal* for quantum computation: mathematically speaking, this is the case if the group generated by $\mathcal{S}$ is dense in U(2^n). Actually, since the overall global phase has no physical meaning, it is enough for the group generated by $\mathcal{S}$ to be dense in SU(2^n). From a practical point of view, this result is essential: it implies that for a given platform used to encode quantum information, a discrete set of physical operations only needs to be built to unlock the ability to perform any quantum algorithm.

Second, a universal set of quantum operators needs to "fill" SU(2^n) quickly enough. Indeed, one could imagine a pathological situation where a quantum algorithm implemented by a unitary $U \in$ SU(2^n) provides an exponential speed-up in the number of input bits n with respect to the best known classical counterpart, but where the approximation of U would require an exponential number of gates in the universal gate set naturally available to the quantum hardware, thus annihilating the theoretical advantage. Fortunately, the Solovay-Kitaev theorem ensures that any unitary can be arbitrarily well approximated with a logarithmic number of gates from a universal gate set:

Theorem 1 (Solovay-Kitaev) *Let $\mathcal{S}$ be a finite set of elements in SU(2^n) containing its own inverses, such that $\langle \mathcal{S} \rangle$ is dense in SU(2^n). Let $\epsilon > 0$. Then there is a constant c such that for any $U \in$ SU(2^n), there is a sequence of gates S from $\mathcal{S}$ of length $\mathcal{O}(\log^c(1/\epsilon)$ such that $\|S - U\| < \epsilon$.*

The constant c depends on the universal gate set but is typically a small number, $c \approx 4$, and the existence of gate sets achieving the minimal value $c = 1$ was established in [63]. Well known universal gate sets include

- CNOT and all single qubit gates
- Clifford group and any single non-Clifford gate
- CNOT, Hadamard and T gate
- Toffoli and Hadamard.

The discovery of quantum algorithms and their potential applications motivates the construction of the quantum computer. Yet, forty years after the field of quantum computation was initiated, building such a machine remains a formidable engineering challenge. The most difficult obstacle to overcome is the *quantum decoherence*, the loss of coherence between quantum states. The coherence between two quantum states has to do with the *superposition* principle, a purely quantum mechanical feature that is crucial for quantum algorithms: it states that if a quantum system, say the famous Schrödinger quantum cat, can exist in two states, $|\text{dead}\rangle$ or $|\text{alive}\rangle$, then it can also exit in a coherent superposition of these two states, $\frac{1}{\sqrt{2}}(|\text{dead}\rangle + |\text{alive}\rangle)$, where the quantum coherence refers to the '+' sign in the superposition. If the quantum cat was a perfectly isolated quantum system, it could remain indefinitely in this

quantum state. Unfortunately, the construction of a perfectly isolated quantum system in a realistic experimental set-up is impossible. The effect of the interaction of the cat with its environment is to *decohere* the state, that is to destroy the coherence between the $|\text{dead}\rangle$ and the $|\text{alive}\rangle$ states. Somehow, the quantum decoherence can be thought of as a loss of information into the environment of the quantum system. Even though this quantum decoherence is inevitable in any real-world experiment, the better the quantum system is isolated, the slower the effect of quantum decoherence. However, in order to reach the extremely low levels of noise required to perform useful quantum computations, improving the isolation of physical quantum systems is not enough and some additional layer of active correction of the remaining errors is required. The next section provides an introduction to the theory of error correction, with a particular focus on the theory of quantum error correction with bosonic qubits used in this work.

1.3 Theory of error correction

1.3.1 The spirit

The theory of error correction enables the detection and correction of errors occurring on the physical system used to store information. In this sense, an "error" is simply an undesired change of the state of the system, that is caused by some physical process which cannot be controlled or cancelled *a priori*. Interestingly, the exact nature of the event that resulted in a change of the system's state does not need to be known for the error to be corrected; rather, the only thing that really matters is the *effect* of this process on the state of the system. Consider, for instance, the punched card used at the beginning of the twentieth century to store data. A punched card is a stiff piece of paper on which holes can be punched. Each location on the card is a system which can be in two states, either punched or not, thus storing a binary digit ('punched' to store a '1' and 'not punched' to store a '0'). There are infinitely many physical reasons that a bit might be corrupted (a punching process could fail to properly make a hole, resulting in a '0' being stored instead of a one, or a mice might eat a piece of the card and make a hole, corrupting a stored '0' into a '1', etc.), but, assuming the card survives the undesired event, each and every one of these uncontrolled events can only ever result in an error called the *bit-flip*, that is an encoded '0' turned into a '1' and vice-versa. The probability that such an error occurs, whatever the cause, is referred to as the *physical error probability* p, as it is set by the probability that some undesired physical event occurred and corrupted the state of the system.

In 1947, the American mathematician Richard Hamming, then working at Bell Telephone Laboratories, started to work on the problem of error correction. Frustrated by the failure of calculating machines, he set himself the goal to design active protocols with the ability to detect a bit-flip error event, and most importantly, to

identify without ambiguity which bit had been corrupted, such that the error could be corrected without losing the information stored in the system. This is to be contrasted with the *error detecting* protocols that already existed at this time: such a protocol is capable of telling whether the information has been corrupted, but is unable to remove the error that happened. Rather, in this case, the information is simply lost. In 1950, Richard Hamming was successful in his quest and proposed the first error correcting code, known as the Hamming(7,4) code.

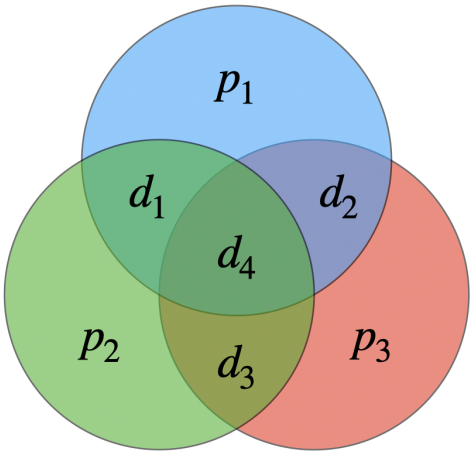


FIGURE 1.2: Parity checks of the Hamming(7,4) Code.

This protocol proposes to encode four bits of information, called the *logical bits*, using seven physical bits, in such a way that a single bit-flip error occurring on any of the seven physical bits can be detected and corrected without damaging any of the four encoded logical bits. This code is said to be a first order error correcting code, because at least two errors must happen between two rounds of error correction for the protocol to fail to remove errors. If the errors occur independently, with the same physical error probability p, then overall *logical error probability* p_L that any of the four encoded logical bits of information is corrupted scales as p^2, which is much smaller than p when p is small. Although this pioneer code is now 70 years old, it is a good illustration of the working principles of error correction: increasing the number of "noisy" physical systems used to encode physical bits of information provides an increased protection for a smaller number of logical bits of information. However, this stronger protection comes at the price of adding complexity to the actual physical system, the hardware, used to store the information. This trade-off between the number of physical systems used to encode a fixed number of bits of information and the resulting logical error probability is at the core of the theory of error correction.

Further insights can be gained by inspecting how the protocol works in practice. In addition to the four bits d_1, d_2, d_3, d_4, the data bits, one wishes to protect, three additional bits p_1, p_2, p_3, called the parity bits, are added. The *encoding* of the four

logical bits into the seven physical bits is performed by setting the value of the three parity bits to

$$p_1 = d_1 \oplus d_2 \oplus d_4$$
$$p_2 = d_1 \oplus d_3 \oplus d_4$$
$$p_3 = d_2 \oplus d_3 \oplus d_4$$

where $\oplus$ is the addition modulo 2 (the eXclusive OR operation in terms of Boolean logic). Once the four data bits are encoded in the seven-bit string $d_1d_2d_3d_4p_1p_2p_3$, they are protected against any single bit-flip acting on any of the seven bits, and ready to undergo some error. The *decoding* of the four logical bits works as follows. The first step is to determine if an error has occurred, and when it has, to identify it, by forming the *error syndrome* $s_1s_2s_3$ composed of the three following bits

$$s_1 = p_1 \oplus d_1 \oplus d_2 \oplus d_4$$
$$s_2 = p_2 \oplus d_1 \oplus d_3 \oplus d_4$$
$$s_3 = p_3 \oplus d_2 \oplus d_3 \oplus d_4.$$

By construction, one can check that when none of the seven bits suffered from a bit-flip, the error syndrome is trivial $s_1s_2s_3 = 000$. The code is successful in correcting any single bit-flip error because each of the remaining $7 = 2^3 - 1$ possible values of the error syndrome points unambiguously to one of the seven possible single bit-flip errors. This can be easily checked on the Venn diagram of the code depicted in Figure 1.2 as follows: when either one of the physical bit flips, say d_1, the syndrome bits s_1 (parity of the bits contained in the blue circle) and s_2 (green circle) flip, resulting in the error syndrome $s_1s_2s_3 = 110$. The error being unambiguously identified, it can be corrected, and the four logical bits retrieved trivially by reading out the values of the data bits d_1, d_2, d_3 and d_4.

The one-to-one correspondence between the possible errors that the code is designed to correct and the possible error syndromes is a fundamental feature of the theory of error correction. The Hamming(7,4) code fails to correct two bit-flips errors because there are different combinations of two bit-flip errors that result in the same error syndrome; for instance, the error syndrome 110 obtained when the d_1 bit flipped could also be caused by a combination of two bit-flips on the parity bits p_1 and p_2. However, if one renounces the ability to correct for single bit errors, then the code can successfully *detect* when two errors have occurred, because there is no combination of two errors that can produce the "no error" syndrome $s_1s_2s_3 = 000$. The Hamming(7,4) code is said to be a single-error correcting code or a two-error detecting code.

The Hamming(7,4) is the ancestor of a more general class of linear error-correcting code called the Hamming codes. More precisely, it can be extended to encode $k = 2^r - r - 1$ logical bits protected against any single bit-flip error using

$n = 2^r - 1$ physical bits for any integer $r \geq 2$.

1.3.2 Quantum error correction

The theory of error correction proves that the error probability afflicting the information encoded in a given physical system could be effectively suppressed by cleverly encoding the information in a larger physical system. An important theoretical question is whether this theory can be extended to quantum systems, in order to actively suppress the quantum decoherence detrimental to building large-scale quantum computers. Some features of quantum mechanics, however, need to be addressed carefully when adapting the theory of classical error correction to quantum systems.

The first of these features, known as the *no-cloning theorem*, is that quantum information cannot be duplicated: there is no quantum operation $\mathcal{E}$ that can produce the mapping

$$\mathcal{E}[|\psi\rangle \otimes |\phi\rangle] = |\psi\rangle \otimes |\psi\rangle$$

for an arbitrary state $|\psi\rangle$. This is actually a consequence of the linearity of quantum mechanics: if such an operation existed, then cloning the superposition of two arbitrary orthogonal states $|\psi_1\rangle$, $|\psi_2\rangle$ would produce the state

$$\begin{aligned}
\mathcal{E}[\tfrac{1}{\sqrt{2}}(|\psi_1\rangle + |\psi_2\rangle) \otimes |\phi\rangle] &= \tfrac{1}{\sqrt{2}}(\mathcal{E}[|\psi_1\rangle \otimes |\phi\rangle] + \mathcal{E}[|\psi_2\rangle \otimes |\phi\rangle]) \\
&= \tfrac{1}{\sqrt{2}}(|\psi_1\rangle \otimes |\psi_1\rangle + |\psi_2\rangle \otimes |\psi_2\rangle) \\
&\neq \tfrac{1}{\sqrt{2}}(|\psi_1\rangle + |\psi_2\rangle) \otimes \tfrac{1}{\sqrt{2}}(|\psi_1\rangle + |\psi_2\rangle).
\end{aligned}$$

Thus, the design of a quantum error correcting code to protect quantum information against decoherence cannot rely on some part of the quantum information being duplicated.

The second feature, perhaps more subtle, is the back-action of a quantum measurement on the system being measured. As illustrated by the example of the Hamming(7,4) code, a crucial step when implementing a quantum error correcting is the ability to measure error syndromes that reveal which errors occurred on the system. This step requires extra care in the quantum case, as we have seen that the direct measurement of a quantum state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ in the computational basis $\{|0\rangle, |1\rangle\}$ actually destroys the information encoded in the complex numbers α and β.

Fortunately, none of these features seem to pose a fundamental problem in designing efficient quantum error correcting code. After all, the Hamming(7,4) code requires neither cloning the data bits $d_1 d_2 d_3 d_4$ nor directly measuring any of these bits. Indeed, extra bits were added to probe the collective *parity* of the bits rather than the value of the bit themselves, a principle that can be successfully translated to the quantum world.

The last problem one faces when building a theory of quantum error correction is the numbers of errors to deal with. Indeed, we argued that in the classical case, any undesired physical process acting on a physical bit could only ever result in a bit-flip error on the stored bit of data. Consequently, the error correcting code can focus on detecting the effect of the physical process, the bit-flip, and on correcting it. In the quantum case, a unitary physical process acting on the state of an isolated qubit $|\psi\rangle$ can produce the state $|\tilde{\psi}\rangle = U_{\text{err}}|\psi\rangle$, where $U_{\text{err}} \in \text{U}(2)$ is the resulting error on the encoded bit of information and can be any unitary operator! Even more generally, the errors afflicting the qubit usually arise from undesired interactions with its surrounding environment, entangling the state of the qubit with uncontrolled degrees of freedom. In this case, the state $|\psi\rangle$ of the qubit becomes mixed and the error channel is described by a completely positive, trace preserving *Kraus map* $\mathcal{E}$

$$\rho = |\psi\rangle\langle\psi| \underset{\text{error}}{\rightarrow} \rho' = \mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger$$

where the *Kraus operators* E_k, acting on the state space, satisfy $\sum_k E_k^\dagger E_k = 1$. The task may seem overwhelming: how can one design a code that unambiguously detect and correct any single error, when there are now uncountable infinitely such errors? In this case, the linearity of quantum mechanics and the quantum measurement back-action actually play on our side as they give rise to the following discretization of error channels theorem:

Theorem 2 (Discretization of error channels) *Let $\mathcal{R}$ be a recovery operation for the error channel $\mathcal{E}$ described by the Kraus operators $\{E_k\}$, i.e for all $\rho \in \mathcal{C}_{\text{code space}}$*

$$\mathcal{R} \circ \mathcal{E}(\rho) = \rho.$$

where $\mathcal{C}_{\text{code space}}$ is the subspace of the state space in which logical information is stored. Let $\mathcal{F}$ be another error channel described by the Kraus operators $\{F_k\}$, such that every F_k can be expressed as a linear combination of the $\{E_k\}$.

Then, the recovery map $\mathcal{R}$ is a recovery operation of the error channel $\mathcal{F}$.

Because of the linearity of quantum mechanics, any Kraus operator E_k is a linear combination of some fixed basis operators, say the Pauli operators. Thus, the task of designing a quantum error correcting code and the associated recovery operation reduces to designing a code that can correct for every element of the Pauli basis. Actually, because $XZ = -iY$, a code that can correct any two errors in the set $\{X, Z\}$ errors can also correct a single Y error.

A consequence of this theorem is that, without loss of generality on the quantum channel describing the noise acting on the quantum bit, one can merely focus on the ability for a quantum error correcting codes to correct for Pauli X, Y and Z errors.

The X error, which is an uncontrolled swap of the computational states

$$\begin{aligned} X : |0\rangle &\rightarrow |1\rangle \\ |1\rangle &\rightarrow |0\rangle \end{aligned}$$

is called the *bit-flip* error, to emphasize the classical analogy, while the Z error is referred to as a *phase-flip* error, which is a "flip" of the sign of the relative phase (the "coherence") between the two computational states

$$\begin{aligned} Z : \tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle) &\rightarrow \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \\ \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle) &\rightarrow \tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \end{aligned}$$

and the Y error is a combination of both (a "bit and phase flip").

Let us now introduce one of the simplest quantum error correcting code, the 9-qubit Shor code, not so much for its pedagogical simplicity as because it is the rightful ancestor of the logical construction proposed in this manuscript. The 9-qubit Shor code encodes a single logical quantum bit of information $|\psi\rangle_L = \alpha|0\rangle_L + \beta|1\rangle_L$, where the L subscript emphasizes that this is a protected bit, using nine physical qubits. In the 512-dimensional Hilbert space of the full nine qubits system, the two computational states chosen as a basis are

$$\begin{aligned} |0\rangle_L &= \tfrac{1}{2\sqrt{2}}(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)(|000\rangle + |111\rangle), \\ |1\rangle_L &= \tfrac{1}{2\sqrt{2}}(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)(|000\rangle - |111\rangle). \end{aligned} \tag{1.1}$$

The two-dimensional subspace of the whole state space generated by the *code words* $|0\rangle_L$ and $|1\rangle_L$ is called the *code space*. The detection of a single qubit X error on any of the nine qubits is made by measuring the six operators Z_1Z_2, Z_2Z_3, Z_4Z_5, Z_5Z_6, Z_7Z_8 and Z_8Z_9, where the subscripts refer to the qubit on which the Z operator is measured. Importantly, the measurement of these operators does not disturb the encoded information: in the absence of errors, each of the six measurement outcomes will be +1 with unit probability and the state of the system after the measurements will remain unchanged.

Suppose now that one of the qubits, say the first one, has been flipped. Since the error X_1 anti-commutes with Z_1, the measurement outcome of the Z_1Z_2 measurement will be -1, while the other five measurement outcome will be +1. Along the same lines, one can check that any single qubit X error produces a pattern of measurement outcome that allows to determine unambiguously which qubit flipped.

Similarly, any single phase-flip error Z can be detected by measurement the two operators $X_1X_2X_3X_4X_5X_6$ and $X_4X_5X_6X_7X_8X_9$. One could legitimately object that in this case, the measurement of these two operators can only produce four different measurement records: $(+1,+1)$, $(+1,-1)$, $(-1,+1)$ and $(-1,-1)$, which does not seem to be enough to label the 10 distinct scenarios (no error at all, or a single Z error

on any of the nine qubits). Yet, one can observe that for the particular choice of the computational states $|0\rangle_L$ and $|1\rangle_L$, the Z errors work in triplets, that is, a Z error on any of the first three qubit (say) produces the same effect

$$Z_i(|000\rangle \pm |111\rangle) \rightarrow (|000\rangle \mp |111\rangle),\ i = 1, 2, 3.$$

In order to correct a Z error on any of the nine qubits, it suffices to learn in which of the three triplets (1,2,3), (3,4,5) or (7,8,9) it has occurred, for which the four measurement outcomes are enough. Here, we checked 'by hand' that any error in the set $\{X_i, Z_i\}_i$ was correctable by the 9-qubit Shor code. The intuition was that any two different errors should result in two different measurement outcome patterns. This general statement is summarized in the Knill-Laflamme condition for error correction:

Theorem 3 (Knill-Laflamme condition for error correction) *Let $\mathcal{C}_{code\ space}$ be the subspace of the state space encoding the logical information, P_c the projector onto $\mathcal{C}_{code\ space}$ and $\mathcal{E}$ an error channel described by the set of operators $\{E_k\}$. A recovery operation $\mathcal{R}$ correcting $\mathcal{E}$ on $\mathcal{C}_{code\ space}$ exists if, and only if*

$$P_c E_i^\dagger E_j P_c = \alpha_{ij} P_c$$

where $\alpha = (\alpha)_{ij}$ is a Hermitian matrix of complex numbers.

So far, we have only considered the use of error correction to protect a bit of information against errors in time, in order to increase its lifetime. However, building a computing machine requires not only the ability to store the data reliably but most importantly the ability to *process* it, that is, to perform operations on it. At first glance, processing logical bits of information stored across a large number of systems could be done by first decoding the bit, bringing back the information into a single, small physical system, applying the required operations, and finally re-encoding the processed data into the error correcting code to store it robustly until it needs to be processed again. This strategy is effective if the decoding, processing, and re-encoding steps can be performed sufficiently fast and reliably. This is usually not the case, and the errors affecting the information while it is not protected will be detrimental to large quantum computations. Fortunately, the theory of quantum error correction establishes that the logical qubits can be processed without being decoded first. Instead, *logical operations* are designed that act directly on the logical bits without ever exposing the logical information to noise.

Consider the Pauli operator X, which exchanges the two computational states $|0\rangle$ and $|1\rangle$. For the 9-qubit Shor code, a logical Pauli X_L operator is an encoded version of the X operator that swaps the two logical states $|0\rangle_L$ and $|1\rangle_L$ in equation (1.1), while a logical Z_L operator should swap the two opposite phase superposition states $\frac{1}{\sqrt{2}}(|0\rangle_L \pm |1\rangle_L)$. One can check by looking at the logical computational states that a logical X_L operation (resp. Z_L) is realized by applying Pauli Z operator (resp. Pauli

X) to all the nine physical qubits

$$X_L = Z_1 Z_2 Z_3 Z_4 Z_5 Z_6 Z_7 Z_8 Z_9$$
$$Z_L = X_1 X_2 X_3 X_4 X_5 X_6 X_7 X_8 X_9.$$

A "legitimate" logical operator is any operator that acts on the code space as the unencoded version would, but it does not matter how this operator acts on the rest of the state space. Because of this gauge degree of freedom, the choice of the logical operators is not unique, and another possible choice for the logical operators of the 9-qubit Shor code is

$$X_L = Z_1 Z_4 Z_7$$
$$Z_L = X_1 X_2 X_3.$$

The minimal number of physical qubits which need to be acted upon to implement a non-trivial logical operator is called the *code distance*. The distance of the 9-qubit Shor code is three, so the second choice of X_L and Z_L is actually the most economic one. A code that encodes k logical bits of information into n physical bits, of distance d, is denoted $[n, k, d]$ code. Such a code can, at least in theory, *detect* up to $d - 1$ errors but no more. Indeed, suppose that d random errors occur and unluckily exactly form a logical operator. Then, the effect of these errors leaves the code space globally invariant and thus cannot be detected. If, additionally, we ask that the code be able to not only detect but *correct* for errors, then a $[n, k, d]$ can only correct up to $t = (d - 1)/2$ errors, that is to say, the errors that can be corrected are those that have not been able to close *half* the distance between two valid code words.

An important remark is in order. Similarly to the classical $[7, 4, 3]$ Hamming(7,4) code, the $[9, 1, 3]$ 9-qubit Shor code is a single-error correcting code, also called a *first-order* error correcting code, in that it can correct any single qubit error occurring on any of the 9 qubits of the code. Assuming that the errors on the physical bits or qubits are independent and happen with equal physical error probability p, this code produces a logical bit of information for which the error rate scales as $p_L \propto p^2$, since the combination of at least two (independent) errors need to happen for the logical bit to be corrupted. In the classical case, given the already extremely low physical error probability p that can be achieved, the resulting logical error probability arising from first-order error correction p_L is sufficiently low for most practical purposes. Thus, the focus of the generalization of such codes is rather the "encoding rate" of the code, the ratio between the number of logical bits of information encoded to the number of physical bits required. As previously mentioned, the family of classical first-order Hamming codes can encode $k = 2^r - r - 1$ logical bits using $n = 2^r - 1$ physical bits, achieving an encoding rate $R = 1 - \frac{r}{2^r - 1}$ which is actually optimal for a first order error correcting code. The focus of the generalization of quantum error correcting codes is different: rather, the goal is to improve the order of the code, that

is the number of independent errors that it can correct. Indeed, a very rough estimation of the physical error probability affecting a physical quantum bit of information over a relevant time-scale is typically in the range $10^{-2} - 10^{-3}$. While these numbers are impressively low, resulting from the extraordinary research that has been conducted over the past 20 years in the development of new materials, qubit design, quantum control, readout techniques, etc., there are still orders of magnitude higher than the typical error probabilities required for large-scale quantum computation, typically in the $10^{-14} - 10^{-15}$ range. The design of quantum error correcting codes that can detect and correct many more than one single error is thus essential to bridge the gap between the actual state of experimental research and the required robustness to perform useful quantum computations.

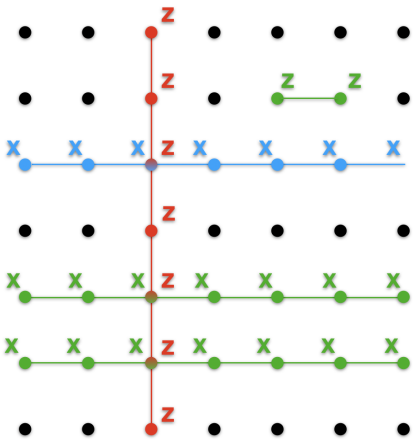


FIGURE 1.3: Layout of the [49,1,7] Bacon-Shor code. The 49 physical qubits (black dots) are arranged in a 7×7 lattice. Single qubit logical operators X_L and Z_L (red and blue) are built with products of physical Z and X operator in a column and a row. Errors are detected by measuring neighbouring Z operators and product of neighbouring rows of X operators (in green).

The natural extension of the $[9, 1, 3]$ Shor code was developed by Bacon [14], now known as the family of Bacon-Shor codes. It comes from the observation that the structure of the Shor code is based on the combination of two first order *repetition codes*. The inner repetition code is designed to protect against bit-flips, and the code words are

$$|0\rangle_L = |000\rangle$$
$$|1\rangle_L = |111\rangle.$$

Such a code can correct a bit-flip on any of the three qubits, but it cannot correct a single phase-flip. Even worse, because any single phase-flip error Z_i is actually a valid logical Z_L operator, the probability that the logical bit of information suffers from a logical phase-flip error is actually increased. The outer repetition code is

designed precisely to protect against phase-flips. Its code words are

$$|+\rangle_L = |++\!+\rangle$$
$$|-\rangle_L = |--\!-\rangle.$$

It is actually exactly the same code where the computational states $|0\rangle$ and $|1\rangle$ have been replaced by there in-phase and out-of-phase superpositions $|\pm\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$, sometimes referred to as the dual basis. The 9-qubit Bacon-Shor code arise from the *concatenation* of these two codes: each of the three physical qubits of the outer code is replaced by one logical qubit of the inner code, itself composed of three physical bits. By construction, the resulting code protects against any single X, Y or Z error, and by Theorem 2 against all single qubit errors. Having exposed the structure of the Shor code, its generalisation to an arbitrary distance family of codes follows straightforwardly by concatenating a distance n repetition code protecting against bit-flips with a distance n repetition code in the dual basis, protecting against phase-flips. The code words of the resulting (symmetric) $[n^2, 1, n]$ Bacon-Shor code are

$$|0\rangle_L = \sqrt{2}^{-n}(|0\rangle^{\otimes n} + |1\rangle^{\otimes n})^{\otimes n}$$
$$|1\rangle_L = \sqrt{2}^{-n}(|0\rangle^{\otimes n} - |1\rangle^{\otimes n})^{\otimes n}.$$

When the n^2 physical qubits are arranged in a 2D lattice as depicted in Figure 1.3, where the physical qubits in a row form a logical qubit of the inner repetition code, a logical X_L operator of minimum weight (in red) can be implemented by applying physical Z to all the physical qubits in a column. Symmetrically, a logical Z_L operator (in blue) can be formed with the product of the X operator in a row. The detection of the errors is achieved by measuring pairs of neighbouring Z operators in the rows and pairs of neighbouring X rows (in green). The Bacon-Shor code, regarded as a strict concatenation of two repetition code, requires the joint measurement of all the X operators in two neighbouring rows, which is an undesired feature. Fortunately, it is unnecessary. Indeed, as detailed in [14], [101], the Bacon-Shor code belong to the class of subsystem codes and it is enough to measure pairs of neighbouring X in the same column to reconstruct the value of the X operators in two neighbouring rows. Even though these neighbouring X operators do no longer commute with the code words, their measurement does not affect the logical encoded information but rather, just induce a change of basis for the logical code words.

For the purpose of appreciating the error correction construction proposed in this work, the theoretical material on error correcting codes presented so far is probably sufficient, given the simplicity of the codes that we intend to use. Nonetheless, before we move on to the challenges of the practical implementation of these codes,

and the path that proposed here to tackle them, let us pause for a moment to introduce the *stabilizer formalism*, developed by Gottesman [54], to which the Bacon-Shor codes belong.

The stabilizer formalism, which was developed as an elegant mean to describe and analyse *stabilizer codes*, is most easily explained with two level systems (qubits), although they apply to a larger class of quantum systems. The description of a stabilizer code that encodes k logical qubits into n physical qubits is specified by the *stabilizer group* $\mathcal{S}$, an abelian subgroup of the n-qubit Pauli group generated by $n-k$ independent Pauli operators. In the stabilizer formalism, an operator U is said to *stabilize* a quantum state $|\psi\rangle$ if $|\psi\rangle$ is a +1 eigenstate of U

$$U|\psi\rangle = |\psi\rangle.$$

The code space of a stabilizer code is defined as the set of states that are stabilized by every element in the stabilizer group $\mathcal{S}$, called the *stabilizers*. In other words, the code space is the common +1 eigenspace of all the stabilizers. Note that since $-I$ has no +1 eigenvector, it is necessary that $-I \notin \mathcal{S}$ for the code to be non-trivial, and since $\mathcal{S}$ is a group, the Pauli operators in $\mathcal{S}$ can only have ± 1 (and not $\pm i$) global phases. Consider a minimal generating set of the group $\mathcal{S}$, composed of $n-k$ Pauli operators. The two eigenspaces associated to the +1 and -1 eigenvalues of every Pauli operator in this set are necessarily of equal dimension 2^{n-1}, "splitting" the full Hilbert space of the n-qubit system in two. As the stabilizer group is abelian, the generators can be diagonalized in the same basis and the common +1 eigenspace of the generating set, the code space, is a subspace of dimension $2^n/2^{n-k} = 2^k$, encoding k logical qubits.

The set of stabilizers $\mathcal{S}$ is not just a nice theoretical way to describe the code space, it is actually the set of operators that ought to be measured to perform error correction (a generating subset of the stabilizer group). The stabilizers are Hermitian operators, so they can be measured, and since they all commute, they can be measured simultaneously. Whenever the state of the system is in the code space, the measurement of the stabilizers produces the $\{+1, ..., +1\}$ outcome, without disturbing the state of the system. Consider now the effect of a Pauli error E on a logical states lying in the code space $|\psi\rangle_L \in \mathcal{C}$. If the error E belongs to the stabilizer group, then by definition it leaves the state unchanged $E|\psi\rangle_L = |\psi\rangle_L$ so it does not need to be detected nor corrected. If it does not belong to the stabilizer group, there are two cases to consider. Since E is a Pauli operator, it either commutes or anti-commutes with all the stabilizers. If it anti-commutes with a stabilizer S, then the state $E|\psi_L\rangle$ is a -1 eigenstate of S

$$S(E|\psi\rangle_L) = -ES|\psi_L\rangle = -(E|\psi\rangle_L)$$

such that the measurement of the stabilizers will produce a -1 outcome for every stabilizer that anti-commutes with the error. The Knill-Laflamme condition for error correction, in this case, states that a correctable set of errors is such that every

error in the set produces a different pattern of $\{+1, -1\}$ outcomes, allowing to uniquely identify the error that occurred. If, on the other hand, the error E belongs to the set $N(\mathcal{S}) - \mathcal{S}$, where $N(\mathcal{S})$ is the normalizer of $\mathcal{S}$, *i.e* the set of operators that commutes with $\mathcal{S}$ (an operator commutes with a set of operators if it leaves it globally invariant), then the error E has a non-trivial effect on the logical state $|\psi\rangle_L$ while being undetectable. The operators in the set $N(\mathcal{S}) - \mathcal{S}$ are actually logical operators, as allow to manipulate the encoded information in a non-trivial way (see subsection 1.3.3).

Operating a quantum error correcting code requires the ability to measure the value of the stabilizer operators that reveal the errors. Typically, these operators are multi-qubit Pauli operators acting on the data qubits (the formalism can actually be extended to non-Pauli stabilizer operators as we will see later in this thesis, see Chapter 3). A Pauli stabilizer operator U (with real global phase) can be measured in a quantum non-demolition (QND) way using an additional ancilla qubit and a controlled-U operation, as depicted in Figure 1.4.

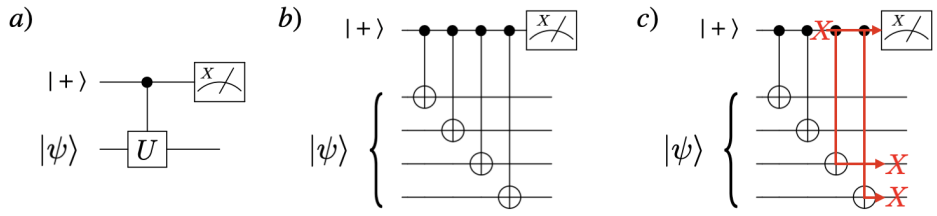


FIGURE 1.4: a) Quantum non-demolition measurement of the stabilizer operator U using an ancilla qubit. b) The required multi-qubit controlled-U operation can be implemented using a sequence of two-qubit gates. Here, the measurement of a $X_1X_2X_3X_4$ operator on the state $|\psi\rangle$ is performed using four CNOT gates. c) Propagation of a single X error on the ancilla qubit to two X errors in the block of qubits by propagation through the measurement circuit.

The controlled-U operation applied between the ancilla qubit in the state $|+\rangle$ and the block of qubits in the state $|\psi\rangle$ effectively maps the value of the stabilizer U on the ancilla qubit

$$\begin{aligned} CS(|+\rangle \otimes |\psi\rangle) &= \tfrac{1}{\sqrt{2}}(|0\rangle \otimes |\psi\rangle + |1\rangle \otimes U|\psi\rangle) \\ &= \tfrac{1}{\sqrt{2}}(|+\rangle \otimes \frac{I+U}{2}|\psi\rangle + |-\rangle \otimes \frac{I-U}{2}|\psi\rangle) \end{aligned}$$

such that the measurement of the X operator of the ancilla qubit reveals the value of the U operator, and projects the state of the block of qubits into the corresponding eigenspace (as $\frac{I\pm U}{2}$ is the projector on the ± 1 eigenspace). The stabilizer operators U typically act on at least two qubits, and most often on four or six qubits. In this case, one may worry that the required controlled-U operation required to implement a

QND measurement is too complex to engineer, but it can be realized by using a sequence of lower weight gates. For instance, in Figure 1.4-b), we depict the usual way one would measure the weight-4 stabilizer operator $U = X_1X_2X_3X_4$ using a sequence of four weight-2 CNOT gates rather than a single weight-five operation controlled-$X_1X_2X_3X_4$.

The measurement circuit depicted in Figure 1.4-c) allows us to discuss in detail a fundamental issue that we will deal with throughout the whole of this work, which is that of *fault-tolerance*. The foundational assumption behind the design of error correcting code is that the errors of the different physical components are independent, such that errors of high weight require the independent failure of many of these components and, consequently, occur with a very low probability. However, this assumption has no reason to hold when gates are applied between different physical components, because these gates may *copy* in a deterministic manner errors from a component to another one, thus increasing the probability of having errors of high weight. To illustrate this, consider for instance the measurement circuit of figure 1.4-b), and suppose that the ancilla qubit undergoes some X error after the two first CNOT gates have been executed as in c). Because the controlled-X gate consists in applying a X gate to the target conditioned on the control qubit being in the $|1\rangle$ state, an X error flipping the state of the control qubit will result in the X gate being applied erroneously (or not applied erroneously), such that in either case the target qubit also suffers from an X error after the CNOT gate is executed. Thus, because the stabilizer measurement circuit is built in such a way that the ancilla qubit interacts with more than one qubit of the encoded block, the probability of high weight errors becomes that of a single error. A circuit that is designed in such a way that it does not increase the weights of the errors beyond what is correctable by the code is called *fault-tolerant*.

In the early days of quantum error correction, several modifications of the measurement circuit of Figure 1.4 were proposed to make it fault-tolerant at the cost of using more ancillae qubits [110, 113, 72]. The intuition behind these constructions is that the ancilla block of qubits should reproduce the effect of a single ancilla qubit, but in such a way that no component of the ancilla block interact with more than a single qubit of the system, to prevent the ability of an error to spread to more than one qubit in the system. The "Steane-style" error correction [113] is only applicable to the codes for which the stabilizers are either composed of X operators exclusively, or of Z operators, called the CSS codes (after their inventors, Calderbank, Shor and Steane). The 9-qubit Shor code introduced above is a CSS code, for which the stabilizers are either tensor products of six X operators or tensor products of two Z operators. For these codes, Steane proposed a fault-tolerant method, depicted in Figure 1.5, to extract all of the information about the Z errors (or the X errors) in a single step, rather than measuring all the stabilizers separately. For the 9-qubit Shor

code, the information about the Z errors is extracted using a 9-qubit ancilla block prepared in the specific state $|0\rangle_L$. A round of nine CNOT gates is applied in a one-to-one correspondence (transversally) between this ancilla block and the qubits of the Shor code. Importantly, because the ancilla block is prepared in the $|0\rangle_L$ state and the CNOT gate is transversal for any CSS code, the round of CNOT gates has no effect on the logical encoded state $|\psi\rangle_L$. Thus, the only effect of these gates is to "copy" the Z errors from the logical encoded block to the ancilla block, where they are revealed by the value of the measurement of all the ancilla qubits.

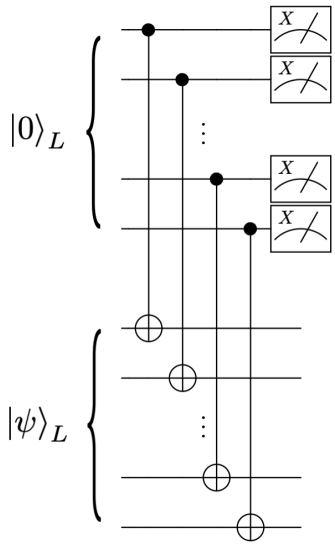


FIGURE 1.5: Circuit to extract information about the Z errors using Steane-style error correction for the 9-qubit Shor code.

In this work, however, the propagation of errors through the stabilizer measurement circuit will not be a concern (but it will be a major issue to address for other circuits that we design in Chapter 3). Let us assume for now that one has access to a qubit that only suffers from Z errors, and the motivations behind this choice will be explained later in this introduction. The detection of the Z errors on such qubits is realized with X-type stabilizers (that anti-commutes with the Z errors, while Z-type stabilizers would be blind to such errors). The measurement of X-type stabilizers as in Figure 1.4 is realized with a sequence of CNOT gates where the control qubit is the ancilla and the target qubits are those of the stabilizer code. While a CNOT gate propagates an X error from the control to the target qubit (and a Z error from the target qubit to the control, which is actually the mechanism behind an X-type stabilizer measurement), it does not propagate Z errors of the control to the target. Thus, if the ancilla qubit has strictly no X errors, then the circuits of 1.4 are fault-tolerant.

A second important issue arising from the fact that the ancilla qubits and

the measurement circuits are themselves prone to errors is that the value of the syndrome measurement is also subject to errors. Since these measurement outcomes are used to determine which errors have arisen, and hence which correction to apply, an error in a stabilizer measurement outcome may lead to a wrong correction being applied, thus introducing additional errors. For this reason, it is usually necessary to *repeat* the stabilizer measurements a certain number of times. Then, one needs to design a procedure to interpret all of the results of the measurement, to decide which correction needs to be applied. For instance, a typical procedure could consist in repeating the stabilizer measurement until d consecutive measurement outcomes agree. Indeed, by choosing the number d large enough (typically, linear in the code distance, but the precise number depends on the error model describing the ancilla), the probability that all of these measurements were wrong (and hence, that one interprets badly the outcome of the measurement, leading to a possibly high weight error on the system after correction) can be made sufficiently small by picking a large enough d. However, such procedures are not very practical, as when the distance of the code is increased, the probability to get such consecutive measurement agreement becomes very unlikely. Better procedures to interpret the stabilizer measurement outcomes, known as *decoding procedures*, can be designed. We provide in Chapter 4 an introduction to the most popular of these methods for stabilizer codes, based on mapping the problem of finding the most likely pattern of errors given an observed sequence of measurement outcomes to a problem in graph theory called *minimum weight perfect matching*.

1.3.3 Protected logical Gates

So far, we have discussed how quantum information could be protected from errors by encoding it into quantum error correcting code, in order to extend its lifetime. A logical qubit in which information is stored is called a quantum memory. However, operating a quantum computer requires not only the ability to retain faithfully the information, but most importantly to *process* it, that is to actually perform computational operations on this information. Actually, the most important metric is not so much the absolute value of the error probability afflicting the qubits, but rather, the typical number of operations that one can perform on these qubits before the inevitable errors destroy the information being processed. In this regard, the role of the extra protection provided by error correcting codes is to lower the probability of errors, thus effectively extending the lifetime of the encoded information, such that (typically long) quantum algorithms may be run within the lifetime of quantum information.

Most importantly, we now argue that information processing can be performed *directly* on the logical encoded information, without re-introducing errors with high probability. How this can be done is not so trivial: as the information is

encoded in highly non-local entangled state, it may seem that manipulating this information might require to first decode it, by bringing it back into a smaller quantum system, before applying some processing operation before re-encoding. However, doing so exposes the information to errors with high probability during the time it is not encoded. Worse, if the operation itself is noisy, then one hardly sees how the good level of protection provided by the code could be maintained during a computation. Fortunately, in a similar way that quantum error correcting codes yield very good logical qubits from many noisy physical qubits, *logical quantum gates* of arbitrarily high fidelity acting at the level of the logical qubit can be constructed from many noisy physical gates. Such logical gates can be used to act on the encoded information without requiring to decode/re-encode it. Such a use of error correction is somewhat new: indeed, classical error correction has found applications to redundantly encode information that needs to be sent through very noisy communication channels (typical applications include the internet, deep-space telecommunications, satellite broadcasting, etc.), but the typical gates error probabilities in classical hardware are such that *classical* computation does not require the use of logical encoded gates. Quantum gates, on the other hand, suffer from levels of noise that are typically of order 10^{-2} to 10^{-4} in state of the art experiment, far too noisy for the demanding requirements of large-scale quantum computation (say, 10^{-10} to 10^{-15} depending on the algorithm), such that using quantum error correction to improve the quality of the gates themselves seems inevitable.

When describing the way error correction works, we have implicitly assumed a non-trivial fact to be true. As emphasized previously, the basic principle of error correction is to increase the redundancy of the encoding of the logical information such that a higher number of independent errors are required to destroy it. Roughly speaking, this strategy can only be helpful if the amount of protection gained by using more physical systems surpasses the amount of errors added by using more physical systems. Indeed, this is only true if the basic components of the physical system are themselves suffering from errors with sufficiently small probability. This key result at the foundation of the theory of quantum error correction is known as the *accuracy threshold* [2, 73, 70]. We recall the theorem as stated in [91]

Theorem 4 (Threshold theorem for quantum computation) *A quantum circuit on n qubits and containing $p(n)$ gates may be simulated with probability of error at most ε using $\mathcal{O}(\log^c(p(n)/\varepsilon)p(n))$ gates (for some constant c) on hardware whose components fail with probability at most p, provided p is below some constant threshold, $p < p_{th}$, and given reasonable assumptions about the noise in the underlying hardware.*

The value of the threshold error probability p_{th} is actually a single number (as is usual in the literature) only when one assumes that all the error models for the various operations in the circuit can be expressed in terms of the same error probability p. For instance, a typical error model that is often studied is the *depolarizing*

error model where all components fail with equal probability p. In general, however, there could be n independent parameters describing error models for different types of operations in the circuit (for instance, the failure probability of an idle qubit could be much lower than that of a qubit that is acted upon, etc.), in which case the threshold is not given by a single number but rather by a hyperplane in the n-dimensional parameter space.

The *reasonable assumptions* about the noise in the underlying hardware on which the threshold theorem rely are actually a crucial hypothesis of the theorem, and the value of the threshold (when it is given by a single number) is highly dependent on these assumptions. Some of the most basic assumptions that typically need to hold true to implement a quantum error correcting code are ([33])

- **Constant error rate**. The strength of the noise acting on the qubits must be independent of the number of qubits in the computer, otherwise there cannot be a threshold.
- **Weakly correlated errors.** Errors must not be too strongly correlated, either in space or in time, as fault-tolerant procedures fail if errors are allowed to act simultaneously on many qubits of the same code block.
- **Parallel operation.** Quantum operations that act on different parts of the quantum computer can be performed at the same time.
- **Reusable memory.** Ancilla qubits can be reinitialized during the computation. The entropy introduced by the errors into the computer are transferred into the ancilla qubits, thus the ability to "flush out" this entropy by re-initializing the ancilla to provide fresh ancilla is essential.
- **Fast and accurate classical processing**. Compared to the speed and accuracy of the quantum computer, the speed and reliability of classical computations can be assumed to be infinitely fast and perfect. In this case, it is more advantageous to defer any computation that does not need to be quantum to the classical hardware, which includes *e.g* the decoding of the syndrome measurements to decide which correction to apply.

The above assumptions are usually required to hold true to perform arbitrarily long quantum computations, but they are not enough. Indeed, one also needs to be able to implement a *universal* set of *fault-tolerant* gates. The universality of the set of logical gates is required to implement arbitrary quantum algorithms. The fault-tolerance is required to implement gates with arbitrary accuracy, such that the computation can be made arbitrarily long. The design of a set of logical gates that is both universal and fault-tolerant, however, is challenging. As we have seen on the particular examples of the stabilizer measurement circuits, the major issue to overcome when designing circuits is that errors can propagate through these circuits, thus producing high weight errors with high probability, when the protection provided by

the code precisely relies on the fact that these high weight errors are expected to occur with extremely low probability. In this regard, one might wonder whether a universal set of gates could be implemented on a code using only transversal circuits, as by construction, these circuits do not increase the weight of errors and are thus fault-tolerant. Quite unfortunately, such a construction has been found to be impossible, as stated by the Eastin-Knill theorem [37].

Theorem 5 (Eastin-Knill theorem) *For any nontrivial local-error-detecting quantum code, the set of logical unitary product operators is not universal.*

For many quantum codes, the encoded version of the gates in the Clifford group can be implemented transversally on the code, which, by virtue of the Eastin-Knill theorem, prevents non-Clifford gates to be implemented transversally. Yet, the fault-tolerant but non-transversal construction of a non-Clifford gate is possible but is usually more expensive in terms of physical resources than the transversal gates. Thus, most of the focus of such constructions has been devoted to find the most efficient strategies to reduce the associated overhead. A long standing leading strategy inspired from gate teleportation techniques [55] is to prepare encoded versions of magic states and to consume these states as a non-Clifford resource during the computation [19]. In these techniques, the magic state can be *injected* in the code using logical Clifford gates to produce a logical non-Clifford gate and the fidelity of this non-Clifford gate is limited by the fidelity of the preparation of the magic state. Thus, the bottleneck of such techniques is usually to prepare very high fidelity magic states, which can be achieved *e.g* using a state *distillation* protocol, that produces a high fidelity state using many copies of low fidelity states. The cost of these techniques, initially very expensive in terms of hardware resources, has been greatly reduced thanks to many years of active research [43, 36, 35, 95, 61, 52, 84]. To this day, this approach to work around the Eastin-Knill theorem is undeniably regarded as the most promising, and many schemes, including the surface codes, rely on it.

In order to avoid magic state preparation, distillation and injection, and the costly overhead associated with it, another approach is to combine different codes that have different sets of transversal gates that might complete one another. An interesting example of this are the color codes: 2D color codes (including the 7-qubit Steane code) have transversal Clifford gates, while 3D color codes (including the 15-qubit Reed-Muller code) have a transversal non-Clifford gate, the T gate (but do not have a transversal Hadamard gate). Constructions that attempt to exploit these facts include concatenating these two codes [65], or combine 2D and 3D color codes in an architecture where a 2D color code would be augmented with 3D T-gate factories where logical qubits could undergo a T gate [17]. More generally, subsystems codes make it possible to deform the encoding in such a way that the information remains protected from errors but the set of allowed transversal gates

changes (see *e.g* the gauge color codes [18, 22, 76, 99] and other code-switching techniques [10]).

Along the same lines, a recent proposal proposed to use code deformation techniques to directly implement a fault-tolerant non-Clifford gate on the surface code [21]. Other strategies focus on circuits that are not transversal, yet can still be made fault-tolerant, such as the pieceable fault-tolerant EC where intermediate rounds of error correction are added in well-chosen locations of the circuit [130, 116], or the flag fault-tolerant EC where additional "flag" ancilla qubits are used to gain more information about the propagating errors [27, 24]. Interestingly, the use of flag qubits found applications in the preparation of high fidelity magic states without the need for state distillation, see *e.g* the preparation of H-type magic states on the Steane code and the color codes [25, 26].

The absolute value of the accuracy threshold is of great practical importance as it sets a target goal for the real-world implementation of quantum error correction. The first estimates of the accuracy threshold were obtained via an analytical analysis of the 7-qubit Steane code concatenated with itself, which produced thresholds about $p_{th} = 10^{-6}$ [69, 2, 74, 102]. These very low estimates of the threshold seemed to put error correction out of reach of experiments, but motivated the constructions of codes with higher thresholds. Indeed, it was discovered soon afterwards that some topological error correcting codes, such as surfaces codes [70], have much higher thresholds [33]: a rigorous lower bound around 10^{-4} can be proved and numerical simulations establish that the actual value of the threshold is around 10^{-2}. The 2D surfaces codes and related topological error correcting codes like the color codes still remain to this day the codes with the highest thresholds known, and the precise value of the threshold for these codes under many different sets of relevant assumptions and error models have been the subject of intensive research (see *e.g* the selected set of studies [105, 46, 41]). While this value remains experimentally challenging, it is now within reach of state of the art experiments (see *e.g* the recent demonstration of the 9-qubit Bacon-Shor code with trapped-ions [39]).

1.3.4 Bosonic qubits

From the first demonstration of the Cooper pair box in the early 2000s with a typical lifetime of a few nanoseconds, to the first implementation of the transmon with a lifetime of a few microseconds in 2007, and in the millisecond range by 2014, the quality of the superconducting qubits has been improved by over five orders of magnitude over the past two decades! The field of superconducting qubits is now very close to meet the demanding requirements of quantum error correcting codes, but the construction of a useful error correcting code will require that the lifetime properties of the qubit are actually *well below* the required threshold. Indeed, an error

correcting code made with qubits that are just "good enough" has to use an overwhelming number of these qubits, while the implementation of the same code with qubits that are one or two orders of magnitude longer lived makes the required size of the code much more realistic.

In "conventional" qubits, the quantum bit of information is stored in a strictly two level quantum system, like the spin of an electron. In the circuit quantum electrodynamics (cQED) framework, the two states used to encode a qubit are usually the two levels of lowest energy of a quantum *anharmonic* oscillator. The quantum harmonic oscillator (QHO) is the quantum version of the classical harmonic oscillator, implemented in the cQED framework by a LC circuit described by the Hamiltonian

$$\hat{H} = 4E_C\hat{n}^2 + \tfrac{1}{2}E_L\hat{\phi}^2 \tag{1.2}$$

where $E_C = e^2/(2C)$ is the charging energy, $E_L = (\phi_0/2\pi)^2/L$ the inductive energy and $\phi_0 = h/(2e)$ the superconducting magnetic flux quantum. This Hamiltonian can be re-written using the "ladder operators" $\hat{a}$ and $\hat{a}^\dagger$, also called the annihilation and creation operators

$$\hat{H} = \hbar\omega(\hat{a}^\dagger\hat{a} + \tfrac{1}{2})$$

defined as

$$\hat{a} = \tfrac{1}{2}(\frac{\hat{\phi}}{\phi_{\text{zpf}}} - i\frac{\hat{n}}{n_{\text{zpf}}})$$
$$\hat{a}^\dagger = \tfrac{1}{2}(\frac{\hat{\phi}}{\phi_{\text{zpf}}} + i\frac{\hat{n}}{n_{\text{zpf}}})$$

where $\phi_{\text{zpf}} = [2E_C/E_L]^{1/4}$, $n_{\text{zpf}} = [E_L/32E_C]^{1/4}$ are the zero-point fluctuations of the phase and charge variables, and $\omega = 1/\sqrt{LC}$ is the QHO's frequency. This Hamiltonian is diagonal in the Fock basis, also called the photon number basis $\{|n\rangle\}_{n\in\mathbb{N}}$

$$\hat{H}|n\rangle = E_n|n\rangle = \hbar\omega(n + \tfrac{1}{2})|n\rangle.$$

The fact that the eigenvalues of $\hat{H}$ are equally spaced, $E_n - E_{n-1} = \hbar\omega$ makes the use of the two lowest levels of the QHO a bad candidate for storing a quantum bit of information, as the transition between the $|0\rangle$ and $|1\rangle$ states cannot be addressed individually without addressing all the transitions between higher energy states $|n-1\rangle$ and $|n\rangle$. This can be fixed by replacing the linear inductance of the LC oscillator by a Josephson junction, a superconducting circuit element playing the role of a lossless non-linear inductor.

Because of this non-linearity, the oscillator becomes anharmonic and its energy levels are no longer evenly spaced. When the anharmonicity α, defined as the difference between the transition frequencies of the ground to first excited state and first to second excited state is sufficiently large compared to the transition frequency,

the two states $\{|0\rangle, |1\rangle\}$ can be selectively addressed and be the recipient of a qubit. This general recipe has been successfully used to father the so-called *transmon qubit*, which is undeniably the most popular superconducting qubit up to this day, as well as a family of superconducting qubits including the flux qubit, the fluxonium qubit, and others, described in the recent reviews [75, 71].

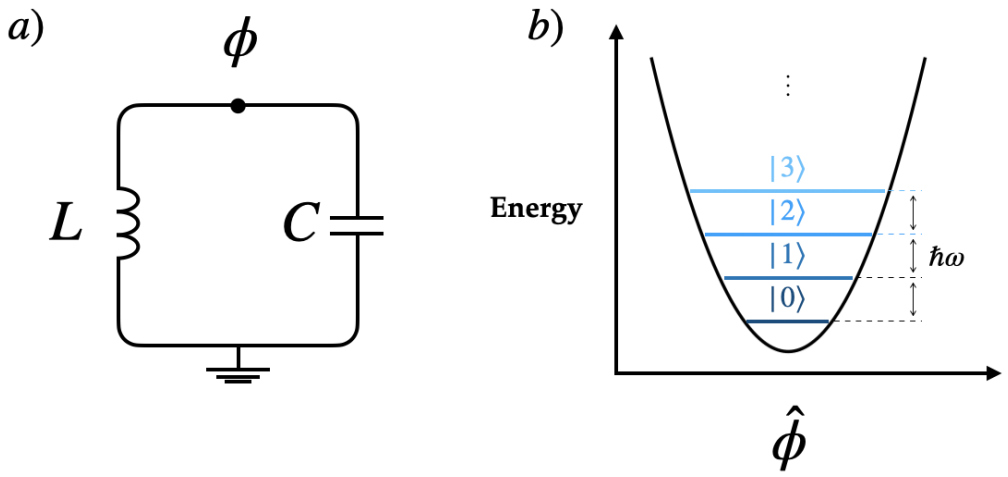


FIGURE 1.6: (a) Circuit representation of the LC oscillator (b) The energy spectrum of the QHO as a function of the superconducting phase $\hat{\phi}$ across the inductor.

A different strategy, that we follow in this work, is to rely instead on *bosonic qubits* [29]. A bosonic qubit is stored using all of the infinite dimensional Hilbert space of the QHO. The choice of the computational basis that encodes the quantum information is referred to as the *bosonic code*. There are three main advantages to this approach. First, the use of an infinite dimensional Hilbert space of a single quantum systems provides a large "quantum space" to protect the information but also to perform operations, with a very "compact" hardware. The property, named *hardware efficiency*, is a key ingredient in the building of large-scale architectures. The second advantage is the intrinsic long lifetimes of the cavity modes that are used for the bosonic qubits. It is hoped that the bosonic qubit will be able to improve the current lifetimes over more conventional two-level systems and thus prove to be better building blocks for quantum error correcting codes. Last, the largely dominant error channel affecting bosonic qubits is photon loss. Since there is mostly a single error channel, designing bosonic qubits that are robust against this particular error is perhaps simpler and could prove to be enough to build long-lived bosonic qubits.

The encoding of the quantum bit of information into the two lowest levels of an anharmonic quantum oscillator, $|0\rangle_L = |0\rangle, |1\rangle_L = |1\rangle$ provides no protection against photon loss. In this case, the single loss of a photon, described by the annihilation operator $\hat{a}$, destroys the encoded bit of information

$$\hat{a}|\psi_L\rangle = \hat{a}(\alpha|0\rangle_L + \beta|1\rangle_L) \propto |0\rangle_L.$$

such that the rate of loss of the encoded information will be essentially the same as the bare rate of single photon loss κ of the oscillator. The use of multi-photonic states of the QHO as a computational basis is precisely to increase the robustness to photon loss errors. Many bosonic encodings have been proposed to achieve this, with particular focus on their robustness against photon loss errors, and against thermal excitations or dephasing errors. A thorough review of this approach is proposed in [66].

The *kitten code*, realized experimentally in [64], encodes a quantum bit of information using the two computational states defined in the Fock basis as

$$|0\rangle_L = \frac{1}{\sqrt{2}}(|0\rangle + |4\rangle)$$
$$|1\rangle_L = |2\rangle.$$

Note that the code space is supported on Fock states with an even number of photons, in order to protect the bit of information against the loss of a single photon $\hat{a}$. Such an event is detected by the measurement of the *parity* of the number of photons operator $\hat{P} = (-1)^{\hat{a}^\dagger \hat{a}}$, which takes the value $+1$ (resp. -1) on any state in the span of $\{|2n\rangle\}$ (resp. $\{|2n+1\rangle\}$). Importantly, the two computational states have the same average number of photon $\bar{n} = {}_L\langle 0|\hat{a}^\dagger \hat{a}|0\rangle_L = {}_L\langle 1|\hat{a}^\dagger \hat{a}|1\rangle_L = 2$, a property that needs to be fulfilled to ensure that the logical information is not distorted upon the loss of a photon. Indeed, the logical state $|\psi\rangle_L = \alpha|0\rangle_L + \beta|1\rangle_L$ evolves under a single photon loss as

$$|\psi\rangle_L \xrightarrow{\hat{a}} \frac{\hat{a}|\psi\rangle_L}{\sqrt{{}_L\langle\psi|\hat{a}^\dagger \hat{a}|\psi\rangle_L}} = \frac{1}{\sqrt{2}}(\sqrt{2}\alpha|3\rangle + \sqrt{2}\beta|1\rangle) = \alpha|3\rangle + \beta|1\rangle$$

and the code words can be restored without losing the quantum bit of information α, β.

By observing that the kitten code cannot protect against the loss of two photons $\hat{a}^2$ because only a single level separates the code words in the Fock space, the family of *binomial codes* was developed in [86] to protect against a larger set of errors containing up to L photon loss errors $\hat{a}$, G photon gain $\hat{a}^\dagger$ and D dephasing errors $\hat{a}^\dagger \hat{a}$ by using a $S = L + G$ spacing in the Fock space. The code words are given by

$$|W_{\uparrow/\downarrow}\rangle = \frac{1}{2^N} \sum_{p \text{ even/odd}}^{[0,N+1]} \sqrt{\binom{p}{N+1}} |p(S+1)\rangle.$$

where $N = \max(L, G, 2D)$ is called the maximum order. The choice of the binomial coefficients after which the code is named ensures that all the first l moments, $l \leq \max(L, G)$ moments of the number operator are equal for the two logical code words $\langle W_\uparrow|(\hat{a}^\dagger \hat{a})^l|W_\uparrow\rangle = \langle W_\downarrow|(\hat{a}^\dagger \hat{a})^l|W_\downarrow\rangle$. The error detection of this extended set of

correctable errors requires the measurement of a "generalized photon number parity", the photon number modulo $S+1$.

The binomial code belongs to the general class of *rotation-symmetric codes* [58] that includes all the codes that are invariant under a discrete rotation in phase space described by the discrete rotation operator

$$\hat{R}_N = e^{i(2\pi/N)\hat{a}^\dagger\hat{a}}$$

where the discrete number N is the order of the rotation symmetry of the code. A general recipe to construct an order-N bosonic rotation code is to pick a certain state $|\Theta\rangle$ called the *primitive* of the code and to form the two computational states by symmetrization

$$|0\rangle_L = \frac{1}{\sqrt{\mathcal{N}_0}} \sum_{m=0}^{2N-1} e^{i(m\pi/N)\hat{a}^\dagger\hat{a}}|\Theta\rangle$$
$$|1\rangle_L = \frac{1}{\sqrt{\mathcal{N}_1}} \sum_{m=0}^{2N-1} (-1)^m e^{i(m\pi/N)\hat{a}^\dagger\hat{a}}|\Theta\rangle$$

where the primitive $|\Theta\rangle$ can be any state that has support on at least one of both the $|2kN\rangle$ and the $|(2k+1)N\rangle$ Fock states for the code words to be non-trivial. The protection provided by these codes can be appreciated in terms of the number distance $d_n = N$ that corresponds to the distance between the code words in the Fock space, and thus quantifies the maximum number of photon loss errors that can be detected $\hat{a}^k$ with $k < d_n$, and in terms of the rotational distance $d_\theta = \pi/N$ that (roughly speaking) quantifies the maximum angle θ of a detectable rotation error of the form $e^{i\theta\hat{a}^\dagger\hat{a}}$.

Although the particular choice of the discretization of errors does not affect a bosonic code error correcting capacity, the intuition of a design of a bosonic code depends much on the form of the error model considered. The codes we have seen so far were thought to protect against errors expressed in terms of multiple photon loss $\hat{a}^k$, photon gain $\hat{a}^{\dagger k}$ or dephasing $(\hat{a}^\dagger\hat{a})^k$ errors. Another model to represent the errors of an imperfect QHO is to express them in terms of shifts of the values of the canonical position operator $\hat{q} = (\hat{a}+\hat{a}^\dagger)/\sqrt{2}$ and momentum operator $\hat{p} = (\hat{a}-\hat{a}^\dagger)/i\sqrt{2}$ of the QHO. One of the earliest bosonic encodings [56], named the *GKP code* as a reference to its authors Gottesman, Kitaev and Preskill, is precisely crafted to protect against such shifts. The "ideal" code words, sometimes referred to as the code words of the perfect GKP code, are given by a coherent sum of an infinite number of position eigenstates

$$|0\rangle_L = \sum_{s=-\infty}^{+\infty} |q = \alpha 2s\rangle$$
$$|1\rangle_L = \sum_{s=-\infty}^{+\infty} |q = \alpha(1+2s)\rangle$$

where $\hat{q}|q = q\rangle = q|q = q\rangle$ is the eigenstate of the position operator with eigenvalue q and $\alpha \in \mathbb{R}_+^*$ is a free parameter that can be varied to tailor the code asymmetry. The dual basis states $|\pm\rangle_L = (|0\rangle_L \pm |1\rangle_L)/\sqrt{2}$ are then given by infinite sums of momentum eigenstates

$$|+\rangle_L = \sum_{s=-\infty}^{+\infty} |p = \frac{\pi}{\alpha} 2s\rangle$$
$$|-\rangle_L = \sum_{s=-\infty}^{+\infty} |p = \frac{\pi}{\alpha}(1+2s)\rangle.$$

As opposed to the rotation-symmetric codes that we discussed above, this code in *translation-symmetric* as one can readily check from the expression of the computational states that they are invariant under a shift of the value of q by 2α or of the value of p by $2\pi/\alpha$. The symmetric GKP code correspond to the choice $\alpha = \sqrt{\pi}$ such that the spacing of the eigenstates supporting the code words in the phase space is the same for both quadratures.

While the GKP code is most easily introduced using the eigenstates of the position and momentum operators, these states are actually unphysical (their energy is not finite) and the computational states as introduced above are not normalized. The realistic version of the GKP code words is superposition of finitely squeezed states that are approximate eigenstates of the position and momentum operators. Because of the properties of a Fourier transform, the finite squeezing σ of the position (resp. momentum) eigenstates results in a Gaussian envelope of width $1/\sigma$ limiting the amplitude of the momentum (resp. position) eigenstates and the realistic code words (recently implemented experimentally for the first time using trapped ions [40], followed by a cQED implementation in [23]) are

$$|0\rangle_L = \mathcal{N}_0 \sum_{s=-\infty}^{+\infty} e^{-\frac{\sigma^2(2\alpha s)^2}{2}} \int_{-\infty}^{+\infty} \mathrm{d}q e^{-\frac{(q-2\alpha s)^2}{2\sigma^2}} |q = q\rangle$$
$$|1\rangle_L = \mathcal{N}_1 \sum_{s=-\infty}^{+\infty} e^{-\frac{\sigma^2(\alpha(1+2s))^2}{2}} \int_{-\infty}^{+\infty} \mathrm{d}q e^{-\frac{(q-\alpha(1+2s))^2}{2\sigma^2}} |q = q\rangle.$$

To understand the error correcting capability of the perfect GKP code, one can observe that the two code words $|0\rangle_L$ and $|1\rangle_L$ (resp. $|\pm\rangle_L$) have disjoint support in the phase-space and that the logical Z_L operation (resp. X_L) amounts to shift the value of the position (resp. momentum) operator by α (resp. π/α). Provided that the natural errors of the QHO, described in terms of shifts of the value of the position and momentum operators, occur sufficiently slow with respect to the typical measurement time of the error syndrome, the GKP encoding produces a bosonic qubit very well-protected against both bit-flip and phase-flip errors. More precisely, by measuring both the position operator $\hat{q}$ modulo α and the momentum operator $\hat{p}$ modulo π/α, one can detect and accurately correct any shift in both quadratures Δq

and Δp provided

$$|\Delta q| < \frac{\alpha}{2}$$
$$|\Delta p| < \frac{\pi}{2\alpha}.$$

An important point is that, while the two position and momentum operators $\hat{q}$ and $\hat{p}$ obey the canonical commutation rule $[\hat{q}, \hat{p}] = i$ and thus cannot be measured simultaneously, the position *modulo* α operator and momentum *modulo* π/α do commute and can thus be measured simultaneously. An unfortunate consequence of the Heisenberg uncertainty principle is that the shifts in momentum and position that the code can correct obey the condition

$$\Delta p \Delta q < \frac{\pi}{4}$$

such that there is no choice of code words that can produce a qubit with arbitrarily low bit-flip error and phase-flip error rates. Yet, for practical purposes this encoding is a very promising candidate to produce long lived bosonic qubits. Actually, even though the protection provided by the code is most easily understood when the errors are expressed as displacements of the field quadratures, the GKP encoding provides an excellent protection against the photon loss error channel $\hat{a}$, as has been demonstrated numerically [7]. In order to reach the extremely low levels of noise required for large-scale quantum computation, the concatenation of the GKP code with a "traditional" quantum error correcting code will remain inevitable, as well as with any other bosonic encoding. In this approach, the use of bosonic qubits such as the GKP is expected to reduce the total hardware needed because of the better lifetimes properties of the "base qubits". Indeed, even though the use of a higher layer of error correction is theoretically useful as soon as the physical error probability of the physical qubit p is as low as the threshold value p_{th}, the number of physical qubits required to reach a target logical error probability becomes experimentally reasonable only if the value of p is actually well below p_{th}. These constructions, and more particularly the "GKP - surface code" ticket, have been thoroughly studied [48, 49, 123, 62, 94, 117], with a particular focus on the overall hardware overhead reduction that can be expected from this construction, and closely related, to the value of fault-tolerant thresholds for various noise models.

The observation of the above examples shows that the working principle behind the protection provided by bosonic codes is somehow similar to that of the traditional quantum error correcting codes. The basic idea is to "hide" the valuable quantum information in from the eyes of the environment. Since the natural errors are typically local, the bosonic codes rely on explicitly non-local states to encode the information, much the same as the traditional quantum error correcting code rely on maximally entangled states across many different quantum systems. While the

GKP encoding is the bosonic encoding that provides the best protection against the pure photon loss error channel when only a single bosonic mode is considered [7], the observation that the bosonic qubit needs to be embedded in an active error correction procedure to achieve the demanding requirements of large-scale quantum computation opens the path to a very different strategy in the use of the inner layer of bosonic error correction.

The encoding that will be the main focus of this work, that belongs to the family of *cat codes*, is the building block of a scheme that aims to achieve a similar, or perhaps even greater, overall hardware overhead reduction by leveraging the use of the bottom bosonic encoding in a different manner. The philosophy of such encodings is to separate the logical computational states of the bosonic qubit $|0\rangle_L$, $|1\rangle_L$ arbitrarily far away (in theory) in the phase space. The reward in doing this is that the effective rate of bit-flip errors induced by any natural errors of the QHO is *exponentially* suppressed with the "phase-space distance" that separates these two states. As we will see, the counterpart of this approach is that the distance of dual states $|\pm\rangle_L$ becomes closer as the distance between the computational states is increased, resulting in a linearly enhanced effective phase-flip error rate. Somehow, the goal of using such an encoding is to produce a base quantum bit for which the noise is almost purely "quantum", in the sense that only a single type of error, the phase-flip, affects the qubit. In practice, the two types of quantum errors will still affect the qubit, but the choice of a sufficiently large phase-space distance between the states $|0\rangle_L$ and $|1\rangle_L$ can produce a bosonic qubit for which the bit-flip error probability becomes extremely small, say close to 10^{-10}, while maintaining the remaining phase-flip error probability sufficiently low, typically around 1%. In this approach, the purpose of the higher level of error correction can thus be entirely devoted to correct for the phase-flip errors. Hence, any classical error correcting code correcting for a single type of error could in principle be promoted to an effective quantum error correcting code and used in a concatenated scheme with cat qubits, producing an overall scheme with undeniable advantages. First, the hardware cost of the higher code automatically benefits from the fact that a single error oughts to be corrected. Second, the physical layout of the qubits could be simplified as the topology of code correcting only one type of error is intrinsically simpler than that of one correcting two types of error. Finally, the real-time detection, processing and correction of errors, known in the literature as the *decoding* of a code, is certainly an easier task for such a code.

Schemes tailored for biased noise qubits have been studied before [8, 9, 129, 120, 122, 121, 11]. Indeed, such a structure in the noise naturally arises *e.g* when the computational basis states are chosen to be the energy eigenstates of the underlying quantum system, in which case phase-flip errors arise from the loss of coherence in the computational basis due to entanglement with uncontrolled degrees of freedom of the environment, while the bit-flip errors are induced by energy exchange with

the environment, which typically happens at a much lower rate. Natural systems that present a biased noise structure are, for instance, certain superconducting qubits [9], NV centers in diamond [28], spin qubits [128], trapped ions [93], quantum dots [112], and many more. While the noise bias, defined as the ratio between the phase-flip error probability to the bit-flip error probability $\eta = p_Z/p_X$ can be as high as $10^3 - 10^4$ in these systems, the value of the low bit-flip error rate is still too high to be ignored by the error correcting code. Rather, in these cases, the optimal choice is to construct a code or a hierarchy of codes that is tailored to the value of the noise bias. One example of such a construction [8, 9] is to first use a repetition code protecting only against the dominant phase-flip error to effectively *unbias* the error channel, producing a very good logical qubit with balanced noise that is then used within a second code (a CSS code in these codes). A slightly different approach to take into account the native noise bias is to use asymmetric code tailored for this asymmetry, such as an asymmetric Bacon-Shor code [90, 20]. Yet another strategy that has been studied recently is to modify the surface code to improve its capacity to account for a noise bias, resulting in improved thresholds and hence reduced overhead [120, 122, 121, 11].

A crucial difference between the works cited above and the construction presented in this work is that the noise bias of the cat qubit is *tunable*, and can be made very large by taking the two computational states sufficiently far away in the phase-space. Accordingly, the overall strategy is rather different. Indeed, the exponential suppression of effective bit-flip errors and the linear increase of effective phase-flip errors on the cat qubit when the amplitude of the coherent states is increased reminds us of the scalings one would get using "traditional" qubits in a repetition code against bit-flips. Consider a qubit undergoing bit-flip (resp. phase-flip) errors per unit of time with probability p_X (resp. p_Z), used in a distance d repetition code protecting against bit-flips ($|0\rangle_L = |0\rangle^{\otimes d}$, $|1\rangle_L = |1\rangle^{\otimes d}$).

Assuming perfect error detection and correction for simplicity, the resulting logical bit-flip and phase-flip error probabilities are given by ($p_Z \ll 1$)

$$\mathbb{P}[X_L] = \mathbb{P}[\text{a majority of the qubits bit-flipped}] = \sum_{k=\frac{d+1}{2}}^{d} \binom{d}{k} p_X^k (1-p_X)^{d-k}$$

$$\mathbb{P}[Z_L] = \mathbb{P}[\text{any single qubit phase-flipped}] \sim d \times p_Z$$

such that the logical bit-flip error probability $\mathbb{P}[X_L]$ is suppressed exponentially in d at the cost of a linear increase in $\mathbb{P}[Z_L]$.

With this in mind, the cat qubit encoding can thus be thought of as a hardware-efficient implementation of a repetition code protecting against bit-flips. Indeed, the same scaling is achieved but using only one single quantum system, the QHO, where the code distance is now played by the mean number of photons $\bar{n}$ in the oscillator, instead of using d different two-level quantum systems.

In order to achieve full protection of the quantum bit of information, we follow the construction of Bacon-Shor codes and propose to embed several cat qubits in a repetition code now protecting against phase-flips, to produce a logical qubit that we call the *repetition cat qubit*. This approach is hardware efficient in the sense that the use of a bosonic qubit, the cat qubit, that "focuses" the natural errors of the oscillator into a single quantum error for the qubit encoded in the cat subspace, allowed us to replace the concatenation of two repetition codes in dual basis by only a single repetition code, the outer one, where each of the "base" qubits is a cat qubit providing itself a protection similar to an inner repetition code.

1.4 Two-photon dissipative cat qubit

1.4.1 Encoding of a cat qubit

The primitives for the cat qubit encoding are the two coherent states of opposite phase $|\alpha\rangle$ and $|-\alpha\rangle$, where the coherent state $|\alpha\rangle$ is parametrized by the complex number $\alpha \in \mathbb{C}$ [30, 80]. The amplitude of this complex number α, called the amplitude of the cat, is actually related to the average number of photons in the coherent state $\bar{n} = \langle\alpha|\hat{a}^\dagger\hat{a}|\alpha\rangle = |\alpha|^2$ and both notations $\bar{n}$ and $|\alpha|^2$ are used in this manuscript to denote the average number of photons in the cat state. A coherent state is expanded in the Fock basis as

$$|\alpha\rangle = e^{-\frac{-|\alpha|^2}{2}} \sum_{n\in\mathbb{N}} \frac{\alpha^n}{\sqrt{n!}}|n\rangle.$$

The coherent states $|\alpha\rangle$ and $|-\alpha\rangle$ are only orthogonal in the limit $|\alpha|^2 = +\infty$, as $\langle-\alpha|\alpha\rangle = e^{-2|\alpha|^2}$. An orthogonal basis for the logical subspace is composed of the in-phase and out-of-phase superposition of these two states, the so-called *Schrödinger cat states* after which the encoding was named

$$|\mathcal{C}_\alpha^+\rangle = \frac{1}{\sqrt{2(1+e^{-2|\alpha|^2})}}(|\alpha\rangle + |-\alpha\rangle) = \frac{e^{-\frac{-|\alpha|^2}{2}}}{\sqrt{2(1+e^{-2|\alpha|^2})}} \sum_{n\in\mathbb{N}} \frac{\alpha^{2n}}{\sqrt{(2n)!}}|2n\rangle$$

$$|\mathcal{C}_\alpha^-\rangle = \frac{1}{\sqrt{2(1-e^{-2|\alpha|^2})}}(|\alpha\rangle - |-\alpha\rangle) = \frac{e^{-\frac{-|\alpha|^2}{2}}}{\sqrt{2(1-e^{-2|\alpha|^2})}} \sum_{n\in\mathbb{N}} \frac{\alpha^{2n+1}}{\sqrt{(2n+1)!}}|2n+1\rangle.$$

The state $|\mathcal{C}_\alpha^+\rangle$ (resp. $|\mathcal{C}_\alpha^-\rangle$) is referred to as the *even cat* (resp. *odd cat*) as it is supported on even Fock states (resp. odd Fock states) which makes it clear that these two states are orthogonal for all values of $|\alpha|^2$. We follow the convention of our recent works [83, 60, 59] where these two cat states are chosen to be the eigenstates

of the X Pauli operator for the cat qubit

$$|+\rangle_L = |\mathcal{C}_\alpha^+\rangle$$
$$|-\rangle_L = |\mathcal{C}_\alpha^-\rangle.$$

With this choice, the computational states are exponentially close in $|\alpha|^2$ to the coherent states $|\pm\alpha\rangle$, while being strictly orthogonal to each other

$$|0\rangle_L = \tfrac{1}{\sqrt{2}}(|\mathcal{C}_\alpha^+\rangle + |\mathcal{C}_\alpha^-\rangle) = |\alpha\rangle + \mathcal{O}(e^{-2|\alpha|^2})$$
$$|1\rangle_L = \tfrac{1}{\sqrt{2}}(|\mathcal{C}_\alpha^+\rangle - |\mathcal{C}_\alpha^-\rangle) = |-\alpha\rangle + \mathcal{O}(e^{-2|\alpha|^2}).$$

We now recall the intuition of why such an encoding provides an arbitrary good protection against bit-flips, but no protection at all against phase-flips. Because the coherent states $|\pm\alpha\rangle$ are eigenstates of the annihilation operator $\hat{a}$ with eigenvalue α, $\hat{a}|\alpha\rangle = \alpha|\alpha\rangle$, an initial coherent state of amplitude α that undergoes photon losses at rate κ evolves after a time t as $|\alpha e^{-\frac{\kappa}{2}t}\rangle$: while the amplitude of the coherent state shrinks, the state is not distorted and is still a coherent state. This effect can be countered by continuously re-injecting energy in the case state, which in the case of a coherent state can be achieved with a resonant drive on the lossy cavity of appropriate amplitude. As a consequence, the exponentially small overlap between the two coherent states of same amplitude and opposite phase $|\pm\alpha\rangle$ is preserved even in the presence of single photon loss. More generally, the exponential suppression of bit-flip errors holds for any natural error of the oscillator that acts *locally* in the phase-space, as it cannot induce a population transfer between the far distance states $|\pm\alpha\rangle$.

The calculation of the precise rate of the exponential suppression of bit-flips induced by any local operator is a hard problem. A rigorous discussion of the bit-flip suppression can be found in [31]. In this thesis, we give numerical evidence for this by performing process tomography for both the idle qubit case and all the operations on cat qubits, in presence of the typical error processes of the QHO, in Chapter 2.

Given the state $|\psi\rangle$ of a QHO that loses photons at a rate κ, the probability to lose a photon during a small time $\delta t \ll \kappa^{-1}$ is given by the mean value of the operator $\kappa\delta t\hat{a}^\dagger\hat{a}$

$$\mathbb{P}[\text{photon loss event during } \delta t] = \kappa\delta t\langle\psi|\hat{a}^\dagger\hat{a}|\psi\rangle = \bar{n}\kappa\delta t.$$

For a two-component cat qubit, the loss of a single photon results in a phase-flip error. Indeed, the eigenstates of the Pauli X operator $|\pm\rangle$ are given by the even and odd cat states $|\mathcal{C}_\alpha^\pm\rangle$, and the annihilation operators acts as

$$\hat{a}|\mathcal{C}_\alpha^\pm\rangle = \alpha t^{\pm 1}|\mathcal{C}_\alpha^\mp\rangle$$

where $t^2 = \tanh(|\alpha^2|) = 1 + \mathcal{O}(e^{-2|\alpha|^2})$. This implies that the effective phase-flip rate induced by photon loss (and similarly, by all the natural errors acting locally in the phase-space) increases linearly with the mean number of photons in the coherent states $\bar{n} = |\alpha|^2$. The cat code based on only two coherent states $|\pm\alpha\rangle$ is sometimes referred to as the *two components* cat code. It is a particular case of the more general rotation-symmetric bosonic codes introduced in subsection 1.3.4, where the primitive is the coherent state $|\alpha\rangle$ and the order of the symmetry is $N = 1$. Using the same primitive but a 2-order symmetry produces the *four components* cat qubit with (dual) code words

$$|+\rangle_L = \frac{1}{\sqrt{\mathcal{N}_0}} \sum_{m=0}^{3} e^{i(m\pi/2)\hat{a}^\dagger\hat{a}}|\alpha\rangle = \frac{1}{\sqrt{\mathcal{N}_0}}(|\alpha\rangle + |i\alpha\rangle + |-\alpha\rangle + |-i\alpha\rangle)$$
$$|-\rangle_L = (-1)^m \frac{1}{\sqrt{\mathcal{N}_1}} \sum_{m=0}^{3} e^{i(m\pi/2)\hat{a}^\dagger\hat{a}}|\alpha\rangle = \frac{1}{\sqrt{\mathcal{N}_1}}(|\alpha\rangle - |i\alpha\rangle + |-\alpha\rangle - |-i\alpha\rangle).$$

This encoding, proposed in [80], led to the first experimental demonstration of QEC beyond the "break-even" point where the lifetime of a quantum bit exceeds that of any of its constituent parts [96]. One can check by expanding the code words in the Fock basis that both code words are supported on even photon number Fock states, congruent to 0 modulo 4 (resp. 2 modulo 4) for the $|0\rangle_L$ state (resp. the $|1\rangle_L$ state). This spacing provides a correction against the loss of a single photon, such that a logical phase-flip induced by photon loss is in this case a second order process, occurring with a much lower probability. This improved protection comes at the cost of a more complicated encoding. Importantly, the distance in the phase-space between the different components of the cat still increase with the size of the cat $\bar{n}$ such that the bit-flip error rate induced by any error acting locally in the phase-space is exponentially suppressed (at the cost of a linear increase of the phase-flip error rate). Following the general rotation-symmetric codes construction [58] and as already detailed in [80], one can construct the $2N$ *components* cat code using two superpositions of $2N$ coherent states of amplitude α uniformly spaced on a circle, with equal $+1$ phases for the $|+\rangle_L$ state and alternating phase for the $|1\rangle_L$ state, that protects against $N-1$ photon loss events. The focus of this work is based on the two components cat code, which produces the largest noise bias, but for which a large class of gates can be performed. In the rest of this thesis, the term cat qubit or cat code will implicitly refer to this code.

1.4.2 Autonomous stabilization of a cat qubit

While the phase-flips errors damaging the cat qubit will be taken care of by the means of an active layer of error correction as we detail in Chapter 3, the exponential suppression of bit-flip errors is autonomously realized by the process engineered to stabilize the cat qubit encoding manifold. This may be thought as a "continuous"

version of quantum error correction. The "discrete" version of quantum error correction that we have considered so far, like the stabilizer codes, relies on the detection of errors through measurements of carefully chosen operators, interpretations of the outcome patterns (the "decoding" of the code) and subsequent correction. On the other hand, a continuous quantum error correcting protocol [3] relies on a continuous monitoring of the state of the system through weak measurement and continuous correction via a feedback loop. A particular realization of continuous error correction relies on *reservoir engineering*. In the theory of open quantum systems, the *reservoir*, or the bath, usually refers to a large quantum system modelling the effect of the environment on the quantum system of interest. The *engineering* of the reservoir is a technique that consists in coupling the quantum system of interest to a dissipative reservoir through a carefully chosen interaction. In a certain sense, this amounts to allow the environment to probe the system, which usually leads to dissipative effects and decoherence, but only through a restricted interaction. When implemented successfully, this allows us to transfer the entropy created by the errors in the system to the reservoir, which is then evacuated through dissipation. This idea has been applied *e.g* to the autonomous stabilization of the ground state of a transmon qubit [51] or of an entangled Bell state between two transmon qubits [108].

The dissipative dynamics that stabilizes autonomously the cat qubit encoding, first proposed in [87] and realized experimentally in [78, 118, 83] is described by the following Lindblad master equation

$$\frac{d\rho}{dt} = \kappa_2 \mathcal{D}[\hat{a}^2 - \alpha^2]\rho \tag{1.3}$$

where the super-operator $\mathcal{D}$ is given by

$$\mathcal{D}[\hat{L}]\cdot = \hat{L}\cdot\hat{L}^\dagger - \tfrac{1}{2}\hat{L}^\dagger\hat{L}\cdot - \tfrac{1}{2}\cdot\hat{L}^\dagger\hat{L}$$

and κ_2 is the rate of the two-photon dissipation. The kernel of the dissipative operator $\hat{a}^2 - \alpha^2$, also called the Lindblad operator, is precisely the code space of the cat qubit as a consequence of the coherent states being eigenstates of the annihilation operator $\hat{a}$

$$(\hat{a}^2 - \alpha^2)|\pm\alpha\rangle = 0.$$

The dissipative evolution of a system governed by a Lindbladian that has more than one steady states is somehow the dissipative analog of Hamiltonians with multiple ground states, see *e.g* [4]. The steady states of such dynamics, that depend on the initial state of the system, can be conveniently expressed by identifying invariant operators of the evolution in the Heisenberg picture [5, 6]. Here, because the stable manifold is two-dimensional, it is enough to determine three degrees of freedom, that correspond to the population on one of the cat states $|\mathcal{C}_\alpha^\pm\rangle$ and to the complex coherence between these two cats. It was shown in [5, 87] that the infinite time steady

states corresponding to the initial state (possibly mixed) $\rho(t=0)=\rho_0$ is given by

$$\rho(t=\infty)=c_{++}|\mathcal{C}_\alpha^+\rangle\langle\mathcal{C}_\alpha^+|+(1-c_{++})|\mathcal{C}_\alpha^-\rangle\langle\mathcal{C}_\alpha^-|+c_{+-}|\mathcal{C}_\alpha^+\rangle\langle\mathcal{C}_\alpha^-|+c_{+-}^*|\mathcal{C}_\alpha^-\rangle\langle\mathcal{C}_\alpha^+|$$

where the conserved quantities $c_{++}=\mathrm{Tr}(J_{++}^\dagger\rho_0)$, $c_{+-}=\mathrm{Tr}(J_{+-}^\dagger\rho_0)$ are the mean value of the invariant operators J_{++}, J_{+-}

$$J_{++}=\sum_{n=0}^{+\infty}|2n\rangle\langle 2n|$$
$$J_{+-}=\sqrt{\frac{2|\alpha^2|}{\sinh(2|\alpha|^2}}\sum_{q=-\infty}^{+\infty}\frac{(-1)^q}{2q+1}I_q(|\alpha|^2)J_{+-}^{(q)}e^{-i(2q+1)\arg(\alpha)}$$

where I_q is the modified Bessel function of the first kind and

$$J_{+-}^{(q)}=\begin{cases}\frac{(\hat{a}^\dagger\hat{a}-1)!!}{(\hat{a}^\dagger\hat{a}+2q)!!}J_{++}\hat{a}^{2q+1} & q\geq 0\\ J_{++}\hat{a}^{\dagger 2|q|-1}\frac{(\hat{a}^\dagger\hat{a})!!}{(\hat{a}^\dagger\hat{a}+2|q|-1)!!} & q<0.\end{cases}$$

By expanding the Lindblad super-operator in equation (1.3), one can check that it is equivalent to apply a two-photon drive to a QHO that loses photon in pairs

$$\frac{d\rho}{dt}=[\epsilon_2\hat{a}^{\dagger 2}-\epsilon_2^*\hat{a}^2,\rho]+\kappa_2\mathcal{D}[\hat{a}^2]\rho \tag{1.4}$$

where the (tunable) amplitude and phase of the two-photon drive ϵ_2 determine the value of the cat code

$$\alpha=\sqrt{\frac{2\epsilon_2}{\kappa_2}}.$$

1.4.3 Realization of the two photon driven-dissipative scheme

In this subsection, we recall the mechanism that can be used to implement the two-photon driven-dissipative dynamics of equation (1.3). The general recipe to engineer a non-trivial dissipation operator of the form $\sqrt{\kappa}\hat{L}$ on a long-lived memory mode $\hat{a}$ is to use an additional mode described by the annihilation operator $\hat{b}$, usually referred to as the buffer mode, and to couple it to the mode $\hat{a}$ via an interaction Hamiltonian

$$\hat{H}_{\mathrm{int}}=g\hat{L}\hat{b}^\dagger+g^*\hat{L}^\dagger\hat{b}. \tag{1.5}$$

When the buffer mode is very lossy with respect to this interaction Hamiltonian, $|g|\ll\kappa$, the full dynamics describing the evolution of the two modes state

$$\dot{\rho}_{\hat{a},\hat{b}}=-i[\hat{H}_{\mathrm{int}},\rho_{\hat{a},\hat{b}}]+\kappa\mathcal{D}[\hat{b}]\rho_{\hat{a},\hat{b}} \tag{1.6}$$

reduces to an *effective* dissipative dynamics on the memory mode $\hat{a}$ given by

$$\dot{\rho}_{\hat{a}}=\frac{4|g|^2}{\kappa}\mathcal{D}[\hat{L}]\rho_{\hat{a}}. \tag{1.7}$$

The complexity of engineering an exotic dissipation of the form $\mathcal{D}[\hat{L}]$ is thus to engineer the corresponding interaction Hamiltonian (1.5). The interaction Hamiltonian to engineer for the two-photon pumping scheme is

$$\hat{H}_{\text{int}} = g(\hat{a}^2 - \alpha^2)\hat{b}^\dagger + g^*(\hat{a}^2 - \alpha^2)^\dagger \hat{b}.$$

The discussion of the realization of this interaction Hamiltonian in circuit QED is postponed to Chapter 2, where its implementation is discussed along with other Hamiltonians required to perform operations on the cat qubit. We now recall why the dynamics of equation (1.6) gives rise to the effective dissipative dissipation (1.7).

The systematic adiabatic elimination of one of the two subsystems of a bipartite open quantum system, where the subsystem to be eliminated is subject to a strong (fast) dissipative dynamics and is coupled to the (slow) subsystem via a Hamiltonian interaction, has been rigorously exposed in [13]. Using the dimensionless time $t \leftarrow \kappa t$, $\hat{H}_{\text{int}} \leftarrow \hat{H}_{\text{int}}/\kappa$, the full dynamics is expressed in the small perturbation parameter $\varepsilon = |g|/\kappa$

$$\dot{\rho}_{\hat{a},\hat{b}} = \mathcal{L}(\rho_{\hat{a},\hat{b}}) = -i\varepsilon[\hat{H}_{\text{int}}, \rho_{\hat{a},\hat{b}}] + \mathcal{D}[\hat{b}]\rho_{\hat{a},\hat{b}}.$$

In the absence of interaction ($\varepsilon = 0$), an initial separable state $\rho_{\hat{a},\hat{b}}(0) = \rho_{\hat{a}}(0) \otimes \rho_{\hat{b}}(0)$ remains separable at all times under this evolution $\rho_{\hat{a},\hat{b}}(t) = \rho_{\hat{a}}(0) \otimes \rho_{\hat{b}}(t)$ and the $\hat{b}$ mode relaxes rapidly to the vacuum state $\rho_{\hat{a},\hat{b}}(\infty) = \rho_{\hat{a}}(0) \otimes |0\rangle\langle 0|$. The generic adiabatic elimination technique of [13] proposes to seek a solution that is a perturbation in ε of the unperturbed situation, by modelling the effective dynamics the subsystem $\hat{a}$ by a density operator ρ_s on a Hilbert space $\mathcal{H}_s$ of the same dimension as $\mathcal{H}_{\hat{a}}$ with dynamics

$$\dot{\rho}_s = \mathcal{L}_{s,\varepsilon}(\rho_s)$$

and to recover the full evolution of the composite system via the mapping

$$\rho_{\hat{a},\hat{b}} = \mathbb{K}_\varepsilon(\rho_s)$$

where $\mathbb{K}_\varepsilon$ is a completely positive trace preserving map. The idea to compute $\mathcal{L}_{s,\varepsilon}$ and $\mathbb{K}_\varepsilon$ is to expand them in power series of the perturbation parameter ε

$$\mathcal{L}_{s,\varepsilon} = \sum_{k\in\mathbb{N}} \varepsilon^k \mathcal{L}_{s,\varepsilon}^{(k)}$$
$$\mathbb{K}_\varepsilon = \sum_{k\in\mathbb{N}} \varepsilon^k \mathbb{K}_\varepsilon^{(k)}$$

Expressing $\dot{\rho}_{\hat{a},\hat{b}}$ in two different ways yields the following invariance equation

$$\mathcal{L}(\mathbb{K}_\varepsilon(\rho_s)) = \mathbb{K}_\varepsilon(\mathcal{L}_{s,\varepsilon}(\rho_s))$$

and the identification of terms with equal power in ε yields recursive formulas to

compute $\mathcal{L}_{s,\varepsilon}^{(k)}, \mathbb{K}_{\varepsilon}^{(k)}$ at any order. The truncation of the infinite sums $\mathcal{L}_{s,\varepsilon}, \mathbb{K}_{\varepsilon}$ to some power in ε provides an approximation of the dynamics up to this order, but in general it is challenging to prove that the truncated sum indeed takes a Lindblad form and a Kraus map form, respectively. In the particular case of a purely Hamiltonian interaction between the two subsystems, and with a fast dissipative dynamics on the $\hat{b}$ mode $\kappa\mathcal{D}[\hat{b}]$, the reduced dynamics on the mode $\hat{a}$, parametrized by the single mode operator ρ_s, reads [13]

$$\dot{\rho}_s = \mathcal{L}_{s,\varepsilon}(\rho_s) = -i\varepsilon[\hat{H}_s^{(1)}, \rho_s] - i\varepsilon^2[\hat{H}_s^{(2)}, \rho_s] + \varepsilon^2 \sum_{\mu} \mathcal{D}[L_{s,\mu}^{(2)}]\rho_s + \mathcal{O}(\varepsilon^3)$$

i.e the first order contribution in ε is purely Hamiltonian, while the second order contribution in ε contains both Hamiltonian terms and dissipative terms (in general). For the particular Hamiltonian interaction $\hat{H}_{\text{int}} = g\hat{L}\hat{b}^\dagger + g^*\hat{L}^\dagger\hat{b}$, the first order contribution $\hat{H}_s^{(1)}$ is given by

$$\hat{H}_s^{(1)} = \text{Tr}(\hat{b}|0\rangle\langle 0|)\hat{L} + \text{Tr}(\hat{b}^\dagger|0\rangle\langle 0|)\hat{L}^\dagger = 0.$$

and the second order contributions $\hat{H}_s^{(2)}, L_{s,\mu}^{(2)}$ are

$$\hat{H}_s^{(2)} = 0$$
$$L_{s,0}^{(2)} = 2\hat{L}$$

where in this special case, there is only a single non-zero dissipation channel $L_{s,\mu}^{(2)}$, such that the final reduced dynamics on the system mode $\hat{a}$, up the second order in ε, is described by the master equation

$$\dot{\rho}_s = 4\varepsilon^2\mathcal{D}[\hat{L}]\rho_s + \mathcal{O}(\varepsilon^3).$$

We have argued that the effective dissipative dynamics $\mathcal{D}[\hat{a}^2 - \alpha^2]$ stabilizing the cat code words is crucial to get the exponential suppression of the bit-flip errors. This desired dynamics is obtained through the second order effective dynamics resulting from the non-linear coupling of the system to a highly dissipative reservoir. However, together with this ideal dissipation comes an infinite number of (small) higher order terms in ε. Fortunately, the additional dynamics described by these higher order terms is not harmful to the exponential suppression of bit-flip errors. An intuitive way to realize this is to note that the two states $|\pm\alpha\rangle \otimes |0\rangle$ of the full memory and buffer system are eigenstates of the full dynamics with the interaction Hamiltonian $\hat{H}_{\text{int}} = g(\hat{a}^2 - \alpha^2)\hat{b}^\dagger + g^*(\hat{a}^2 - \alpha^2)^\dagger\hat{b}$, such that any higher order term in the expansion can not cause transitions between the two coherent states $|\pm\alpha\rangle$.

1.5 Outline of the manuscript

The entire work of this manuscript is to devoted to answering the following question: can repetition cat qubits be used to perform large-scale quantum computations? We present and analyze the set of operations that can be performed on cat qubits in Chapter 2, with a particular focus on the noise structure of these operations. Indeed, these operations must be "compatible" with the overall scheme construction in the following sense: since the active error correcting code does not protect against bit-flips, these errors need to remain exponentially suppressed while operations are executed, a property of the operations called *bias preserving*. In Chapter 3, we construct a universal set of logical encoded gates for repetition cat qubits from the bias-preserving operations of Chapter 2. The bottleneck of this construction is the realization of a fault-tolerant encoded version of the non-Clifford Toffoli gate, for which we propose two schemes: the first one, that requires code concatenation, is more readily adaptable to a *local* architecture; while the second one achieves a better reduction of the hardware overhead at the cost of a more complicated physical implementation. The thorough numerical analysis of the performance of this approach using full circuit-level analysis of the circuits proposed is presented in Chapter 4. Chapter 5 contains some concluding remarks and perspectives regarding this work.

Chapter 2

Bias-preserving operations on cat qubits

This chapter and the following one cover the work that was published in [60].

2.1 Bias-preserving gates: two conditions to satisfy

The purpose of this Chapter is to describe how the quantum bit of information stored in a cat qubit can be processed. While the original motivation behind the design of cat qubits was to build a well protected quantum *memory* [80], that is, a quantum system in which information can be stored reliably but without any processing capacity, the following work [87] demonstrated that the stored information could be processed *in situ*, thus promoting the cat code to a legitimate qubit. For the sake of completeness, and because some of these gates are needed later in this dissertation, we describe both the operations that were already known prior to this work as well as the new ones. The very design of the cat qubit introduces an asymmetry in the

structure of the noise, which we believe might lead to the experimental observation of larger noise bias than what has been observed in any quantum system so far. Since the entire construction presented in this thesis relies on the assumption that the noise bias can be made extremely large, it is crucial that this remains true even *during* the processing of the information encoded in the cat qubit. Hence, much of the focus of this Chapter is to design operations on the cat qubit that are "compatible" with the noise bias, and such operations are called *bias-preserving*. There are actually two distinct conditions to fulfill to successfully design a bias-preserving operation, that are discussed separately in the next two subsections.

2.1.1 Forbidden operations

The first condition concerns the operation itself, as operators that *convert* the Z operator into an X (or Y) operators are automatically non bias-preserving. The textbook example of such an operation is the Hadamard gate H, that converts a Pauli Z operator into X (and vice-versa). Applying a Hadamard gate to a cat qubit converts a phase-flip error that occurs just before the gate to a bit-flip error. Thus, even though bit-flips occur with an exponentially small probability, the application of a Hadamard gate on a cat qubit re-introduces bit-flips with a probability that is similar to the phase-flip error probability. This observation can be generalized to other gates such as the S gate or the controlled-H gate, etc. The gates that commute with the Z operator are readily acceptable candidates. Gates that do not commute with Z may still be acceptable, as long as the error produced by the propagation of the Z error through the gate remains of the phase-flip type. Note that in this regard, we only require that Z errors are not converted to other types of errors, while X or Y errors that occur with exponentially small probability can be converted to other types of errors.

Single-qubit gates Consider the case of a unitary operator U acting on a single qubit. For the purpose of this discussion, one can disregard the global phase and identify the unitary to a rotation on the Bloch sphere of an angle θ around the axis specified by the real valued unitary vector $\vec{n} = (n_x, n_y, n_z)$

$$U = \mathcal{R}_{\vec{n}}(\theta) = e^{-i\frac{\theta}{2}\vec{n}\cdot\vec{\sigma}} = \cos\frac{\theta}{2} - i\sin\frac{\theta}{2}(n_x X + n_y Y + n_z Z).$$

where $\vec{\sigma} = (X, Y, Z)$. A phase-flip error Z that occurred before the gate $\mathcal{R}_{\vec{n}}(\theta)$ propagates through the gate as a Z error together with an additional unitary error $\mathcal{E}_Z(\vec{n}, \theta)$

$$\mathcal{R}_{\vec{n}}(\theta)Z = Z\mathcal{E}_Z(\vec{n}, \theta)\mathcal{R}_{\vec{n}}(\theta)$$

where the additional error $\mathcal{E}_Z(\vec{n},\theta)$ is given by

$$\mathcal{E}_Z(\vec{n},\theta) = (\cos\theta + 2\sin^2\frac{\theta}{2}n_z^2)I - i(\sin\theta n_x + 2\sin^2\frac{\theta}{2}n_y n_z)X \\ - i(\sin\theta n_y - 2\sin^2\frac{\theta}{2}n_x n_z)Y.$$

Thus, the unitary $\mathcal{R}_{\vec{n}}(\theta)$ does not convert Z errors into X or Y error if and only if the following conditions are satisfied

$$\begin{cases} \sin\theta n_x + 2\sin^2\frac{\theta}{2}n_y n_z = 0 \\ \sin\theta n_y - 2\sin^2\frac{\theta}{2}n_x n_z = 0. \end{cases} \tag{2.1}$$

As one could expect, this includes the rotations around the Z axis of the Bloch sphere of an arbitrary angle $Z(\theta)$, which commute with the Z operator (the π rotation around the Z-axis) as confirmed by checking the above conditions are satisfied for $n_z = 1$ (and hence $n_x = n_y = 0$) and any angle θ. Note that the set of rotations $\{Z(\theta)\}_{\theta\in[0,2\pi[}$ actually contains gates from arbitrarily high levels of the Clifford hierarchy (see Appendix B), as one can check that the rotation around a *cardinal axis* $+X, +Y, +Z$ of an angle $\theta \in \{k\pi/2^{n-1}\}_{k\in\mathbb{Z}_{2^n}}$ is in the n-th level of the Clifford hierarchy.

The conditions 2.1 can also be satisfied for unitaries that do not commute with the Z operator. For instance, they are satisfied by the π-rotations around any axis in the (X,Y) plane, $\theta = \pi$ and $(n_x, n_y, n_z) = \vec{n}_{X,Y}(\varphi) = (\cos(\varphi), \sin(\varphi), 0)$, producing the unitary

$$U = \mathcal{R}_{\vec{n}_{X,Y}(\varphi)}(\pi) = \cos(\varphi)X + \sin(\varphi)Y.$$

Note that this set also contains gates from arbitrary levels of the Clifford hierarchy, following from the fact that $\mathcal{R}_{\vec{n}_{X,Y}(\varphi)}(\pi)$ is in the $(n-1)$-th level if $\varphi \in \{k\pi/2^{n-1}\}_{k\in\mathbb{Z}_{2^n}}$. This can be checked *e.g* from the above fact using the decomposition

$$\mathcal{R}_{\vec{n}_{X,Y}(\varphi)}(\pi) = Z(\varphi)XZ(-\varphi).$$

The conditions (2.1) also automatically rule out some gates. One can check, as expected, that the Hadamard gate ($\theta = \pi$, $\vec{n} = (1,0,1)/\sqrt{2}$) does not match the criteria, or that the only rotations around the X or Y axis that are allowed are those of an angle π, that is the Pauli rotations.

Entangling gates The same analysis can be carried through for entangling gates. For two qubits (or more generally, two subsystems), an entangling gate U is a gate that cannot be factorized in the form $U = U_1 \otimes U_2$, where $U_{1,2}$ are gates acting each on one of the two qubits (or two subsystems).

Here, we investigate the bias-preserving compatibility of a specific subset of entangling gates composed of the "controlled" gates. A two-qubit controlled gate is built using two single-qubit unitary operators (different from the identity) U_1 and

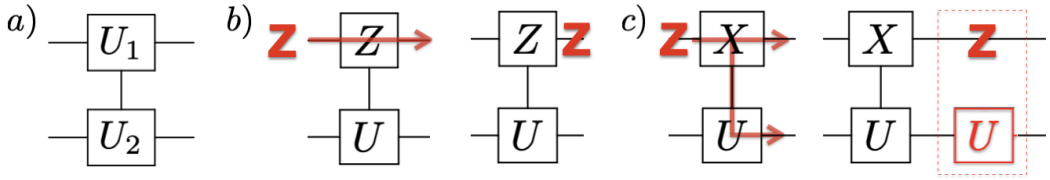


FIGURE 2.1: a) Circuit representation of a general U_1-controlled-U_2 gate. (b-c) A Pauli Z error commutes with a Z-controlled-U operation, but produces an additional U error by propagating through an X-controlled-U gate.

U_2, where one of the two (say U_1) has to be Hermitian. The resulting entangling gate, called the "U_1-controlled$-U_2$" gate, acts as follows. The unitary operator U_1 being Hermitian (and non-trivial), it has exactly two eigenvalues: ± 1. The two associated eigenspaces split the Hilbert space of the first qubit in two. The U_1-controlled-U_2 consists in applying the unitary U_2 to the second qubit, called the *target* qubit, whenever the state of the first qubit, called the *control* qubit, is in the -1 eigenspace, and applying identity otherwise. The circuit representation of such a gate is depicted in Figure 2.1 a). In general, it suffices that only one of the two unitaries U_1 or U_2 be Hermitian to define such an operation, where the qubit corresponding to the Hermitian unitary is taken as the control qubit. Interestingly, when both unitaries are Hermitian, choosing either one of the qubits as the control qubit produces the same quantum gate. Note that in the literature, a Z-controlled-U gate is simply referred to as a controlled-U operation, because the ± 1 eigenstates of the Z operator are the computational $|0\rangle, |1\rangle$ states and the Z control is represented by the symbol $\bullet$ in circuit notation to emphasize the classical analogy. This definition is readily generalized to multi-qubit unitaries.

In this work, we focus on a specific subset of multi-qubit controlled gates where the basic unitaries used to construct entangling gates are only X and Z Pauli operators. This includes, for example, the two-qubit "controlled-NOT" gate (Z-controlled-X), denoted CNOT or CX, the two-qubit "controlled-Z" gate (Z-controlled-Z) gate, denoted CZ, or the three-qubit Toffoli gate (Z-controlled-Z-controlled-X), denoted CCX.

As with the single-qubit unitaries, we are here interested in two things. First, one can check that an n-qubit controlled gate where each of the involved unitaries is a (non-trivial) Pauli operator is in the n-th level of the Clifford hierarchy. Consider first the three two-qubit controlled gates that can be formed using X and Z operators

$$\{U_1 - \text{controlled} - U_2,\ U_{1,2} \in \{X, Z\}\}.$$

We are interested in how Z errors propagate through such gates. As Z trivially commutes with Z but anti-commutes with X, it is clear that the ± 1 eigenspaces of

the Z operator are not disturbed by a Z error, while the ± 1 eigenspaces of X are swapped. Thus, as depicted in Figure 2.1 (b-c), a Z error acting on a "Z-controlled-U" commutes with the gate, while a Z error acting on a "X-controlled-U" produces an additional error U on the corresponding qubit. From this observation, it is clear that a multi-qubit controlled gate built with X and Z operators does not convert Z errors if and only if U is composed of Z operators only. In other words, the eligible gates cannot contain more than a single X control: the CZ gate and the CNOT gate are not forbidden by our bias-preserving definition, while the "X-controlled-X" gate is.

The same analysis carries through straightforwardly to a higher number of qubits. Using only X and Z operators, it is necessary to use at least three qubits to construct a non-Clifford gate (gates that do not belong to the first two levels of the Clifford hierarchy). Out of the four gates of the form

$$\{U_1 - \text{controlled} - U_2 - \text{controlled} - U_3,\ U_{1,2,3} \in \{X, Z\}\}$$

only those that contain zero (the CCZ) or one (the CCX gate) X operator do not convert Z type errors into X type errors.

2.1.2 Bias-preserving implementation

The second condition that needs to be fulfilled by a bias-preserving gate in addition to not convert Z-type errors into X or Y-type errors is that it should be *implementable* in a bias-preserving manner. While the first condition is agnostic to the specific technology implementing the qubit but rather only depends on the structure of the unitary itself, this second condition can only be checked on the description of the process actually implementing the gate on a given physical platform. Let us consider, for instance, rotations around the X (or Y axis). As discussed in the previous subsection, only the π-rotation around the X axis is a viable candidate for a bias-preserving implementation. The authors of [8] rightfully noted that the structure of the noise *induced* by the implementation of the gate may have no reason to be highly biased: in the case of a π-rotation around the X-axis, for instance, the effect of a slight over-rotation or under-rotation may introduce an error proportional to X rather than Z, thus re-introducing bit-flip errors that are not exponentially unlikely.

Fortunately, no gate is lost at this step and we argue in the next subsections that all the candidates introduced above can indeed be implemented in a bias-preserving manner on cat qubits. Because the exponential bias in the noise structure comes from the distance in the phase-space between the two computational states, one general guiding principle that needs to be followed when designing such implementations is that this distance should never be decreased during the process implementing a gate.

This guiding principle is necessary, but sufficient only for the gates that commute with the Z errors *at all times* during the execution of the gates. It has been shown in

[87] how such gates, that include arbitrary rotations around the Z-axis or the CZ gate, can be implemented using a weak Hamiltonian in presence of the strong two-photon dissipative dynamics. The effect of the weak Hamiltonian implementing the gate is to induce a slow evolution in the two-dimensional stable manifold of the cat qubits. The precise bias-preserving implementation of these gates is discussed in subsection 2.2.2.

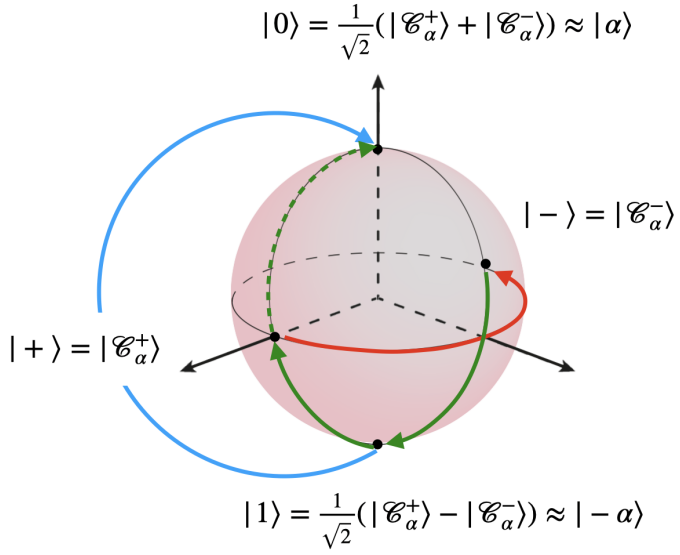


FIGURE 2.2: Schematic illustration of the no-go theorem for a bias-preserving X gate on a two-level system, and of the trick used to work around this no-go in the case of the cat qubit. A continuous evolution mapping the state $|-\alpha\rangle$ to $|\alpha\rangle$ without leaving the code space evolves the state on a path at the surface of the Bloch sphere (green arrows). While doing so, a phase-flip error (here, happening in the middle of the gate and depicted by the red arrow) may be converted to a bit-flip error after the gate is executed. On the other hand, by exploiting the infinite dimensional Hilbert space in which the cat qubit is embedded, a process taking a path outside the code space (blue arrow) can implement a bias-preserving X gate.

The gates that do not commute with the Z error pose additional challenges, and were usually discarded from general hardware agnostic studies of computing with biased noise qubits (see *e.g* [8, 129]). Indeed, considering again the π-rotation around the X-axis of the Bloch sphere, it is actually *impossible* to design a bias-preserving implementation without leaving the code subspace. The rigorous proof of this theorem is detailed in Appendix C for the case of the CNOT gate, but can be readily adapted to the X or Toffoli gates. The intuition of why an X gate cannot be performed in a bias-preserving manner without leaving the code space is illustrated in Figure 2.2, adapted from [81]. A continuous process that rotates the state $|-\alpha\rangle$ to $|\alpha\rangle$ (and vice-versa) without leaving the code space takes the state through a path on the surface of the Bloch sphere (green arrows in Figure 2.2). If a phase-flips occurs, say, at the middle of this evolution (red arrow), then the remaining part of the process (green arrow) results in a bit-flip after the gate is executed. The

way around this no-go theorem is to rather design an implementation that takes the state outside the code space (blue arrow) during the whole evolution, in such a way that the errors that occur during the evolution cannot introduce bit-flip errors. The idea was originally introduced to perform a bias-preserving CNOT gate in the context of Kerr-cat qubits [103]. The gates that are implemented in this manner are discussed in section 2.2.3. The rest of the Chapter is organized as follows.

In section 2.2, we describe the precise implementations of all the operations in the set

$$\mathcal{S}' = \{\mathcal{P}_{|+\rangle}, \mathcal{P}_{|0\rangle}, \mathcal{M}_X, \mathcal{M}_Z\} \cup \{Z(\theta), ZZ(\theta), ZZZ(\theta)\} \cup \{X, \mathrm{CX}, \mathrm{SWAP}, \mathrm{CCX}\}$$

with a particular focus on the bias-preserving property of the implementation. This set is split in three subsets, corresponding to three different ways to achieve a bias-preserving implementation, and discussed separately in the three subsections of section 2.2. The first one (subsection 2.2.1) concerns state preparation and measurement. Here, the bias-preserving property is either trivial or comes from considerations very specific to the realization of the operation. The second subset (subsection 2.2.2) contains the gates that are realized through the quantum Zeno effect, by using a weak Hamiltonian that triggers the accumulation of the desired "dynamical phase" in the cat qubit subspace. Here, the bias-preserving property is ensured by the fact that phase-flips commutes with the continuous process implementing the gate, and by the fact that the two-photon dissipation is always turned on. We note that the operations in these two subsets were already known prior to this work [87]. The last subset (subsection 2.2.3) is composed of the gates that are implemented using a continuous deformation of the code space that impart a topological π phase around the X axis of the Bloch sphere. The bias-preserving implementation of these gates is decomposed in two parts: first, the two-photon dissipative scheme is made time-dependent, and for multi-qubit gates, conditional, in order to implement the required code deformation. Additionally, we argue that the fidelity of these gates is greatly improved by adding a Hamiltonian during the gate execution. We emphasize that these (optional) Hamiltonians are not required for the gate implementation, nor for the bias-preserving property of the gates, but merely to greatly reduce the phase-flip errors induced by the non-adiabaticity (finite time) of these gates.

Then, in section 2.3, we either give or derive explicitly analytical error models for the dominant phase-flip error probabilities, As will become clear upon inspection of the error models, the phase-flip errors occurring during the execution of the gates come from two different sources that we both take into account. The first are the phase-flip errors induced by the main error channel of the QHO, namely the photon loss, characterized by the single photon dissipation rate κ_1. The second source of

phase-flip errors is the finite gate time of the gates. The analytical error models the phase-flip error probability are useful both to better understand the source of errors of the gates and for the numerical study of the performance of the logical circuits presented in Chapter 4.

Last, in section 2.4, we discuss how all the proposed implementations can be realized within the framework of circuit QED. We begin by describing how the two-photon pumping scheme that stabilizes the cat qubit is implemented, as has already been demonstrated experimentally in [78, 118, 83]. Then, we proceed to describe how the weak Hamiltonians required for the Zeno gates have been realized [118] or could be realized. Last, we discuss the realization of the topological gates. This can be split in two parts: the (required) implementation of the time dependence of the two-pumping scheme that realized the topological deformation of the cat qubit code space, and the (optional) implementation of the feed-forward Hamiltonians that might be added during the gates execution to reduce the phase-flip errors induced by non-adiabaticity.

2.2 Bias-preserving implementations

2.2.1 State preparation and measurement

2.2.1.1 State measurement

Measurement of the X operator The only measurement on the cat qubit required in the construction of the scheme is the measurement of the X operator, whose eigenstates are the cat states $|\mathcal{C}_\alpha^\pm\rangle$. Because these states have a well-defined photon-number parity, the measurement of X can be realized by a photon-number parity measurement. Here, the "bias-preserving" property of this operation is trivially ensured by the fact that, every time an X measurement is needed in our circuit, it is performed on an ancilla qubit whose state is discarded after the measurement and prepared again in a fresh state. The measurement of the X operator could be either destructive or quantum non-demolition (QND) as it is only used on ancilla cat qubits, which are discarded after the measurement, except at the very end of the execution of the quantum algorithm where the data cat qubits are also measured (destructively) to get the output of the algorithm. The QND parity measurement proposed in [85] and realized in [15, 114] is perfectly suitable for our scheme. The main idea behind this protocol is to couple to an ancilla (transmon) qubit to the mode whose photon-number parity is to be measured via the dispersive interaction Hamiltonian

$$\hat{H} = -\chi |e\rangle\langle e| \hat{a}^\dagger \hat{a}.$$

The unitary evolution generated by this Hamiltonian on a time interval $T = \pi/\chi$ is given by

$$\hat{\mathcal{U}} = |g\rangle\langle g| I + |e\rangle\langle e| e^{i\pi \hat{a}^\dagger \hat{a}}$$

entangling the state of the ancilla with the parity of the state of the cavity. Preparing the ancilla qubit in a superposition state $|+\rangle = \frac{1}{\sqrt{2}}(|g\rangle + |e\rangle)$, the effect of the unitary $\hat{\mathcal{U}}$ is to flip the ancilla to the state $|-\rangle = \frac{1}{\sqrt{2}}(|g\rangle - |e\rangle)$ when the cavity contains an odd number of photons and to leave it unchanged otherwise. A measurement of the $\hat{\sigma}_x$ operator of the qubit thus reveals the parity of the cavity state.

Note that in order to perform such a parity measurement, the two-photon driven dissipation on the measured cat qubit has to be turned off. However, given that these measurements are performed on ancilla cat qubits that are thrown out after each measurement, the absence of protection during the measurement merely affects the measurement fidelity and does not have any consequence on the rest of the circuit. Fidelities of photon-number parity measurement of about 98.5% have been previously achieved using this protocol [96].

Measurement of the Z operator The measurement of the Pauli Z operator of the cat qubits is not required in the rest of the scheme. Yet, it might be required some day to design new logical operations, or to simplify some of the logical circuits. An example of this is discussed in the considerations about a 2D implementation of our scheme in Chapter 5, where the measurement of the logical Z_L operator can be used to teleport a logical CZ_L gate. Note that the eigenstates of the Z operator are (exponentially close to) the coherent states $|\pm\alpha\rangle$, such that a destructive measurement of the Z operator can be implemented *e.g* the protocol used to measure the phase of a coherent state of [57].

2.2.1.2 State preparation

Preparation of the cat states $|\mathcal{C}_\alpha^\pm\rangle$ The preparation of the eigenstates of the X operator is trivially compatible with the noise bias since a bit-flip does not affect these states, as noted in [8]. Indeed, because the cat states $|\mathcal{C}_\alpha^\pm\rangle$ have equal population on the $|\pm\alpha\rangle$ states, the bit-flip operator X cannot modify these population. One way to prepare the even cat state $|+\rangle = |\mathcal{C}_\alpha^+\rangle$ is performed by initializing the QHO in the vacuum state $|0\rangle$ and turning on the driven two-photon dissipation [87]. Indeed, the two-photon driven dissipation conserves the photon-number parity, such that unique steady state of the system is given by the even cat state. Such a state preparation has already been realized experimentally [78] and the fidelity of this operation is set by the ratio between the two-photon dissipation rate κ_2, setting the rate of convergence to the cat state, and the undesired single-photon loss rate κ_1, setting the parity jump rates (equivalent to phase-flip errors) mixing the even cat with the odd one. Then, the odd cat state $|-\rangle = |\mathcal{C}_\alpha^-\rangle$ can be prepared from the even cat state by applying the Z described in subsection 2.2.2. The preparation of the cat states can also be performed using an active protocol rather than relying on the passive two-photon dissipation. Such protocol, like the mapping of an arbitrary state of a transmon to a cat qubit [79], have the advantage to be faster and to produce states with higher fidelity. A fast and reliable operation that prepares a given state on the

cat qubit is immediately followed by the activation of the stabilization scheme. In particular, in the experiment [118], the state $|\mathcal{C}_\alpha^+\rangle$ was generated using optimal control techniques which can significantly improve the fidelity with respect to a passive preparation with two-photon driven dissipation.

It will become clear in Chapter 3 that the general repetition cat qubit construction only requires that the physical cat qubits used to process the information can be initialized in the $|\mathcal{C}_\alpha^\pm\rangle$ states, as the distance d repetition code protecting against phase-flips has logical code words (in the dual basis) $|\pm\rangle_L = |\mathcal{C}_\alpha^\pm\rangle^{\otimes d}$. However, the preparation of a cat qubit in the coherent state $|0\rangle \approx |\alpha\rangle$ will prove useful to prepare the logical ancilla cat qubit state $|0\rangle_L$ (see Chapter 3) for Steane error correction, so we introduce this preparation now.

Preparation of the coherent state $|\alpha\rangle$ The eigenstates of the Z operator of the cat qubit are exponentially close to the coherent states $|\pm\alpha\rangle$. A fast and reliable preparation of these states is realized by applying a strong microwave pulse to the oscillator initialized in the vacuum state to generate a displacement $\mathcal{D}(\pm\alpha)$, and to turn on the two-photon driven-dissipative stabilization immediately after the displacement. Note that unlike the cat states $|\mathcal{C}_\alpha^\pm\rangle$, the $|\pm\alpha\rangle$ are not intrinsically robust to bit-flip errors, as the bit-flip error operator induces population transfer between $|\alpha\rangle$ and $|-\alpha\rangle$.

Here, the preparation of the state $|\alpha\rangle$ is only bias-preserving in the sense that the phase of the microwave pulse applied to displace the oscillator state from the vacuum to the coherent state $|\pm\alpha\rangle$ can be made very precise, such that the state of the oscillator after this displacement is in a certain coherent state $|\tilde{\alpha}\rangle$ in the neighbourhood of $|\alpha\rangle$, where $|\tilde{\alpha}\rangle$ is in general slightly different from $|\alpha\rangle$ to account for the small imprecision in the displacement. Then, the two-photon pumping is activated just after the displacement, such that the state $|\tilde{\alpha}\rangle$ relaxes to the coherent state $|\alpha\rangle$. The resulting bit-flip probability (*i.e*, the probability to be in the state $|-\alpha\rangle$ after a displacement $\mathcal{D}(\alpha)$ has been applied to the vacuum) can be very small. Indeed, the population of the state $|-\alpha\rangle$ at the end of this protocol is (roughly) given by the probability that a phase error of at least π has occurred in the displacement, which can be sufficiently small. However, because here the "bias-preserving" is ensured solely by the fact that the phase of microwave pulses is very well controlled, it is specific to our circuit QED implementation of the scheme and it is important to check that the probability of a bit-flip error occurring during this protocol is of the same order as the exponentially suppressed bit-flip error of the cat qubit.

2.2.2 Dynamical phase gates with the Quantum Zeno Effect

2.2.2.1 $Z(\theta)$ gate

The $Z(\theta)$ gate is the rotation of an arbitrary angle θ around the Z axis of the Bloch sphere of the cat qubit:

$$Z(\theta) = e^{-i\frac{\theta}{2}Z_\alpha} = \cos\frac{\theta}{2}I_\alpha - i\sin\frac{\theta}{2}Z_\alpha. \tag{2.2}$$

It was first proposed in [87] and realized experimentally in [118]. The subscript α in the Pauli operators I_α and Z_α are here to emphasize that these are operators the Pauli operators acting on the cat qubit. They can be expressed as

$$I_\alpha = |\mathcal{C}_\alpha^+\rangle\langle\mathcal{C}_\alpha^+| + |\mathcal{C}_\alpha^-\rangle\langle\mathcal{C}_\alpha^-|$$
$$Z_\alpha = |\mathcal{C}_\alpha^+\rangle\langle\mathcal{C}_\alpha^-| + |\mathcal{C}_\alpha^-\rangle\langle\mathcal{C}_\alpha^+|$$

In the rest of this manuscript, the subscript α is dropped and the operators acting on the cat qubit are simply written using the usual qubit notations. The $Z(\theta)$ gate is realized by applying a weak resonant drive described (in the rotating frame of the cavity mode) by the Hamiltonian $\hat{H} = \epsilon_Z\hat{a} + \epsilon_Z^*\hat{a}^\dagger$ in the presence of the two-photon driven dissipation modelled by the Lindblad super-operator $\kappa_2\mathcal{D}[\hat{a}^2 - \alpha^2]$, with $|\epsilon_Z|$ small with respect to κ_2. The combination of these two dynamics implements the gate as follows. The fast two-photon driven-dissipative part of the dynamics confines the state in the cat qubit manifold, while the single photon drive induces a change of the photon number parity. If the cat qubit is initialized in the state $|\mathcal{C}_\alpha^+\rangle$, of even photon number parity, the effect of the weak drive is to induce Rabi oscillations between the even cat $|\mathcal{C}_\alpha^+\rangle$ and the odd cat $|\mathcal{C}_\alpha^-\rangle$. The rate of these Rabi oscillations is set by the first order perturbation induced by the Hamiltonian, given by the projection of the Hamiltonian on the cat qubit subspace

$$(|\mathcal{C}_\alpha^+\rangle\langle\mathcal{C}_\alpha^+| + |\mathcal{C}_\alpha^-\rangle\langle\mathcal{C}_\alpha^-|)(\epsilon_Z\hat{a} + \epsilon_Z^*\hat{a}^\dagger)(|\mathcal{C}_\alpha^+\rangle\langle\mathcal{C}_\alpha^+| + |\mathcal{C}_\alpha^-\rangle\langle\mathcal{C}_\alpha^-|) = 2\mathcal{R}[\alpha\epsilon_Z]Z + \mathcal{O}(e^{-2|\alpha|^2}).$$

The fact that the effective dynamics, up to the first order in the small parameter ϵ_Z/κ_2, is given by the projection of the perturbative Hamiltonian is the well known quantum Zeno effect. A rigorous mathematical derivation proving this fact can be found *e.g* in [12].

The oscillation rate is maximized when the phase of the drive is opposite to the phase of α such that $\alpha\epsilon_Z$ is a real number, and the rotation of an angle θ is obtained by applying the weak drive during a time

$$T = \frac{\theta}{4\alpha\epsilon_Z} = \frac{\theta}{4\sqrt{\bar{n}}|\epsilon_Z|}.$$

2.2.2.2 $ZZ(\theta)$ gate and CZ gate

The same recipe can be readily applied to construct the two qubit entangling gate [87]

$$Z_1Z_2(\theta) = e^{-i\frac{\theta}{2}Z_1Z_2} = \cos\frac{\theta}{2}I_1I_2 - i\sin\frac{\theta}{2}Z_1Z_2$$

where the subscript (1,2) label the two cat qubits. This gate is realized by applying a weak beam-splitter Hamiltonian

$$\hat{H} = \epsilon_{Z_1Z_2}\hat{a}_1\hat{a}_2^\dagger + \epsilon_{Z_1Z_2}^*\hat{a}_1^\dagger\hat{a}_2$$

in the presence of the two-photon driven dissipation on both of the cat qubits. Here and for all the multi-qubit gates involved in this work, we always assume that the same α is used for the two (or more) cat qubits. This assumption is made for the sole purpose of reducing the number of notations, but this assumption can be relaxed everywhere is this work and all the gates presented can be straightforwardly adapted to cat qubits of different sizes α and β. Taking $\epsilon_{Z_1Z_2}$ to be real, the projection of $\hat{H}$ on the two cat qubit subspaces gives the oscillation rate $\Omega_{Z_1Z_2} = 2|\alpha|^2\epsilon_{Z_1Z_2}$ such that the rotation $Z_1Z_2(\theta)$ is obtained upon the application of the weak Hamiltonian during a time

$$T = \frac{\theta}{4|\alpha|^2\epsilon_{Z_1Z_2}} = \frac{\theta}{4\bar{n}\epsilon_{Z_1Z_2}}.$$

The two gates $Z(\theta)$ and $Z_1Z_2(\theta)$ commute, such that one can be combine them. For instance, noting that a controlled-Z gate can be decomposed as

$$CZ = (-1)^{|11\rangle\langle 11|} = e^{-i\frac{\pi}{4}(I_1-Z_1)(I_2-Z_2)}$$

and taking α real, the CZ gate is implemented through the Zeno effect by applying the Hamiltonian

$$\hat{H} = \epsilon_{CZ}[-(\hat{a}+\hat{a}^\dagger+\hat{b}+\hat{b}^\dagger) + \frac{1}{\sqrt{\bar{n}}}(\hat{a}\hat{b}^\dagger + \hat{a}^\dagger\hat{b})]$$

for a time $T = \pi/(8\sqrt{\bar{n}}\epsilon_{CZ})$.

2.2.2.3 $ZZZ(\theta)$ gate

From a theoretical point of view, the above Zeno mechanism can be generalized to construct arbitrary rotations on n qubits. However, the weak Hamiltonian required is of higher with each added qubit, which makes it increasingly hard to implement, for reasons that we detail in the experimental section 2.4. The same trade-off is encountered in the case of the topological gates introduced next.

For instance, the three qubit entangling gate $ZZZ(\theta)$ (which, combined with CZ and Z gates, can be used to implement *e.g* the CCZ gate) can be generated *e.g* using

the weak Hamiltonian between the three cat qubits $\hat{a}_{1,2,3}$

$$\hat{H} = \epsilon_{Z_1Z_2Z_3}\hat{a}_1\hat{a}_2\hat{a}_3^\dagger + \text{h.c}$$

for a time $T = \dfrac{\theta}{8|\alpha|^3\epsilon_{Z_1Z_2Z_3}}$.

2.2.3 Topological phase gates with adiabatic code deformation

2.2.3.1 X gate

Dissipative implementation As we have argued in section 2.1, the only rotation around the X-axis that does not convert Z errors into X or Y errors is the rotation of an angle π, the Pauli X gate. The problem of the bias-preserving implementation can be roughly stated as such. How can we design a process, having a unitary action on the two-dimensional subspace of the cat qubits, that implements a rotation of the coherent state $|\alpha\rangle$ to the coherent state $|-\alpha\rangle$ (and vice-versa) while ensuring that the physical errors of the QHO (*e. g* photon loss) result in at most an exponentially small population remaining on the state $|\alpha\rangle$ after the transfer is done? As we have seen in section 2.1, there is no process that can realize this while keeping the state of the system inside the two-dimensional cat qubits manifold during the gate execution. Rather, this is realized using an excursion outside the code space, that can be thought of as a continuous adiabatic deformation of the code space, obtained by varying the complex number α of the two-photon dissipation $\kappa_2\mathcal{D}[\hat{a}^2 - \alpha^2]$ in time. When the variations of $\alpha(t)$ are sufficiently slow with respect to κ_2^{-1}, the driven-dissipative dynamics modelled by the super-operator $\kappa_2\mathcal{D}[\hat{a}^2 - \alpha(t)^2]$ stabilizes the two-dimensional manifold spanned by the coherent states $|\alpha(t)\rangle$ and $|-\alpha(t)\rangle$ at all times t, realizing a slow motion of the fixed points of the dynamics in the phase-space.

Remarkably, the quantum information is preserved while the code space is deformed, provided the two states $|\alpha(t)\rangle, |-\alpha(t)\rangle$ remain sufficiently separated in phase-space at all times. This point is crucial in order to implement a *unitary* operation within the cat qubit manifold. The state $|\psi_0\rangle = c_0|\alpha\rangle + c_1|-\alpha\rangle$ at time $t = 0$ evolves under the effect of $\kappa_2\mathcal{D}[\hat{a}^2 - \alpha(t)^2]$, with $\alpha(0) = \alpha$, to

$$|\psi(t)\rangle = c_0|\alpha(t)\rangle + c_1|-\alpha(t)\rangle$$

provided that at all times intermediate times $t' \in [0, t]$, the two following conditions are satisfied

$$|\dot{\alpha}(t')|/|\alpha(t')| \ll \kappa_2$$
$$|\langle\alpha(t')|-\alpha(t')\rangle|^2 \ll 1.$$

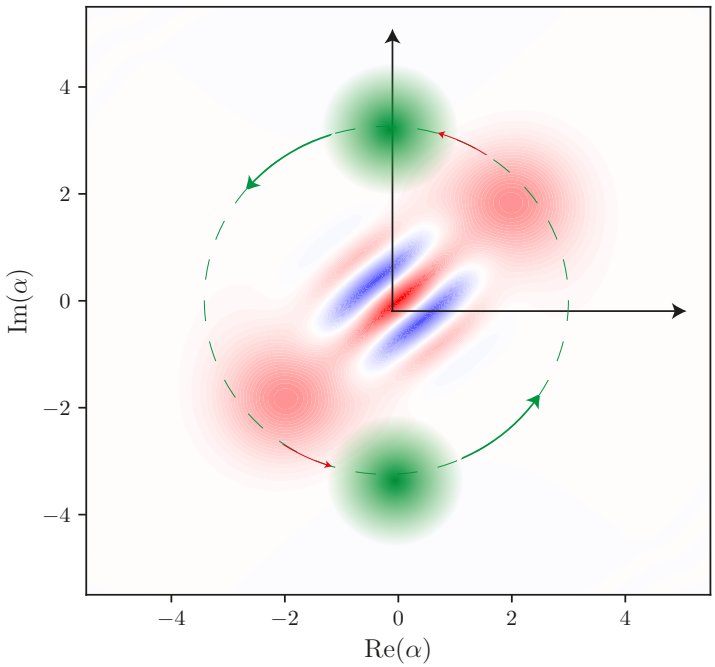


FIGURE 2.3: Wigner function of the state of a cat qubit during the execution of an X operation. The green dots are the Wigner functions of the instantaneous steady states of the dynamics $\dot{\rho} = \kappa_2 \mathcal{D}[\hat{a}^2 - \alpha(t)^2]$. These attracting points are slowly rotated from $\pm\alpha$ to $\mp\alpha$ on the dashed circle, as shown by the green arrows. When this rotation is performed slowly, the cat follows the attractive points (red arrows).

The X gate is realized by choosing a "path" function $\alpha(t)$ such that $|\alpha\rangle$ and $|-\alpha\rangle$ are swapped, *e.g* $\alpha(t) = \alpha e^{i\pi t/T}$, $t \in [0, T]$ where $T \gg \kappa_2^{-1}$ is the gate time. Indeed, the swap $|\alpha\rangle \leftrightarrow |-\alpha\rangle$ corresponds to the map $|\mathcal{C}_\alpha^+\rangle \rightarrow |\mathcal{C}_\alpha^+\rangle$ and $|\mathcal{C}_\alpha^-\rangle \rightarrow -|\mathcal{C}_\alpha^-\rangle$ which is an X operation for the cat qubit.

The process implementing the X gate thus swaps the two states $|\pm\alpha\rangle$ while keeping the quantum information encoded in the superposition of these states intact. With this regard, the gate consists in imparting a topological π-phase to the coherent states. We call this phase "topological" because the state of the system during the execution of the gate does no longer belong to the cat qubit subspace. It is only at the end of the gate that the state is brought back into this manifold together with an exact π-phase. This phase is not affected by the imprecision in the rotation angle. Indeed, the phase of the coherent states $|\pm\alpha\rangle$ are locked to the phase of the pump drives. In this sense, each cat qubit is defined with respect to its own pumps. Therefore, even if the rotation angle is not precisely π, which could happen *e.g* because of the amplitude and phase fluctuations of the pumping drive, the state has still accumulated a topological π-phase with respect to its local oscillator. This is to be contrasted with the accumulation of the dynamical phase realized *inside* the code manifold for the Zeno type gates of the previous section.

Note that in addition to this topological π phase, there is a geometric phase accumulated due to the particular path taken by $\alpha(t)$. However, this phase is the same for the two states $|\pm\alpha\rangle$ and correspond to a physically meaningless global phase.

In the ideal case of a loss-less harmonic oscillator and in the limit where the gate time $T = +\infty$, the fidelity of this operation with respect to the X operator is 1. This operation is bias-preserving as the errors caused by the finite gate time are only of the phase-flip type, but the bit-flips remain exponentially suppressed in the size of the cat $\bar{n}$. Intuitively, this is possible because the two-photon pumping is never turned off during the gate execution. A schematic representation of this evolution in the phase-space is depicted in Figure 2.3.

Optional addition of a feed-forward Hamiltonian To reduce the phase-flip error rate due to the finite gate time, called the non-adiabatic errors, the feed-forward Hamiltonian

$$\hat{H} = -\frac{\pi}{T}\hat{a}^\dagger\hat{a}$$

is turned on while the pumping is being rotated. This Hamiltonian generates the unitary $\hat{\mathcal{R}}(t) = e^{i\frac{\pi}{T}\hat{a}^\dagger\hat{a}t}$ which rotates deterministically the qubit state

$$\hat{\mathcal{R}}(t)|\psi_0\rangle = c_0|\alpha(t)\rangle + c_1|-\alpha(t)\rangle$$

so that it remains at all times in the kernel of the time dependent dissipative channel:

$$[\hat{a}^2 - \alpha(t)^2]\hat{\mathcal{R}}(t)|\psi_0\rangle = 0$$

In presence of this Hamiltonian, there is no need to proceed adiabatically, that is the gate time T can be arbitrarily short.

2.2.3.2 CNOT gate

Dissipative implementation The idea behind the X gate can be adapted to realize a controlled-X (CNOT) gate between two cat qubits, which consists in applying an X gate to the "target" cat qubit when the "control" qubit is in the computational $|1\rangle \approx |-\alpha\rangle$ state and applying the identity otherwise

$$\text{CNOT} = \tfrac{1}{2}(I_1 + Z_1) \otimes I_2 + \tfrac{1}{2}(I_1 - Z_1) \otimes X_2.$$

In terms of operators acting on the cat qubit, the CNOT gate can be written

$$\text{CNOT} \approx |\alpha\rangle\langle\alpha| \otimes (|\alpha\rangle\langle\alpha| + |-\alpha\rangle\langle-\alpha|) + |-\alpha\rangle\langle-\alpha| \otimes (|\alpha\rangle\langle-\alpha| + |-\alpha\rangle\langle\alpha|)$$

The approximation is exponentially precise in $|\alpha|^2$. This operation is realized by making a rotation of the pumping of the target qubit implementing the X gate that depends on the state of the control qubit, by modifying the dissipation channels of the cat qubits $\mathcal{L}_{\hat{a}} = \mathcal{D}[\hat{L}_{\hat{a}}]$ and $\mathcal{L}_{\hat{b}} = \mathcal{D}[\hat{L}_{\hat{b}}(t)]$, with:

$$\begin{aligned}\hat{L}_{\hat{a}} &= \hat{a}^2 - \alpha^2 \\ \hat{L}_{\hat{b}}(t) &= \hat{b}^2 - \tfrac{1}{2}\alpha(\hat{a} + \alpha) + \tfrac{1}{2}\alpha e^{2i\frac{\pi}{T}t}(\hat{a} - \alpha)\end{aligned}$$

where we denote by $\hat{a}$ (resp. $\hat{b}$) the mode of the control cat qubit (resp. target cat qubit). The dissipation channel on the control qubit $\mathcal{L}_{\hat{a}}$ is the two-photon pumping scheme stabilizing the control cat qubit. The second dissipation channel, however, acts on the target cat qubit but also depends on the first mode $\hat{a}$. It should be understood as follows: when the control qubit $\hat{a}$ is in the state $|\alpha\rangle$, the operator $\hat{L}_{\hat{b}}(t)$ acts on the target mode as $\hat{b}^2 - \alpha^2$, stabilizing the idle code space, but when the control qubit is in the state $|-\alpha\rangle$, the pumping becomes $\hat{b}^2 - (\alpha e^{i\frac{\pi}{T}t})^2$, thus implementing the time-dependent two-photon pumping dissipation used for the X gate. Just like for the X gate, the pumping is always turned on during the gate and the bit-flip errors remain exponentially suppressed throughout the gate, ensuring that the CNOT gate preserves the biased structure of the noise.

In the case of the X gate, the geometric phase corresponded to a physically meaning-less global phase, but here this phase is conditioned on the state of the control qubit. As a consequence, the geometric phase induces a deterministic rotation around the Z-axis of the control qubit. The rotation angle is given by

$$\vartheta = -i\int_0^T \langle \pm\alpha(t)|\frac{d}{dt}|\pm\alpha(t)\rangle dt = \pi|\alpha|^2.$$

This deterministic geometric phase can be removed by applying the appropriate $Z(\theta)$ operation discussed above. Another option is to ensure the rotation angle ϑ is a multiple of 2π, either by setting the number of photons to be an even integer or by choosing a path $\alpha(t)$ such that the result of the integral is a multiple of 2π. Even in this case, the fluctuations along the chosen path will inevitably lead to a certain imprecision in the final value of the geometric phase, leading to additional phase-flip errors.

Optional addition of a feed-forward Hamiltonian A major part of the phase-flip errors induced by non-adiabatic effects can be compensated in the same way as for the X gate, by adding a Hamiltonian evolution of the form

$$\hat{H} = \frac{1}{2}\frac{\pi}{T}\frac{\hat{a} - \alpha}{2\alpha} \otimes (\hat{b}^\dagger\hat{b} - \bar{n}) + h.c.$$

while rotating the pumping. In presence of two-photon pumping, this Hamiltonian is an approximation of the "ideal" Hamiltonian that would perfectly cancel all of the

non-adiabatic errors

$$\hat{H}^* = -\frac{\pi}{T}|-\alpha\rangle\langle-\alpha| \otimes (\hat{b}^\dagger\hat{b} - \bar{n})$$

which triggers a rotation of the target cat qubit in the phase-space conditional to the control cat qubit being in the state $|-\alpha\rangle$. Similar Hamiltonians have been already realized using parametric methods [119], (see section 2.4).

2.2.3.3 Toffoli gate

Dissipative implementation The Toffoli gate is the three-qubit controlled-controlled-X gate (CCX)

$$\begin{aligned}\text{Toffoli} = \tfrac{1}{4}(I_1 + Z_1)(I_2 + Z_2)I_3 + \tfrac{1}{4}(I_1 + Z_1)(I_2 - Z_2)I_3 \\ + \tfrac{1}{4}(I_1 - Z_1)(I_2 + Z_2)I_3 + \tfrac{1}{4}(I_1 - Z_1)(I_2 - Z_2)X_3.\end{aligned}$$

This unitary is in the third level of the Clifford hierarchy, thus it does not belong to the Clifford group. In many of the schemes achieving universality, the non-Clifford operations are the most difficult operations to implement. While the Toffoli gate is undeniably the most complicated gate of the physical gate set, its implementation is similar to the two-qubit CNOT gate.

Similarly to the CNOT gate, only the dissipation channel of the target cat qubit needs to be modified, $\mathcal{L}_{\hat{a}} = \mathcal{D}[\hat{L}_{\hat{a}}]$, $\mathcal{L}_{\hat{b}} = \mathcal{D}[\hat{L}_{\hat{b}}]$ and $\mathcal{L}_{\hat{c}} = \mathcal{D}[\hat{L}_{\hat{c}}(t)]$,

$$\begin{aligned}\hat{L}_{\hat{a}} &= \hat{a}^2 - \alpha^2 \\ \hat{L}_{\hat{b}} &= \hat{b}^2 - \alpha^2 \\ \hat{L}_{\hat{c}}(t) &= \mathcal{D}[\hat{c}^2 - \tfrac{1}{4}(\hat{a}+\alpha)(\hat{b}+\alpha) + \tfrac{1}{4}(\hat{a}+\alpha)(\hat{b}-\alpha) \\ &\quad + \tfrac{1}{4}(\hat{a}-\alpha)(\hat{b}+\alpha) - \tfrac{1}{4}e^{2i\frac{\pi}{T}t}(\hat{a}-\alpha)(\hat{b}-\alpha)].\end{aligned}$$

Here, $\mathcal{L}_{\hat{a}}$ and $\mathcal{L}_{\hat{b}}$ keep stabilizing the two control modes $\hat{a}$ and $\hat{b}$ in manifolds spanned by $|\pm\alpha\rangle$, and $\mathcal{L}_{\hat{c}}$ rotates the two-photon pumping on the target mode $\hat{c}$ only when the control cat qubits are in the state $|-\alpha,-\alpha\rangle$.

In theory, assuming the required couplings between any number of modes are available, the mechanism behind the topological X, CNOT and Toffoli gates can be straightforwardly adapted to implement the n-qubit entangling gate $C^{n-1}X$ belonging to the n-th level of the Clifford hierarchy, where C^{n-1} denotes the controls on the first $n-1$ qubits. Note that in practice, the implementation of the required dissipative channels would involve non-linear processes of higher order which are much more complex to realize and that would typically be weak.

As for the CNOT gate, the deterministic geometric phase associated to the path taken by the target cat qubit can also be eliminated by tailoring the path followed in the phase-space by the cat states during the execution of the gate, or by physically applying $Z(\theta)$ and $ZZ(\theta)$ gates.

Optional addition of a feed-forward Hamiltonian Similarly to all the topological gates implemented on the cat qubits involving a continuous evolution of the code space, the fidelity of the dissipative implementation can be improved by adding a feed-forward Hamiltonian whose role is merely to reduce the phase-flip errors induced by non adiabaticity. Again, the systematic construction of such a Hamiltonian is based on the adaptation of the ideal feed-forward Hamiltonian for the X gate to the particular case where this rotation is realized conditionally on some control cat qubits state. In the specific case of the Toffoli gate, the target cat qubit (mode $\hat{c}$) undergoes a rotation in the phase-space implementing the X gate only when the joint state of the two control cat qubits is $|-\alpha,-\alpha\rangle$. Thus, in analogy with the X gate, a "perfect" feed-forward Hamiltonian that would exactly cancel the non-adiabatic phase-flip errors is

$$\hat{H} = -\frac{\pi}{T}|-\alpha\rangle\langle-\alpha| \otimes |-\alpha\rangle\langle-\alpha| \otimes \hat{c}^\dagger\hat{c}.$$

Just like for the Toffoli gate, the projectors on coherent states are not Hamiltonians that can be implemented; but they can be well approximated by (where the approximation on the cat manifold is exponentially good in α)

$$|-\alpha\rangle\langle-\alpha| \approx \frac{\alpha-\hat{a}}{2\alpha}$$

and the resulting approximate Hamiltonian that we propose to apply while a Toffoli gate is performed to remove most of the non-adiabatic phase-flip errors is thus given by

$$\hat{H} = -\frac{1}{2}\frac{\pi}{T}\frac{\hat{a}-\alpha}{2\alpha} \otimes \frac{\hat{b}-\alpha}{2\alpha} \otimes (\hat{c}^\dagger\hat{c} - \bar{n}) + h.c.$$

2.2.3.4 SWAP gate

Dissipative implementation The two-qubit SWAP gate, which acts like its name suggests, is trivially compatible with a bias-preserving implementation as it does not convert Z-type errors into X- or Y-type errors. As it will become clear in the next Chapter, the SWAP gate is often useful to adapt logical circuits to actual constraints on the connectivity graph of the physical qubits. Noting that a SWAP gate can be implemented using three CNOT gates establishes that the SWAP gate can be implemented in a bias-preserving manner, but there is a more direct way to do this. Following the guiding principle of the CNOT gate, the SWAP gate is realized by replacing the regular two-photon dissipation operators $\hat{L}_{\hat{a}} = \hat{a}^2 - \alpha^2$ and $\hat{L}_{\hat{b}} = \hat{b}^2 - \alpha^2$ by the following time-dependent operators that combine both modes

$$\begin{aligned}
\hat{L}_{\hat{a}}(t) &= \hat{a}^2 - \tfrac{1}{2}\hat{a}\hat{b}(1 - e^{2i\frac{\pi}{T}t}) - \tfrac{1}{2}\alpha^2(1 + e^{2i\frac{\pi}{T}t}),\\
\hat{L}_{\hat{b}}(t) &= \hat{b}^2 - \tfrac{1}{2}\hat{a}\hat{b}(1 - e^{-2i\frac{\pi}{T}t}) - \tfrac{1}{2}\alpha^2(1 + e^{-2i\frac{\pi}{T}t}).
\end{aligned}$$

for $t \in [0, T]$ where T is the SWAP gate time. The instantaneous joint kernel of these operators is the four dimensional Hilbert space spanned by the set of coherent states

$$\{|\alpha, \alpha\rangle, |-\alpha, -\alpha\rangle, |\alpha e^{i\frac{\pi}{T}t}, -\alpha e^{-i\frac{\pi}{T}t}\rangle, |-\alpha e^{i\frac{\pi}{T}t}, \alpha e^{-i\frac{\pi}{T}t}\rangle\}.$$

Recalling that $|0\rangle \approx |\alpha\rangle$ and $|1\rangle \approx |-\alpha\rangle$, these two dissipation channels implement the correct mapping corresponding to a SWAP gate:

$$\begin{aligned} |\alpha, \alpha\rangle &\rightarrow |\alpha, \alpha\rangle \\ |-\alpha, -\alpha\rangle &\rightarrow |-\alpha, -\alpha\rangle \\ |\alpha, -\alpha\rangle &\rightarrow |-\alpha, \alpha\rangle \\ |-\alpha, \alpha\rangle &\rightarrow |\alpha, -\alpha\rangle. \end{aligned}$$

Optional addition of a feed-forward Hamiltonian Similarly to the others gates that are implemented using a rotation of the steady states of the driven-dissipative super-operators in the phase space, the phase-flip errors of a SWAP gate caused by non-adiabaticity are reduced when the Hamiltonian

$$\hat{H} = -\frac{\pi}{4\alpha^2 T}(\hat{a}^\dagger\hat{a} - \hat{b}^\dagger\hat{b})(\alpha^2 - \hat{a}\hat{b}) + \text{h.c}$$

is added during the gate. The operator $(\alpha^2 - \hat{a}\hat{b})/2\alpha^2$ acts as identity on the states $|\alpha e^{i\frac{\pi}{T}t}, -\alpha e^{-i\frac{\pi}{T}t}\rangle$ and $|-\alpha e^{i\frac{\pi}{T}t}, \alpha e^{-i\frac{\pi}{T}t}\rangle$ while it vanishes on the states $|\alpha, \alpha\rangle$ and $|-\alpha, -\alpha\rangle$. The above Hamiltonian thus reduces to the required rotating term $\pi(\hat{a}^\dagger\hat{a} - \hat{b}^\dagger\hat{b})/T$ only when the cat qubits are in a state that is moved around in the phase space, and vanishes otherwise.

2.3 Error models

In this section, we detail the error models of the various gates introduced above. We give a particular attention to the CNOT gate, as this gate is crucial for the stabilizer measurement and hence it is important to have a detailed understanding of its errors in order to assess the overall performance of the scheme (Chapter 4). On the other hand, this discussion about the error models does not include the $ZZZ(\theta)$ and the SWAP gates. These gates are not used later in the circuit simulations, hence the derivation of their precise error model is independent of the rest of this work, and requires a careful treatment that we postpone to future work.

We analyze the error models resulting from two different sources of errors: the single-photon loss of the QHO at a rate κ_1, and the non-adiabaticity of the gates. Actually, apart from state preparation and measurement, and in the absence of any source of decoherence, all the gates have unit fidelity in the limit where the gate

time is infinite. We discuss phase-flip and bit-flip errors in two different ways.

The phase-flip errors are exponentially dominant. For these errors, we give explicit analytical formulas. More precisely, the analytical formula for the phase-flip errors induced by photon loss are explicitly calculated. The analytical formula for the phase-flip errors induced by non-adiabaticity are derived using a combination of a systematic adiabatic elimination theory to guess the scaling of the formula together with a numerical fit to determine the constant prefactor. Using this method, we were able to derive the non-adiabatic phase-flip errors for the topological gates (actually, the X gate has no non-adiabatic phase-flip error when the feed-forward Hamiltonian is added). While we were writing this dissertation, a very thorough study of all the gates on dissipative cat qubits was published [26]. Using a new method introduced in this paper, based on the "shifted Fock basis" adapted to the cat states, the authors were able to also derive analytically the non-adiabatic phase-flip errors for the Zeno gates, thus completing the analysis of phase-flip errors. For the sake of completeness, we give these formulas without including the derivation, which is thoroughly exposed in [26]. Because the phase-flip errors induced by the natural losses of the QHO increase with the gate time, while the phase-flip errors induced by non-adiabaticity decrease with the gate time, the combination of these two sources of errors gives rise to an optimal finite gate time that minimizes the phase-flip errors. We give these optimal gate times based on the analytical formulas, and the error models obtained for these optimal gate times are the input of the circuit simulations of Chapter 4.

We claim that the cat qubit encoding, the two photon stabilization, and the careful bias-preserving implementations of the gates, result in gates for which the bit-flip errors are exponentially suppressed even during the execution of the gates, this point being crucial for the whole construction of the scheme. Here, we give numerical evidence for this claim by performing numerical process tomography of two gates, the $Z(\theta)$ and the CNOT gate, for increasing cat sizes, for which the exponential suppression of bit-flips is indeed observed. We deem that checking the exponential suppression of bit-flips on these two gates is enough, because the other Zeno gates have a similar mechanism as the $Z(\theta)$ gate. For the topological gates, while the X gate, in presence of the feed-forward Hamiltonian, is actually equivalent to the two-photon pumping under a change of frame (see below), the Toffoli gate was beyond reach of numerical simulations. Again, the recent and thorough study [26] filled in this gap by using the shifted Fock basis in which the Toffoli gate was simulated and the exponential suppression of bit-flips verified numerically.

2.3.1 Identity and SPAM errors

Photon loss phase-flip and bit-flip errors The dynamics of an idling cat qubit subject to photon loss is modelled by the master equation

$$\frac{d\rho}{dt} = \kappa_2 \mathcal{D}[\hat{a}^2 - \alpha^2]\rho + \kappa_1 \mathcal{D}[\hat{a}]\rho \tag{2.3}$$

where κ_2 is the rate of the engineered two-photon dissipation and κ_1 the rate of single photon loss. The photon loss operator $\hat{a}$ induces a phase-flip error on the cat qubit, $\hat{a}|\mathcal{C}_\alpha^\pm\rangle = \alpha \tanh(|\alpha|^2)^{\pm 1/2}|\mathcal{C}_\alpha^\mp\rangle$. Thus, the phase-flip error probability induced by single photon loss at rate κ_1 during a time T is given by $p_Z = \bar{n}\kappa_1 T$. This leading order contribution can be calculated explicitly by considering the evolution of the cat state $|\mathcal{C}_\alpha^\pm\rangle$ under the evolution (2.3). Assuming that $\kappa_1 \ll \kappa_2$ such that the dynamics remains in the cat qubit manifold and looking for a solution of the form $\rho(t) = (1-p(t))|\mathcal{C}_\alpha^+\rangle\langle\mathcal{C}_\alpha^+| + p(t)|\mathcal{C}_\alpha^-\rangle\langle\mathcal{C}_\alpha^-|$, the evolution of the population of the cat states is given by (dropping the terms exponentially small in α)

$$\dot{p}(t) = \kappa_1 \bar{n}(1 - 2p(t)).$$

Thus, starting from the initial cat state $|\mathcal{C}_\alpha^\pm\rangle$ ($p(0) = 0$), the phase-flip error probability is given by $p(t) = \frac{1}{2}(1 - e^{-2\bar{n}\kappa_1 t}) \approx \bar{n}\kappa_1 t$ when $\bar{n}\kappa_1 t \gg 1$.

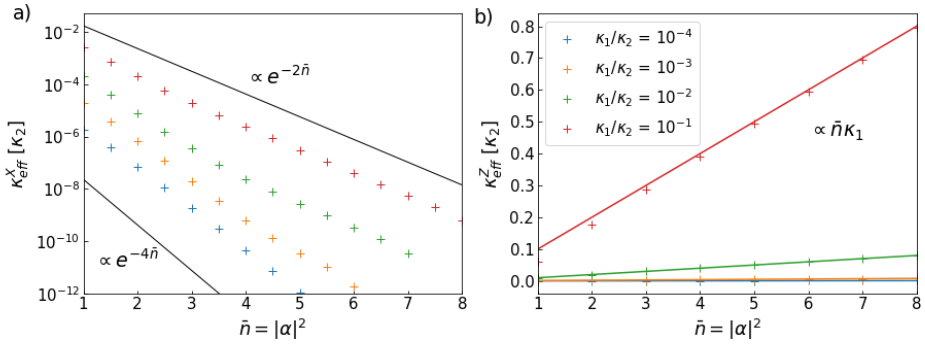


FIGURE 2.4: Numerical simulation of the effective bit-flip a) and phase-flip b) error probability induced by photon loss at rate κ_1.

While it is possible to derive explicitly the dominant phase-flip error probability, the exact calculation of the (exponentially suppressed) bit-flip error probability is a hard problem (the same holds for the error models of the gates, where analytical formulas can be derived for the dominant phase-flip errors induced either by photon loss or by the finite gate times but not for the exponentially small bit-flip errors). Rather, we confirm numerically (Figure 2.4 that the bit-flip errors are exponentially suppressed with the average number of photon $\bar{n}$ in the cat qubit. We note that even though Figure 2.4 is plotted using the effective dynamics (2.3), these scalings have been observed experimentally in [83].

Finally, we expect that the preparation of the cat states $|\mathcal{C}_\alpha^\pm\rangle$ and the measurement of these states can be performed with a similar phase-flip error probability $p_Z = \bar{n}\kappa_1 T$, where T is the typical preparation and measurement time.

2.3.2 Zeno gates

2.3.2.1 $Z(\theta)$ gate

The master equation describing the rotation of an angle θ around the Z axis in time $T = \dfrac{\theta}{4\sqrt{\bar{n}}|\epsilon_Z|}$ is

$$\dot{\rho} = -i[\epsilon_Z \hat{a} + \epsilon_Z^* \hat{a}^\dagger, \rho] + \kappa_2 \mathcal{D}[\hat{a}^2 - \alpha^2]\rho. \tag{2.4}$$

Photon loss phase-flip errors The phase-flip errors induced by photon loss at rate κ_1, described by adding the term $\kappa_1 \mathcal{D}[\hat{a}]\rho$ to the above master equation, commute with the gate at all times. For this reason, the effect of photon loss can be accounted for separately, and the phase-flip errors induced by photon loss are the same as in the memory case

$$p_Z[\text{photon loss}] = \bar{n}\kappa_1 T = \frac{\kappa_1 \sqrt{\bar{n}}\theta}{4|\epsilon_Z|}$$

Non-adiabatic phase-flip errors In [26], the analytical formula proposed for the non-adiabatic phase-flip errors is

$$p_Z[\text{non-adiabaticity}] = \frac{\theta^2}{16\kappa_2 \bar{n}^2 T} = \frac{\theta|\epsilon_Z|}{4\kappa_2 \bar{n}^{3/2}}.$$

We perform a numerical simulation of master equation (2.4) in Figure 2.5, for a rotation of angle $\theta = \pi$ and in the absence of photon loss (that is, to check the non-adiabatic error model). The dotted points (numerical results) are in good agreement with the analytical formula (blue curve).

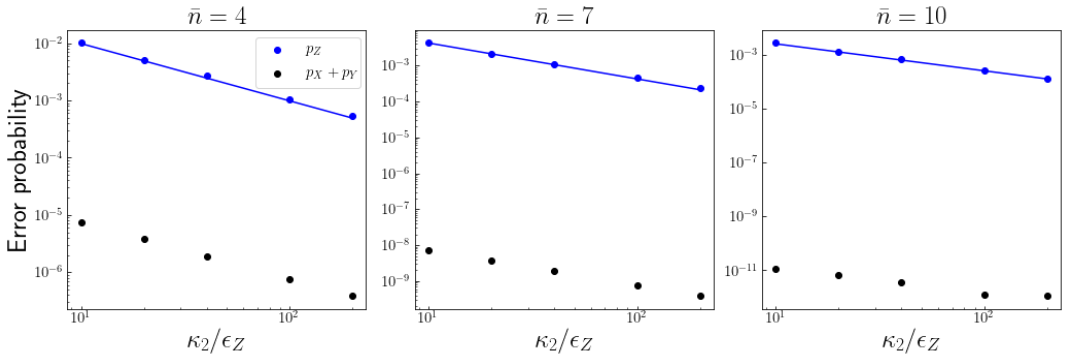


FIGURE 2.5: Numerical simulation of the non-adiabatic errors of the $Z = Z(\pi)$ gate implemented by the master equation (2.4). The non-adiabatic phase-flip error Z is linearly suppressed with the gate time T, while the bit-flip type errors X and Y are exponentially suppressed with the mean number of photons $\bar{n}$.

Optimal gate time Taking into account both the phase-flip errors induced by photon loss and by non-adiabaticity, the total phase-flip error probability for a $Z(\theta)$ gate implemented in time T is given by

$$p_Z = \bar{n}\kappa_1 T + \frac{\theta}{16\kappa_2 \bar{n}^2 T}$$

which is minimal for $T^* = \frac{\sqrt{\theta}}{4\bar{n}^{3/2}\sqrt{\kappa_1\kappa_2}}$, for which the phase-flip rate is

$$p_Z = \frac{\sqrt{\theta}}{2\sqrt{\bar{n}}}\sqrt{\frac{\kappa_1}{\kappa_2}}.$$

2.3.2.2 CZ gate

The same analysis has been carried through in [26] for the $ZZ(\theta)$ gate, from which the CZ gate can be implemented by combining the single qubit Z rotations. While the errors induced by photon loss result in independent Z errors on the two cat qubits with same probability $\bar{n}\kappa_1 T$ as before, it is shown that the non-adiabatic phase-flip errors result in both independent Z_1 and Z_2 errors as well as correlated Z_1Z_2 errors. The analytical formula for the overall phase-flip errors are given by

$$p_{Z_1} = p_{Z_2} = \bar{n}\kappa_1 T + \frac{\theta^2}{64\kappa_2 \bar{n}^2 T}$$
$$p_{Z_1Z_2} = \frac{\theta^2}{32\kappa_2 \bar{n}^2 T}.$$

Note that the photon loss induced errors increase linearly with T while the non-adiabatic errors decrease linearly with T. The optimal gate time minimizing these errors is given by $T^* = \frac{\sqrt{\theta}}{4\sqrt{2}\bar{n}^{3/2}\sqrt{\kappa_1\kappa_2}}$.

2.3.3 Topological gates

2.3.3.1 X gate

The master equation implementing the topological X gate in time T is given by

$$\dot{\rho} = i[\frac{\pi}{T}\hat{a}^\dagger\hat{a}, \rho] + \kappa_2\mathcal{D}[\hat{a}^2 - (\alpha e^{i\frac{\pi}{T}t})^2],$$

where the (optional) Hamiltonian term is added to compensate the non-adiabatic phase-flip errors while the dissipative term implements the continuous deformation of the cat qubit subspace. Actually, for this gate, the Hamiltonian removes *all* of the non-adiabatic phase-flip errors. Indeed, in the rotating frame of this Hamiltonian, the dynamics reads

$$\dot{\rho} = \kappa_2\mathcal{D}[\hat{a}^2 - \alpha^2]$$

which is simply the two-photon stabilization. Thus, for this gate, there are only the phase-flips errors induced by photon loss, which are given by

$$p_Z = \bar{n}\kappa_1 T.$$

In this effective model, it seems that the gate can be executed arbitrarily fast (the optimal gate time is $T^* \to 0$). In practice we will be limited by the adiabatic elimination of the buffer mode that is required for the implementation of the two-photon dissipation (see subsection 1.4.3 for the generic adiabatic elimination of the buffer mode and section 2.4 for the precise discussion of the experimental implementations). This more thorough study is out of the scope of the current manuscript.

2.3.3.2 CNOT gate

We now investigate the error model of the CNOT gate. As we have argued before, this gate is particularly important for error correction, such that a detailed analysis is provided. In particular, we numerically check that the analysis is robust when adding additional sources of errors on the QHO, including thermal excitation and dephasing.

Photon loss phase-flip errors In order to understand the effect of the loss of a photon during the execution of the CNOT, let us consider the operation approximately generated by the two dissipation channels $\mathcal{L}_{\hat{a}}$ and $\mathcal{L}_{\hat{b}}$. In the cat qubits subspaces where the dynamics is confined, these channels implement a unitary operation of the form:

$$\mathcal{U}(t) = |\alpha\rangle\langle\alpha| \otimes I + |-\alpha\rangle\langle-\alpha| \otimes e^{i\frac{\pi}{T}t\hat{b}^\dagger\hat{b}}$$

with $\mathcal{U}(0) = I \otimes I$ and $\mathcal{U}(T) = \text{CNOT}$.

Consider the effect of a loss of a single photon of the control mode $\hat{a}$ at an arbitrary time $t \in [0, T]$. The noisy quantum operation $\mathcal{E}_{\hat{a}}$ performed instead of the CNOT is given by

$$\begin{aligned}\mathcal{E}_{\hat{a}} &= \mathcal{U}(T-t)[\hat{a} \otimes I]\mathcal{U}(t) \\ &= \alpha|\alpha\rangle\langle\alpha| \otimes I - \alpha|-\alpha\rangle\langle-\alpha| \otimes e^{i\pi\hat{b}^\dagger\hat{b}} \\ &= [\hat{a} \otimes I]\text{CNOT}\end{aligned}$$

which can be written in terms of Pauli operators for the cat qubits as

$$\mathcal{E}_{\hat{a}} = Z_1\text{CNOT}.$$

In other words, the loss of a photon on the control cat qubit causes a phase-flip on that qubit but does not affect the target cat qubit.

On the other hand, a photon loss occurring on the target cat qubit $\hat{b}$ at time t propagates as

$$\begin{aligned}\mathcal{U}(T-t)[I\otimes\hat{b}]\mathcal{U}(t) &= (I\otimes\hat{b})(|\alpha\rangle\langle\alpha|\otimes I + e^{-i\pi\frac{T-t}{T}}|-\alpha\rangle\langle-\alpha|\otimes e^{i\pi\hat{b}^\dagger\hat{b}}) \\ &= (I\otimes\hat{b})(|\alpha\rangle\langle\alpha|\otimes I + e^{-i\pi\frac{T-t}{T}}|-\alpha\rangle\langle-\alpha|\otimes I)\text{CNOT}.\end{aligned}$$

The resulting error

$$I\otimes\hat{b}(|\alpha\rangle\langle\alpha|\otimes I + e^{-i\pi\frac{T-t}{T}}|-\alpha\rangle\langle-\alpha|\otimes I)$$

induced by the propagation of the photon loss can be expressed in terms of the Pauli operators of cat qubits as

$$\hat{U}_{\text{err}}(\theta) = \frac{1}{2}(1+Z_1)Z_2 + \frac{1}{2}e^{i\theta}(1-Z_1)Z_2$$

where $\theta = -i\pi(1-t/T)$ is a random phase. The time of the jump being uniformly distributed over the interval [0,T], the noisy operation $\mathcal{E}_{\hat{b}}$ can be written

$$\begin{aligned}\mathcal{E}_{\hat{b}}(\rho) &= \bar{n}\kappa_1 T\int_{-\pi}^{0}\frac{d\theta}{\pi}\hat{U}_{\text{err}}(\theta)\tilde{\rho}\hat{U}_{\text{err}}(\theta)^\dagger \\ &= \bar{n}\kappa_1 T[\frac{1}{2}Z_2\tilde{\rho}Z_2 + \frac{1}{2}Z_1Z_2\tilde{\rho}Z_1Z_2 + \frac{i}{\pi}Z_1Z_2\tilde{\rho}Z_2 - \frac{i}{\pi}Z_2\tilde{\rho}Z_1Z_2]\end{aligned}$$

where $\tilde{\rho} = \text{CNOT}\rho\text{CNOT}$ is the image of ρ by a perfect CNOT operation and $\bar{n}\kappa_1 T$ is the average number of photons lost in each mode during the execution of the gate. Note that this analytical formula is an approximation that only accounts for the effect of the loss of a single photon loss. In addition to this dominant phase-flip error corresponding to the loss of a single photon, the cat states are also slightly deformed towards the center of the phase space, causing (exponentially small) bit-flip errors. Importantly, while the bit-flip are still exponentially suppressed with the cat size as $e^{-2\bar{n}}$, the constant prefactor in front of this exponential suppression is significantly larger than for the case where the two-photon dissipation is time independent. Also, the loss of more than a single photon result in a phase-flip error rate slightly different from this one. However, the numerical simulation of the process confirms that this approximation captures well most of the errors that are caused by photon loss.

The factorization of the operation $\mathcal{E}_{\hat{b}}$ as a perfect CNOT gate followed by some noise operators makes it easier to analyze the effect of the errors. The first two terms indicate that the effect of photon loss on the target cat qubit produces two types of errors of the same strength: phase-flips on the target cat qubit $\frac{1}{2}Z_2\tilde{\rho}Z_2$ as well as a correlated phase-flips on both qubits $\frac{1}{2}Z_1Z_2\tilde{\rho}Z_1Z_2$, with some degree of coherence between these two errors.

When losses on both modes are taken into account, the noisy CNOT gate $\mathcal{E}_{\hat{a},\hat{b}}$ is described by the Kraus map:

$$\mathcal{E}_{\hat{a},\hat{b}}(\rho) = \sum_{k=1,2,3} M_k \tilde{\rho} M_k^\dagger$$

where, noting $r = \frac{1}{2}\arcsin(2/\pi)$, the Kraus operators are given by

$$\begin{aligned} M_1 &= \sqrt{\bar{n}\kappa_1 T} Z_1 \\ M_2 &= \sqrt{\tfrac{\bar{n}\kappa_1 T}{2}}(\cos r I_1 + i \sin r Z_1) Z_2 \\ M_3 &= \sqrt{\tfrac{\bar{n}\kappa_1 T}{2}}(\sin r I_1 + i \cos r Z_1) Z_2. \end{aligned}$$

Non-adiabatic phase-flip errors Let us now consider the phase-flip errors induced by non-adiabaticity. The addition of the feed-forward Hamiltonian

$$\hat{H} = \frac{1}{2}\frac{\pi}{T}\frac{\hat{a}-\alpha}{2\alpha} \otimes (\hat{b}^\dagger \hat{b} - \bar{n}) + h.c.$$

compensates most of the errors induced by the finite gate time T, and it is possible to characterize the remaining errors. Using the systematic adiabatic elimination techniques of [12], one can check that it is only composed of phase-flips on the control cat qubit Z_1, with a rate proportional to $(\bar{n}\kappa_2 T)^{-1}$

The exact coefficient of proportionality is estimated by a numerical fit and is found to be around 0.159. In our work [60], we used the approximation $0.159 \approx (2\pi)^{-1}$. The phase-flip probability is given by

$$p_{Z_1}[\text{non-adiabaticity}] = 0.159(\bar{n}\kappa_2 T)^{-1}.$$

We note that using the shifted Fock basis, the authors of [26] derived an analytical error model and found that

$$p_{Z_1}[\text{non-adiabaticity}] = \frac{\pi^2}{64}(\bar{n}\kappa_2 T)^{-1},$$

in close agreement with our numerical estimation $\frac{\pi^2}{64} \approx 0.154$.

Optimal gate time The probability of the "environment" induced phase-flip errors, *e.g* by photon loss, increase linearly with the gate time T, whereas phase-flip errors caused by non-adiabaticity are reduced when the gate time is increased. This opposite behavior gives rise to a finite optimal gate time T^* for which the gate fidelity is maximal.

More precisely, taking into account phase-flip errors caused by both photon loss and non-adiabaticity, the total phase-flip error probability on the control cat qubit is

given by

$$p_{Z_1} = p_{Z_1}[\text{photon loss}] + p_{Z_1}[\text{non-adiabaticity}] = \bar{n}\kappa_1 T + 0.159(\bar{n}\kappa_2 T)^{-1}.$$

The gate fidelity $\mathcal{F}$ of the implemented CNOT operation, defined in equation (2.5), is given by

$$\mathcal{F} = \sqrt{1 - (p_{Z_1} + p_{Z_2} + p_{Z_1 Z_2})} = \sqrt{1 - 2\bar{n}\kappa_1 T - 0.159(\bar{n}\kappa_2 T)^{-1}}.$$

The highest value of the fidelity that can be achieved is set by the ratio κ_1/κ_2

$$\mathcal{F} = \sqrt{1 - 1.13\sqrt{\frac{\kappa_1}{\kappa_2}}},$$

achieved for the optimal gate time

$$T^* = 0.282[\bar{n}\sqrt{\frac{\kappa_1}{\kappa_2}}]^{-1}\kappa_2^{-1}.$$

For the ratio $\frac{\kappa_1}{\kappa_2} = 10^{-3}$ considered in Figure 2.6, this theoretical formula predicts a gate fidelity of $\mathcal{F} = 98.2\%$, in agreement with the numerical simulation.

Numerical process tomography The validity of this error model is checked numerically in Figure 2.6 (a,b,c). The full master equation of the system is simulated in presence of photon loss. The process matrix χ plotted in (a) completely characterizes the quantum operation $\mathcal{E}$ performed via the relation [92]

$$\mathcal{E}(\rho) = \sum_{mn} \chi_{mn} P_m \rho P_n^\dagger$$

where $\{P_j\}$ is the set of two-qubit Pauli operators. The gate fidelity $\mathcal{F}$ is defined as [92]

$$\mathcal{F}(U, \mathcal{E}) = \min_{|\psi\rangle} \mathcal{F}(U|\psi\rangle, \mathcal{E}(|\psi\rangle\langle\psi|)) \tag{2.5}$$

where $U = \text{CNOT}$ is the perfect CNOT operation the minimum is taken over the set of all possible two-qubit states $|\psi\rangle$. The unitary of the perfect CNOT is factored out in order to obtain the process error matrix χ^{err} (real part in (b), imaginary part in (c)), which characterizes the noise alone:

$$\mathcal{E}(\rho) = \sum_{mn} \chi^{\text{err}}_{mn} P_m \tilde{\rho} P_n^\dagger$$

with $\tilde{\rho} = \text{CNOT}\rho\text{CNOT}$ the image of ρ by a perfect CNOT. In other words, we decompose the noisy CNOT into a perfect CNOT followed by some noise process, characterized by the process error matrix χ^{err}. As can be seen in the real part of χ^{err} (Figure 2.6-b), photon loss and non-adiabaticity only cause phase-flip errors Z_1, Z_2 or $Z_1 Z_2$.

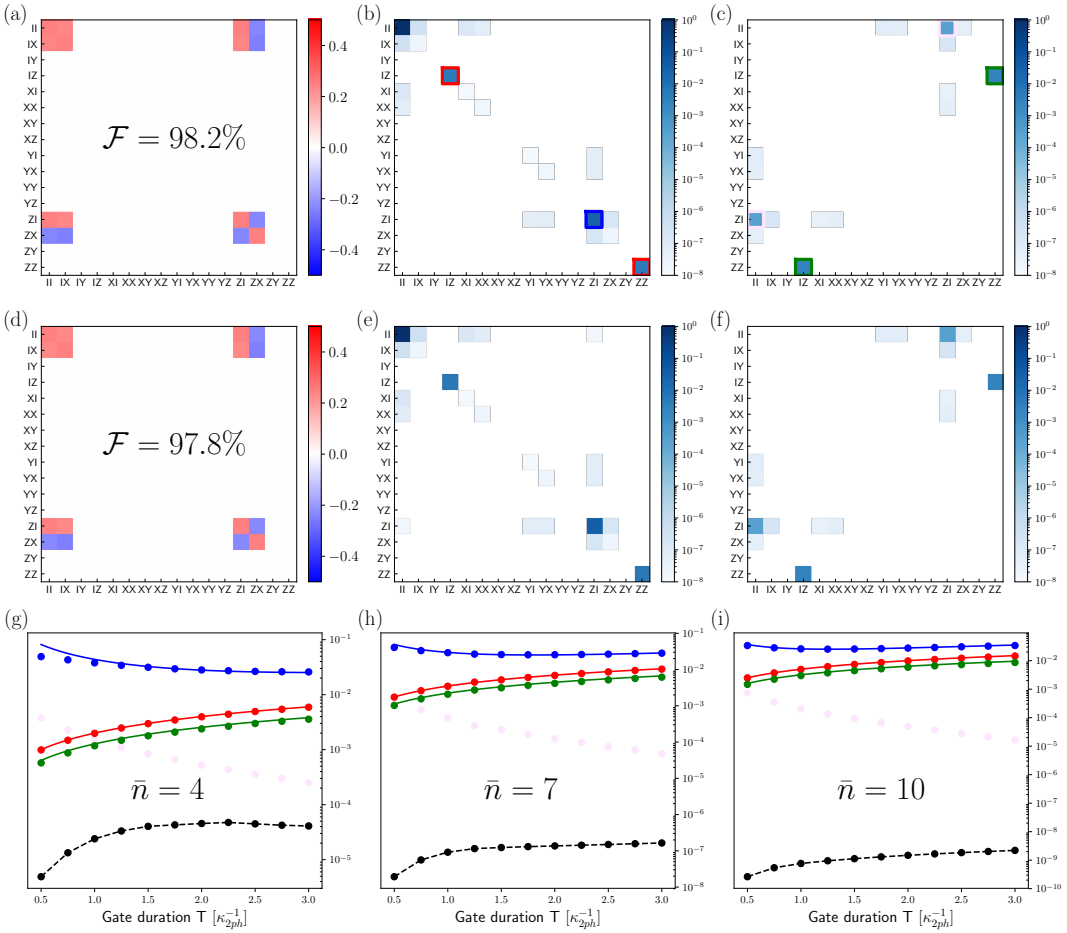


FIGURE 2.6: Process tomography of the CNOT gate in presence of noise. The CNOT process is numerically simulated for $\bar{n} = 7$ photons cat qubits using two different error models. First, we consider photon loss on both modes $\kappa_1\mathcal{D}[\hat{a}] + \kappa_1\mathcal{D}[\hat{b}]$ (a-c). Then, we consider a more elaborate error model including photon loss $\kappa_1(1+n_{th})\mathcal{D}[\hat{a}] + \kappa_1(1+n_{th})\mathcal{D}[\hat{b}]$, thermal excitations $\kappa_1 n_{th}\mathcal{D}[\hat{a}^\dagger] + \kappa_1 n_{th}\mathcal{D}[\hat{b}^\dagger]$ ($n_{th} = 10\%$) and dephasing on both modes $\kappa_\phi\mathcal{D}[\hat{a}^\dagger\hat{a}] + \kappa_\phi\mathcal{D}[\hat{b}^\dagger\hat{b}]$ (d-f). In both cases, we set $\kappa_1/\kappa_2 = 10^{-3}$ and the gate time is chosen optimal $T^* = 0.282[\bar{n}\sqrt{\kappa_1\kappa_2}]^{-1} \approx 1.27\kappa_2^{-1}$ (see main text). We plot the real part of the process matrix χ (a,d), and the real (b,e) and imaginary (c,f) part of the error matrix χ^{err}. In the lower row (g,h,i), we check the validity of the analytical error model for photon loss for various gate times and cat sizes. The dots illustrate the simulation results where the full master equation in presence of loss is considered, the plain lines correspond to the analytical formula provided in the main text. The blue dots correspond to the diagonal process matrix element corresponding to Z_1 errors, the red dots correspond to the coinciding diagonal matrix elements corresponding to Z_2 and Z_1Z_2 errors. The green dots correspond to the off-diagonal elements corresponding to the coherence between Z_2 and Z_1Z_2 errors. The pale magenta dots correspond to the off-diagonal elements corresponding to coherence between Z_1 and I, this coherence is due to high-order non-adiabatic effects (not included in our model). The black dots correspond to all of the remaining errors, including bit-flip type ones. It is clear that these errors are exponentially suppressed with the mean number of photons $\bar{n}$.

We further investigate our theoretical model for errors caused by photon loss by plotting in Figure 2.6(g,h,i) the values of the coefficients of the error matrix χ^{err}

(marked by colored squares) as a function of gate duration. The blue dots correspond to phase-flip errors on the control cat qubit Z_1 induced by a combination of non-adiabatic errors and the photon loss. The plain blue line corresponds to the analytical formula

$$p_{Z_1} = \bar{n}\kappa_1 T + 0.159(\bar{n}\kappa_2 T)^{-1}.$$

The red dots represent the phase-flip errors on target qubit Z_2 and the correlated phase-flip errors Z_1Z_2. These values coincide and are given by

$$P_{Z_2} = P_{Z_1Z_2} = \tfrac{1}{2}\bar{n}\kappa_1 T,$$

as is represented by the plain line in red. The off-diagonal term corresponding to the coherence between Z_2 and Z_1Z_2 errors (green dots) also fit very well our expectation. The pale purple dots correspond to the off-diagonal term representing the coherence between I and Z_1 errors. In order to capture such a coherence, one needs to push the non-adiabatic perturbation techniques [12] up to third order, which we have not yet done. Most importantly, the remaining errors (namely the ones that contain an X or Y Pauli operator) represented by the black lines are exponentially suppressed by the cat size, that confirms the bias-preserving aspect of the gate.

As discussed in [31] and recalled in Chapter 1, in presence of the two-photon pumping scheme, any physical noise process with a local effect in the phase space of the harmonic oscillator causes bit-flips that are exponentially suppressed in the size of the cat qubits, thus preserving the biased structure of the noise. We now provide a numerical evidence of this fact for a more elaborate set of physical noise processes for the superconducting cavity: photon loss $\hat{a}$, thermal excitation $\hat{a}^\dagger$ with a non-zero temperature, and photon dephasing $\hat{a}^\dagger\hat{a}$.

In Figure 2.6, we characterize the performed operation by plotting the process matrix χ (d), and the real part (e) and imaginary part (f) of the error matrix. In this simulation, $\kappa_1/\kappa_2 = 10^{-3}$, the photon loss is given by $\kappa_1(1+n_{\text{th}})\mathcal{D}[\hat{a}]$ and thermal excitations by $\kappa_1 n_{\text{th}}\mathcal{D}[\hat{a}^\dagger]$ with $n_{\text{th}} = 10\%$, and the dephasing on the cavity is given by $\kappa_\phi\mathcal{D}[\hat{a}^\dagger\hat{a}]$ with $\kappa_\phi = \kappa_1$.

Note that the resulting error matrix and gate fidelity are barely affected by the added thermal excitations and photon dephasing. The addition of thermal noise and dephasing slightly decrease the fidelity of the operation, from 98.2% to 97.8%, but as expected, this decrease is caused by an increased rate of phase-flip errors, while all bit-flip errors remain exponentially suppressed.

Very interestingly, the phase-flip error probability induced by the cavity dephasing was computed explicitly in the recent work [26]. The fact that cavity dephasing at rate κ_ϕ, described by the dissipation super-operator $\kappa_\phi\mathcal{D}[\hat{a}^\dagger\hat{a}]$, can lead to phase-flip errors might be surprising. Indeed, the dephasing operator $\hat{a}^\dagger\hat{a}$ commutes with

the photon number parity operator $(-1)^{\hat{a}^\dagger \hat{a}}$, such that one may naively think that it cannot induce transitions between the two cat states $|\mathcal{C}_\alpha^\pm\rangle$ of well-defined photon number parity. While this is in general true for an idling cat qubit, the photon number parity of the cat states *during* the execution of the CNOT gate does not remain well-defined, thus exposing the cat state to some dephasing-induced phase-flips. In [26], it was shown that cavity dephasing $\kappa_\phi \mathcal{D}[\hat{a}^\dagger \hat{a}]$ results in a phase-flip error only on the control cat qubit, with probability

$$p_{Z_1} = \frac{1}{2}\kappa_\phi \bar{n} T.$$

Note that we have not included this error in the rest of our work, for which we assumed that $\kappa_\phi = 0$.

Numerical considerations. The numerical study of the CNOT gate in presence of noise (Figure 2.6) requires the simulation over a time T of the Lindblad master equation

$$\dot{\rho} = -i[\hat{H}, \rho] + \kappa_2 \mathcal{D}[\hat{L}_{\hat{a}}]\rho + \kappa_2 \mathcal{D}[\hat{L}_{\hat{b}}(t)]\rho + \kappa_1 \mathcal{D}[\hat{a}]\rho + \kappa_1 \mathcal{D}[\hat{b}]\rho$$

for 16 different initial states. The numerical computations were performed using the cluster of Inria Paris, composed of 68 nodes for a total of 1244 cores. The nodes are divided in a few hardware generations: 28 bi-processors Intel Xeon X5650 of 6 cores, 12 bi-processors E5-2650v4 2.20 of 12 cores, 16 bi-processors XeonE5-2670 of 10 cores, 8 bi-processors E5-2695 v4 of 18 cores, 4 bi-processors E5-2695 v3 of 14 cores. The simulation of the CNOT gate for cat qubits of $\bar{n} = 10$ and a gate time of $T = 3\kappa_2^{-1}$ takes about 13 hours on the cluster. The simulation of the Toffoli gate with the same parameters would be about 2000 times longer, for this reason, we do not provide numerical simulations for the Toffoli gate.

However, we note that in the recent work [26], the shifted Fock basis adapted to large cat states was introduced, in which the simulation time of the Toffoli gate on a cluster is reasonable.

2.3.3.3 Toffoli gate

Photon loss phase-flip errors The effect induced by photon loss during the execution of the Toffoli gate can be derived in the same way as for the CNOT. A photon loss occurring on one of the two control modes $\hat{a}$, $\hat{b}$ does not propagate to the other modes and results in a dephasing error Z_1 and Z_2, respectively. When the target mode $\hat{c}$ loses a photon, it gives rise to a correlated error between the three modes. More precisely, the noisy Toffoli operation $\mathcal{E}_{\hat{a},\hat{b},\hat{c}}$ can be decomposed into a perfect Toffoli operation, again denoted by

$$\tilde{\rho} = \text{Toffoli}\ \rho\ \text{Toffoli}$$

followed by a noise process modelled by the Kraus map

$$\mathcal{E}_{\hat{a},\hat{b},\hat{c}}(\rho) = \sum_{k=1,2,3,4} M_k \tilde{\rho} M_k^\dagger$$

$$\begin{aligned}
M_1 &= \sqrt{\bar{n}\kappa_1 T} Z_1 \\
M_2 &= \sqrt{\bar{n}\kappa_1 T} Z_2 \\
M_3 &= \sqrt{\bar{n}\kappa_1 T}(\cos r(I_1 I_2 - \mathcal{Z}_{12}) - i \sin r \mathcal{Z}_{12}) Z_3 \\
M_4 &= \sqrt{\bar{n}\kappa_1 T}(\sin r(I_1 I_2 - \mathcal{Z}_{12}) - i \cos r \mathcal{Z}_{12}) Z_3.
\end{aligned}$$

where $\mathcal{Z}_{12} = \frac{1}{4}(I_1 I_2 - Z_1 - Z_2 - Z_1 Z_2)$ acts on the two control cat qubits.

Non-adiabatic phase-flip errors Because of the analogies in the way the CNOT and the Toffoli gates are implemented, it is useful to think of the Toffoli gate as a CNOT where the control state $|-\alpha\rangle$ is replaced by $|-\alpha,-\alpha\rangle$. In particular, the methods of [12] that we used to characterize the effect of non-adiabaticity predict similar results for the Toffoli gate. The analysis of the errors induced by the approximate Hamiltonians was not realized in this work and it still the topic of current research. However, we anticipate that the effect of the finite gate time is to dephase the "trigger" state $|-\alpha,-\alpha\rangle$ with respect to the other three possible states of the pair of control cat qubits. In terms of Pauli operator, this only results in phase-flip errors Z_1, Z_2 and $Z_1 Z_2$ on the two control cat qubits with equal probability $p = (4\pi\bar{n}\kappa_2 T)^{-1}$ but it does not cause any error on the target cat qubit, or bit-flip type errors.

We note that the thorough analysis of [26] include the non-adiabatic phase-flip errors for the Toffoli gate and are in accordance with these predictions.

Optimal gate time Taking into account phase-flip errors caused by photon loss and non-adiabaticity, the gate fidelity $\mathcal{F}$ of the implemented Toffoli operation is given by

$$\begin{aligned}
\mathcal{F} &= [1 - p_{Z_1} - p_{Z_2} - p_{Z_3} - p_{Z_1 Z_2} - p_{Z_1 Z_3} - p_{Z_2 Z_3} - p_{Z_1 Z_2 Z_3}]^{\frac{1}{2}} \\
&= [1 - \frac{3}{4\bar{n}\pi\kappa_2 T} - 3\bar{n}\kappa_1 T]^{\frac{1}{2}}.
\end{aligned}$$

This fidelity is maximum for the same gate time $T^* = [2\bar{n}\sqrt{\pi\frac{\kappa_1}{\kappa_2}}]^{-1}\kappa_2^{-1}$ optimizing the CNOT gate, producing a gate fidelity of

$$\mathcal{F} = \sqrt{1 - \sqrt{\frac{9}{\pi}\frac{\kappa_1}{\kappa_2}}}.$$

The ratio $\frac{\kappa_1}{\kappa_2} = 10^{-3}$ considered in Figure 2.6 corresponds to a gate fidelity $\mathcal{F} = 97.3\%$ with respect to a perfect Toffoli. Note that the optimal gate time for the CNOT and the Toffoli gate decreases with the mean number of photons $\bar{n}$.

2.4 Experimental realization

We now discuss how all the different operations proposed above could be realized within the framework of circuit QED. As it will become clear throughout this section, all the recipes for realizing these operations belong to the well studied class of parametric methods with circuit QED. The multi-wave mixing property of Josephson junctions in a transmon or in a more elaborate circuit such as the ATS [83] can be combined with the application of microwave drives (also called pumps) at well chosen frequencies to realize the stabilization of the cat qubits, and to process the information encoded in them. Throughout this section, one can roughly judge of whether the operation that we propose is reasonable by looking at the order of non-linearity required to implement the operation. Indeed, while low orders parametric processes (at most quadratic or cubic) are now more and more common in the framework of circuit QED, it remains a formidable challenge to engineer higher order non-linearities with sufficient strength. In all of the implementations proposed below, we restrict ourselves to (rather) low order non-linearities with these facts in mind, such that our whole scheme shall seem reasonable to implement in a near future. Specifically, the most difficult operation to implement experimentally (according to the non-linearity order metric) should be the feed-forward Hamiltonian required for the reduction of the phase-flip error rate of the Toffoli gate.

We begin in subsection 2.4.1 by describing how the two-photon pumping scheme that stabilizes the cat qubit is implemented, following the experiments [78, 118, 83]. Then, we discuss in subsection 2.4.2 how the weak Hamiltonians required for the Zeno gates have been realized [118] or could be realized [87]. Last, we discuss the realization of the topological gates in subsection 2.4.3. For each gate, the discussion is split in two parts: the (required) implementation of the time dependence of the two-pumping scheme that realized the topological deformation of the cat qubit code space, and the (optional) implementation of the feed-forward Hamiltonians that might be added during the gates execution to reduce the phase-flip errors induced by non-adiabaticity.

2.4.1 Two-photon pumping scheme

We have argued in Chapter 1 that the effective two-photon driven-dissipative stabilization of the cat qubit $\kappa_2\mathcal{D}[\hat{a}^2 - \alpha^2]$ on a high Q mode $\hat{a}$ can be implemented using a low Q "buffer" mode $\hat{b}$ by the dynamics

$$\dot{\rho} = -i[\hat{H}_{\text{int}}, \rho] + \kappa\mathcal{D}[\hat{b}]\rho$$

where

$$\begin{aligned}\hat{H}_{\text{int}} &= \hat{H}_{\text{conversion}} + \hat{H}_{\text{drive}} \\ \hat{H}_{\text{conversion}} &= g\hat{a}^2\hat{b}^\dagger + g^*\hat{a}^{\dagger 2}\hat{b} \\ \hat{H}_{\text{drive}} &= -g\alpha^2\hat{b}^\dagger - g^*\alpha^{*2}\hat{b}.\end{aligned}$$

The drive Hamiltonian $\hat{H}_{\text{drive}}$ can be simply implemented by applying a resonant drive on the buffer mode of appropriate amplitude and phase. The Hamiltonian $\hat{H}_{\text{conversion}}$ is a non-linear interaction modelling the coherent conversion of two photons of the memory mode $\hat{a}$ into a single photon of the buffer mode $\hat{b}$. In order to implement this Hamiltonian, it is necessary to use a non-linear circuit element. Within the framework of circuit QED, this non-linear circuit element is the Josephson junction. In the next two subsections, we review two different ways to use a Josephson junction to implement the above Hamiltonian. The first scheme relies on a transmon and was realized experimentally in [78, 118]. The second one, realized experimentally in [83], relies on a more elaborate circuit element called the Asymmetrically threaded SQUID (ATS) whose non-linearity is inherited from the Josephson junction.

2.4.1.1 Two-photon exchange using a transmon

The first experimental demonstration of the conversion Hamiltonian, proposed in [87] and realized in [78], was achieved by coupling the two modes to a Josephson junction in a bridge transmon configuration [68]. More precisely, the Josephson junction is used as a four-wave mixer to generate the coupling that converts two-photons of the memory mode $\hat{a}$ into one photon of the buffer mode $\hat{b}$. In order to do so, the buffer is pumped off-resonantly at the angular frequency

$$\omega_p = 2\omega_a - \omega_b$$

where $\omega_{a,b}$ are the frequencies of the memory and buffer mode, respectively, which stimulates the conversion of two photons of the mode $\hat{a}$ into one photon of the pump and one photon of the buffer $\hat{b}$. Additionally, the Hamiltonian $\hat{H}_{\text{drive}}$ is implemented by a weak resonant irradiation of the buffer.

Derivation of the two-photon exchange Hamiltonian The resulting Hamiltonian of the two modes coupled to the Josephson junction with the two tones (the drive and the pump) on the buffer mode is given by

$$\begin{aligned}\hat{H}/\hbar &= \bar{\omega}_a\hat{a}^\dagger\hat{a} + \bar{\omega}_b\hat{b}^\dagger\hat{b} - \frac{E_J}{\hbar}(\cos(\hat{\varphi}) + \hat{\varphi}^2/2) + 2\mathcal{R}(\epsilon_p e^{-i\omega_p t} + \epsilon_d e^{-i\omega_d t})(\hat{b} + \hat{b}^\dagger) \\ \hat{\varphi} &= \varphi_a(\hat{a} + \hat{a}^\dagger) + \varphi_b(\hat{b} + \hat{b}^\dagger).\end{aligned}$$

The first two terms are the quadratic Hamiltonian of the two modes, where the bare frequencies $\bar{\omega}_{a,b}$ are shifted due to the contribution of the Josephson junction in the Hamiltonian. The cosine term is the Josephson junction Hamiltonian (where the quadratic terms are included in the quadratic part of the Hamiltonian) where $\hat{\varphi}$ is the phase across the junction, to which both modes $\hat{a}$ and $\hat{b}$ participate with respective coefficients φ_a and φ_b. The last term describes the two (pump and drive) tones applied to the buffer mode, with complex amplitude ϵ_p and ϵ_d and frequencies ω_p and ω_d respectively.

The regime of parameters producing the correct term is

$$\omega_p, \omega_d, \bar{\omega}_{a,b} \gg \epsilon_p, \omega_p - \bar{\omega}_b \gg \frac{E_J}{\hbar}\frac{\|\hat{\varphi}\|^4}{4!}.$$

Following [78], the fast time scales corresponding to the system frequencies and pump amplitude are eliminated by the change of basis

$$\begin{aligned} U &= e^{i\omega_d t \hat{b}^\dagger \hat{b}} e^{i\frac{\omega_d+\omega_d}{2} t \hat{a}^\dagger \hat{a}} e^{-\tilde{\xi}_p \hat{b}^\dagger + \tilde{\xi}_p^* \hat{b}} \\ \frac{d\tilde{\xi}_p}{dt} &= -i\bar{\omega}_b \tilde{\xi}_p - 2i\mathcal{R}(\epsilon_p e^{-i\omega_p t}) - \frac{\kappa_b}{2}\tilde{\xi}_p. \end{aligned}$$

The amplitude of the displacement quickly converges (in a few κ_b^{-1}) to

$$\tilde{\xi}_p \approx \xi_p e^{-i\omega_p t} = \frac{-i\epsilon_p}{\kappa_b/2 + i(\bar{\omega}_b - \omega_p)} e^{-i\omega_p t}.$$

In this frame, the Hamiltonian is given by

$$\begin{aligned} \tilde{H}/\hbar &= (\bar{\omega}_b - \omega_d)\hat{b}^\dagger \hat{b} + (\bar{\omega}_a - \frac{\omega_d + \omega_p}{2}\hat{a}^\dagger \hat{a} - \frac{E_J}{\hbar}(\cos(\tilde{\varphi}) + \tilde{\varphi}^2/2) \\ \tilde{\varphi} &= \varphi_a(e^{-i\frac{\omega_p+\omega_d}{2}t}\hat{a} + e^{i\frac{\omega_p+\omega_d}{2}t}\hat{a}^\dagger) + \varphi_b(e^{-i\omega_d t}\hat{b} + e^{i\omega_d t}\hat{b}^\dagger + \tilde{\xi}_p + \tilde{\xi}_p^*). \end{aligned}$$

Expanding the cosine up to the fourth order and performing the rotating wave approximation, this Hamiltonian becomes

$$\hat{H} = \hat{H}_{\text{shift}} + \hat{H}_{\text{Kerr}} + \hat{H}_{\text{int}}$$

where

$$\begin{aligned} \hat{H}_{\text{shift}} &= (\bar{\omega}_b - \omega_d - \delta_b - \chi_{bb}|\xi_p|^2)\hat{b}^\dagger \hat{b} + (\bar{\omega}_a - \frac{\omega_p + \omega_d}{2} - \delta_a - \chi_{ab}|\xi_p|^2)\hat{a}^\dagger \hat{a} \\ \hat{H}_{\text{Kerr}} &= -\tfrac{1}{2}\chi_{aa}\hat{a}^{\dagger 2}\hat{a}^2 - \tfrac{1}{2}\chi_{bb}\hat{b}^{\dagger 2}\hat{b}^2 - \chi_{ab}\hat{a}^\dagger \hat{a}\hat{b}^\dagger \hat{b} \\ \hat{H}_{\text{int}} &= g_2^* \hat{a}^2 \hat{b}^\dagger + g_2 \hat{a}^{\dagger 2}\hat{b} + \epsilon_d \hat{b}^\dagger + \epsilon_d^* \hat{b}. \end{aligned}$$

The Hamiltonian $\hat{H}_{\text{shift}}$ corresponds to the modes frequency shifts, caused by the AC Stark shift induced by the pump ($|\xi_p|^2$), and by a term due to the particular

ordering of the self-Kerr terms ($\delta_{a,b}$). This Hamiltonian vanishes when the frequency matching conditions $\omega_d = \omega_b$ and $\omega_p = 2\omega_a - \omega_b$ are satisfied with

$$\omega_a = \bar{\omega}_a - \delta_a - \chi_{ab}|\xi_p|^2$$
$$\omega_b = \bar{\omega}_b - \delta_b - \chi_{bb}|\xi_p|^2.$$

The Hamiltonian $\hat{H}_{\text{Kerr}}$ corresponds to the self-Kerr (χ_{aa}, χ_{bb}) and the cross-Kerr (χ_{ab} terms), where $\chi_{aa} = \frac{E_J}{\hbar}\varphi_a^4/2$, $\chi_{bb} = \frac{E_J}{\hbar}\varphi_b^4/2$ and $\chi_{ab} = \frac{E_J}{\hbar}\varphi_a^2\varphi_b^2$.

Finally, the Hamiltonian $\hat{H}_{\text{int}}$ contains the two terms required to implement the two-photon driven-dissipative scheme. The coupling strengh g_2 is given by

$$g_2 = \chi_{ab}\xi_p^*/2. \tag{2.6}$$

Limitations As we have seen in section 2.3, the crucial point in the design of the cat qubit is the ability to engineer a two-photon dissipation rate much larger than the rate at which single photon decay. Combining the expression (1.7) from Chapter 1 with (2.6), the two-photon dissipation rate is given by (under the assumption $|g_2| < \kappa_b$)

$$\kappa_2 = 4|g_2|^2/\kappa_b = \chi_{ab}^2|\xi_p|^2/\kappa_b.$$

This rate is proportional to the the pump power $|\xi_p|^2$ and to the memory-buffer cross-Kerr χ_{ab}, and the assumption that $|g_2| < \kappa_b$ implies that κ_2 is bounded by a value of order $|g_2| = \chi_{ab}|\xi_p|/2$. However, it is not possible to pump arbitrarily hard as this would lead to undesired effects such as a reduction of the coherence time or an increase of the effective temperature of the involved modes. These undesired effects are partially explained by the escape into the unconfined states due to the bounded potential of the Josephson junction [82]. In [78], the value of $|\xi_p|^2 = 1.2$ was achieved with a pump power of 100mW, the cross-Kerr $\chi_{ab}/2\pi = 200$kHz. Given the value of the $T_1^b = 0.025\mu s$ of the buffer, this corresponds to a ratio $\kappa_1/\kappa_2 \approx 1$.

This experiment was improved in [118] to demonstrate a single qubit gate on the cat (more detail in 2.4.2). The major improvement was the use of a post-cavity [106] for the memory mode for which a longer lifetime with respect to energy decay $T_1 = 92\mu s$ was achieved (versus $T_1 = 20\mu s$ in [78]). In this experiment, the parameters achieved for the single-photon and two-photon dissipation achieved were $\kappa_1/2\pi = 1.7$kHz and $\kappa_2/2\pi = 176$kHz, achieving a ratio of 1%, close to the threshold required to implement active error correction (.5%, see Chapter 3).

However, in these two experiments, the spurious cross-Kerr terms prevented the observation of the exponential suppression of the bit-flip rate. Indeed, in these experiments a large cross-Kerr strength with respect to the desired two-photon interaction g_2, causes an effective suppression of this interaction as soon as the buffer mode is thermally excited. Such undesired effects should be totally suppressed as soon as

one reaches g_2 values that exceed the cross-Kerr interaction. We now introduce the second generation of experiment developed to overcome this problem.

2.4.1.2 Two-photon exchange using an ATS

The Asymmetrically Threaded SQUID (ATS) In order to circumvent the limitations of the first experiments, a novel non-linear circuit element, called the Asymmetrically Threaded SQUID (ATS) was realized in [83] to implement the two-to-one photon conversion Hamiltonian $\hat{H}_{\text{int}}$ between the memory and the buffer mode. It consists in a SQUID (Superconducting Quantum Interference Device) shunted in its center by a large inductance, thus forming two loops. Before we detail how the two-photon exchange Hamiltonian is obtained from the interaction Hamiltonian between the buffer mode and the memory mode, let us recall the properties of the ATS. The potential energy of this element alone depends on only one degree of freedom, the phase φ across the inductor, and is given by

$$U(\varphi) = \frac{1}{2}E_{L,b}\varphi^2 - E_{J,1}\cos(\varphi + \varphi_{\text{ext},1}) - E_{J,2}\cos(\varphi + \varphi_{\text{ext},2}).$$

Denoting $E_{J,1} = E_J + \Delta E_J$ and $E_{J,1} = E_J - \Delta E_J$, the potential energy can be written

$$U(\varphi) = \frac{1}{2}E_{L,b}\varphi^2 - 2E_J\cos(\varphi_\Sigma)\cos(\varphi + \varphi_\Delta) + 2\Delta E_J \sin(\varphi_\Sigma)\sin(\varphi + \varphi_\Delta)$$

where $\varphi_\Sigma = \frac{1}{2}(\varphi_{\text{ext},1} + \varphi_{\text{ext},2})$ and $\varphi_\Delta = \frac{1}{2}(\varphi_{\text{ext},1} - \varphi_{\text{ext},2})$. Setting $\varphi_\Sigma = \pi/2 + \epsilon(t) = \pi/2 + \epsilon_0\cos(\omega_p t)$ and $\varphi_\Delta = \pi/2$, where ϵ_0 is small, the time-dependent potential energy becomes (to the first order in $\epsilon(t)$)

$$U(\varphi) = \frac{1}{2}E_{L,b}\varphi^2 - 2E_J\epsilon(t)\sin(\varphi) + 2\Delta E_J\cos(\varphi).$$

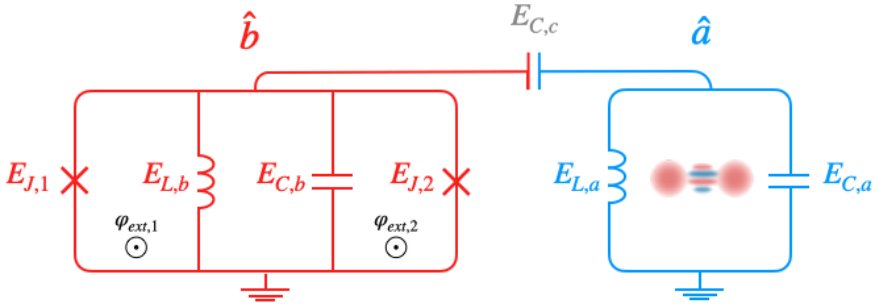


FIGURE 2.7: Circuit representation of the low Q mode of an ATS (red) coupled capacitively to a high memory mode $\hat{a}$ (red) hosting the cat qubit. The ATS is DC biased at the asymmetric flux bias point $\varphi_\Sigma = \frac{1}{2}(\varphi_{\text{ext},1} + \varphi_{\text{ext},1}) = \pi/2 + \epsilon_0\cos(\omega_p t)$, $\varphi_\Delta = \frac{1}{2}(\varphi_{\text{ext},1} - \varphi_{\text{ext},1}) = \pi/2$ to mediate the required two-photon exchange between the memory and the buffer mode (see main text).

Now, because of the shunt, this potential is unbounded which prevents the system to escape to unconfined state like in the case of an unshunted Josephson junction. In practice, the inductance can be replaced by an array of N Josephson junctions for which the potential energy $\frac{1}{2}E_{L,b}\varphi^2$ is replaced by $NE_{J,L}\cos(\varphi/N)$ where $E_{J,L}$ is the Josephson of each individual junction of the array; which is now a bounded potential, but for which the number of junctions N can be chosen such that the depth of the potential is high enough ($NE_{J,L} \gg 2E_J\epsilon_0$) so that the state remains confined in the parabolic part of the cosine potential.

Derivation of the two-photon exchange Hamiltonian The two modes we consider are a high Q mode (the memory $\hat{a}$) and the (low Q) mode of the ATS. When these two modes are coupled using the ATS as in Figure 2.7, and a weak resonant drive is applied on the buffer, the Hamiltonian governing the dynamics is given by

$$\hat{H}/\hbar = \omega_a\hat{a}^\dagger\hat{a} + \omega_b\hat{b}^\dagger\hat{b} - 2E_J\epsilon(t)\sin(\hat{\varphi}) + 2\mathcal{R}(\epsilon_d e^{-i\omega_d t})(\hat{b}+\hat{b}^\dagger)$$

where $\hat{\varphi} = \hat{\varphi}_a + \hat{\varphi}_b = \varphi_a(\hat{a}+\hat{a}^\dagger) + \varphi_b(\hat{b}+\hat{b}^\dagger)$ is the global phase across the ATS dipole, written as the sum of the phase across each of the two modes $\hat{a}$ and $\hat{b}$ weighted by the contribution $\varphi_{a,b}$ of each mode to the zero point fluctuations of $\hat{\varphi}$.

Expanding the sine up to the third order, the Hamiltonian reads

$$\hat{H}/\hbar = \omega_a\hat{a}^\dagger\hat{a} + \omega_b\hat{b}^\dagger\hat{b} - 2E_J\epsilon(t)\hat{\varphi}_a - 2E_J\epsilon(t)\hat{\varphi}_b + \tfrac{1}{3}E_J\epsilon(t)(\hat{\varphi}_a+\hat{\varphi}_b)^3 + 2\mathcal{R}(\epsilon_d e^{-i\omega_d t})(\hat{b}+\hat{b}^\dagger).$$

The rest of the derivation proceeds as in the case of the single junction. By going to the frame displaced by $\xi_a(t) = \xi_a e^{-i\omega_p t}$ for $\hat{a}$ and $\xi_b(t) = \xi_b e^{-i\omega_p t}$ for $\hat{b}$, to the frame rotating at frequency $(\omega_p+\omega_d)/2$ for $\hat{a}$ and ω_d for $\hat{b}$ and keeping only the non rotating terms, the Hamiltonian reads

$$\hat{H} = \hat{H}_{\text{shift}} + \hat{H}_{\text{int}}$$

where

$$\hat{H}_{\text{shift}} = (\bar{\omega}_b - \omega_d - \Delta_b)\hat{b}^\dagger\hat{b} + (\bar{\omega}_a - \frac{\omega_p+\omega_d}{2} - \Delta_a)\hat{a}^\dagger\hat{a}$$
$$\hat{H}_{\text{int}} = g_2^*\hat{a}^2\hat{b}^\dagger + g_2\hat{a}^{\dagger 2}\hat{b} + \epsilon_d\hat{b}^\dagger + \epsilon_d^*\hat{b}.$$

The AC Stark Shift induced by the pump is now given by $\Delta_{a,b}/\hbar = \frac{1}{3}E_J\varphi_{a,b}^2(\mathcal{R}(\xi_a)\varphi_a) + \mathcal{R}(\xi_b)\varphi_b$ and the coupling strength is given by $\hbar g_2 = \frac{1}{2}E_J\epsilon_0\varphi_a^2\varphi_b$.

Crucially, unlike in the case of a single Josephson junction, the only leading order non-rotating term is precisely the desired Hamiltonian, without all the Kerr-like terms that resulted in spurious effects. This led to the first observation of the exponential suppression of the bit-flip error rate with the size of the cat qubit, at the cost

of a linear increase of the phase-flip error rate [83]. More precisely, the parameters achieved in this experiment are $\kappa_1/2\pi = 53$kHz and $\kappa_2/2\pi = 40$kHz, thus the ratio is of the two rates of dissipation is of order 1. This ratio is expected to be greatly improved by using a long-lived 3D cavity for the memory mode. However, because of the absence of Kerr-terms, it was possible to witness the exponential increase of lifetime with respect to bit-flip errors: for each added photon in the cat qubit, the bit-flip time is multiplied by 4.2, up to a point where it saturates around 1ms (300 times longer than the $T_1 = 3\mu s$). In this experiment, the saturation of the bit-flip exponential suppression was caused by the thermal occupation of the transmon used for the Wigner tomography of the cat qubit, which contaminated the memory mode through the dispersive coupling between the transmon and the cavity mode used for the readout.

2.4.2 Zeno Hamiltonians

$Z(\theta)$ gate The use of quantum Zeno dynamics (subsection 2.2.2) to perform bias-preserving rotations around the Z-axis of the cat qubit was demonstrated experimentally in [118], and it is, up to this day, the only gate that has been demonstrated experimentally on the stabilized cat qubit.
The experimental realization followed the proposal [87] that we recalled above. In addition to the (time-independent) two-photon dissipation realized as detailed in subsection 2.4.1, the continuous rotation around the Z axis is triggered by turning on a weak resonant drive at the frequency of the cat qubit mode ω_a.

$ZZ(\theta)$ gate In [87], it was suggested that the same recipe (weak Hamiltonian in presence of the two-photon pumping) could be used to realized two qubit entangling gates.

The Hamiltonian required for the $Z_1Z_2(\theta)$ gate is (subsection 2.2.2) the following "beam-splitter" Hamiltonian

$$\hat{H}_{BS} = \epsilon_{Z_1Z_2}(\hat{a}\hat{b}^\dagger + \hat{b}\hat{a}^\dagger)$$

where $\hat{a}$ and $\hat{b}$ are now both longed lived modes each hosting a cat qubit.

This Hamiltonian was recently realized experimentally [50], using the four-wave mixing capability of the Josephson junction in presence of two pump tones. More precisely, denoting ξ_1, ξ_2 and ω_1, ω_2 the normalized amplitudes and frequencies of the two pumps, respectively, and by $\hat{c}$ the anharmonic mode of the transmon used in the bridge configuration to mediate the coupling between $\hat{a}$ and $\hat{b}$, the Hamiltonian of the system in the displaced frame of the drives and rotating frame of $\hat{a}$, $\hat{b}$ is given

by

$$\hat{H} = -E_J \cos[\varphi_a(\hat{a} + \hat{a}^\dagger) + \varphi_b(\hat{b} + \hat{b}^\dagger) + \varphi_c(\hat{c} + \hat{c}^\dagger + \xi_1 + \xi_1^* + \xi_2 + \xi_2^*).$$

Expanding the cosine and keeping the non-rotating term when the frequency matching condition $\omega_1 - \omega_2 = \omega_a - \omega_b$ is verified produces the required beam-splitter interaction

$$\hat{H}_{\text{int}}/\hbar = g(t)(e^{i\varphi}\hat{a}\hat{b}^\dagger + e^{-i\varphi}\hat{a}^\dagger\hat{b})$$

where φ is determined by the relative phase of the two drives and the coupling coefficient is

$$g(t) = E_J\varphi_a\varphi_b\varphi_c^2\xi_1(t)\xi_2(t) = \sqrt{\chi_{ab}\chi_{bc}}\xi_1(t)\xi_2(t).$$

Note that usually, in these parametric methods, the specific term that one is engineering using wave mixing with appropriate pumps, giving the desired term after an appropriate change of frame and in the limit of the rotating wave approximation, typically result in weak coupling strength. The challenge is then to manage to increase this coupling strength, as in the example of the two-to-one photon conversion Hamiltonian required for the two-photon pumping. Here, however, it is fine that the coupling $g(t)$ is typically weak because it has to be small with respect to the two-photon dissipation in order to implement the Zeno gate with high fidelity.

2.4.3 Topological gates

X gate The realization of the X gate requires to modify the two-photon pumping scheme continuously in time to implement the effective dissipation operator

$$\kappa_2\mathcal{D}[\hat{a}^2 - \exp(2i\pi t/T)\alpha^2]$$

We have seen that the complex number α that parametrizes the two-dimensional cat qubit subspace is given by $\alpha = \sqrt{2\epsilon_d/\kappa_2}$. Thus, the phase of this complex number can be tuned by changing the phase of the resonant drive applied on the buffer ϵ_d between 0 and 2π in a time T. Hence, the realization of the dissipative part of the X gate is actually a straightforward modification of the two-photon pumping scheme already realized experimentally.

In order to remove the phase-flip errors induced by the non-adiabaticity of this variation, one can additionally implement a Hamiltonian of the form $-\Delta\hat{a}^\dagger\hat{a}$ with $\Delta = \pi/T$. This can be done by taking the pump at frequency $2\omega_a - \omega_b - 2\Delta$ instead of $2\omega_a - \omega_b$ and furthermore detuning the drive ϵ_d from resonance by value Δ.

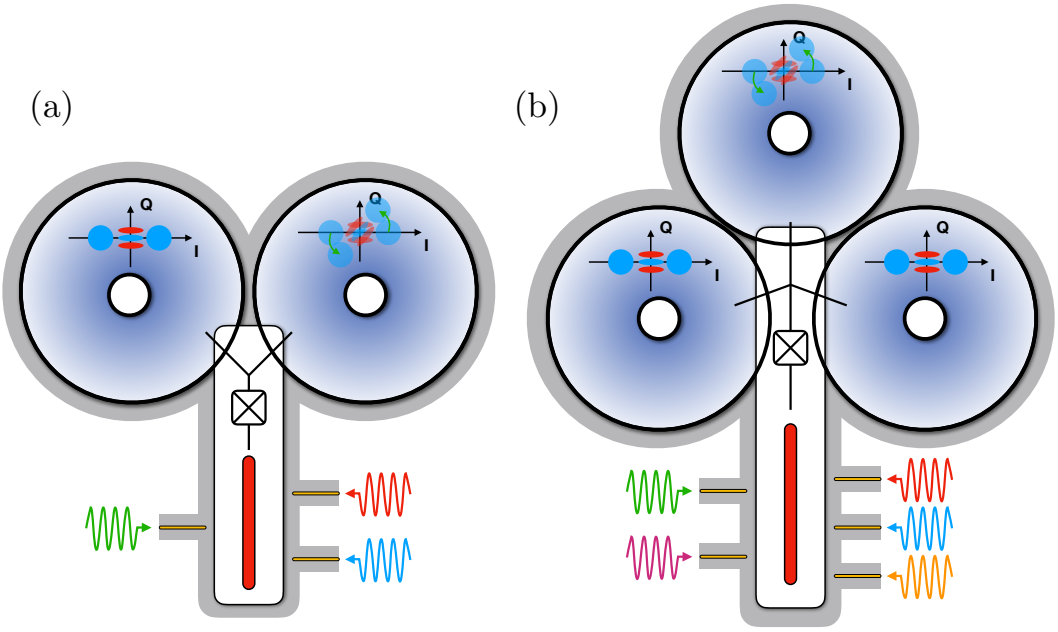


FIGURE 2.8: Proposal for an experimental implementation of bias-preserving CNOT and Toffoli gates for cat qubits using a transmon as the non-linear element. A similar sketch can be drawn to use a fluxed pumped ATS instead as the source of non-linearity. (a) Setup for implementing a bias-preserving CNOT gate. The cat qubits are encoded in high-Q cylindrical post-cavities (in blue, resonance frequencies ω_a and ω_b). The two cavities are coupled via a Y-shape transmon as in [125] to a low-Q stripline resonator (in red, resonance frequency ω_d) playing the role of the buffer mode. The system is driven with three micro-wave pumps at frequencies $\omega_1 = 2\omega_b - \omega_d$, $\omega_2 = (\omega_a - \omega_d)/2$, $\omega_3 = \omega_d$. (b) Similar setup for implementing a bias-preserving Toffoli gate with three cat qubits encoded in high-Q post-cavities (frequencies $\omega_a, \omega_b, \omega_c$) all coupled to a single stripline resonator (frequency ω_d). The system is driven with five micro-wave pumps at frequencies $\omega_1 = 2\omega_c - \omega_d$, $\omega_2 = \omega_a + \omega_b - \omega_d$, $\omega_3 = (\omega_a - \omega_d)/2$, $\omega_4 = (\omega_b - \omega_d)/2$, and $\omega_5 = \omega_d$.

CNOT gate The implementation of a CNOT gate between two cat qubits encoded in storage modes $\hat{a}$ and $\hat{b}$ requires the implementation of the two dissipative channels

$$\kappa_2 \mathcal{D}[\hat{a}^2 - \alpha^2]$$
$$\kappa_2 \mathcal{D}[\hat{b}^2 - \alpha^2 - \frac{\alpha}{2}(1 - e^{\frac{2i\pi t}{T}})(\hat{a} - \alpha)].$$

The implementation of the second one can be realized by coupling the two storage modes $\hat{a}$ and $\hat{b}$ to a buffer mode $\hat{d}$, using a Y-shape transmon similar to [125] or a fluxed pump ATS (see Figure 2.8). Driving the buffer mode at three different frequencies

$$\omega_1 = 2\omega_b - \omega_d$$
$$\omega_2 = \tfrac{1}{2}(\omega_a - \omega_d)$$
$$\omega_3 = \omega_d$$

one can engineer an interaction Hamiltonian of the form

$$\hat{H}_{\text{CNOT}} = (g_{bd}\hat{b}^2\hat{d}^\dagger + g_{bd}^*\hat{b}^{2\dagger}\hat{d}) + (g_{ad}\hat{a}\hat{d}^\dagger + g_{ad}^*\hat{a}^\dagger\hat{d}) + (\epsilon_d\hat{d}^\dagger + \epsilon_d^*\hat{d}).$$

Note that following the experiment [50], the "beam-splitter" conversion triggered by the pump at frequency ω_2 could also be realized by two pumps at frequencies ω_2, ω_2' verifying the matching condition $\omega_2 - \omega_2' = \omega_a - \omega_d$.

In this interaction Hamiltonian, the first term $g_{bd}\hat{b}^2\hat{d}^\dagger + g_{bd}^*\hat{b}^{2\dagger}\hat{d}$ models the exchange of two storage photons at frequency ω_b with one buffer photon at frequency ω_d via a pump photon at frequency ω_1. The second term $g_{ad}\hat{a}\hat{d}^\dagger + g_{ad}^*\hat{a}^\dagger\hat{d}$ models the exchange of one storage photon at frequency ω_a with one buffer photon at frequency ω_d via two pump photons at frequency ω_2. The amplitudes and phases of g_{bd} and g_{ad} are modulated by the amplitude and phase of the corresponding pumps. Finally, the last term $\epsilon_d\hat{d}^\dagger + \epsilon_d^*\hat{d}$ models the resonant interaction of the drive at frequency ω_d with the buffer mode. Similarly to the driven two-photon dissipation, one can adiabatically eliminate the highly dissipative buffer mode to achieve an effective dissipation operator

$$\kappa_2\mathcal{D}[\hat{b}^2 + c_a\hat{a} + c]$$

where the dissipation rate κ_2 is roughly given by $4|g_{bd}|^2/\kappa_d$, κ_d being the loss rate of the buffer mode, the complex constant c_a is given by g_{ad}/g_{bd} and the complex constant c by ϵ_d/g_{bd}. Similarly to the X-operation, it is clear that by varying the amplitudes and phases of the pump at frequency ω_2 and the resonant drive at frequency ω_d, one can engineer a dissipation operator with time-varying constants c_a and c given by

$$c_a(t) = -\frac{\alpha}{2}\left(1 - e^{\frac{2i\pi t}{T}}\right)$$
$$c(t) = -\frac{\alpha^2}{2}\left(1 + e^{\frac{2i\pi t}{T}}\right).$$

This corresponds to the dissipator required for the bias-preserving CNOT operation. Importantly, the time-dependent function c_a takes the value 0 at times $t = 0$ and $t = T$. For this reason, before and after the gate, the two cat qubits involved in the CNOT are defined by their own local oscillators. The fluctuations of the pumps during the execution of the gate merely result in a slight modification of the geometric paths taken. This can only lead to small fluctuations of the geometric phase and therefore an effective phase-flip type error. The phase-flip probability induced by the non-adiabaticity of the evolution can be reduced by adding the effective Hamiltonian

$$\hat{H} = \frac{1}{2}\frac{\pi}{T}\frac{\hat{a} - \alpha}{2\alpha} \otimes (\hat{b}^\dagger\hat{b} - |\alpha|^2) + \text{h.c.}$$

Such a Hamiltonian has also been recently implemented using a detuned parametric pumping method [119].

Toffoli In order to realize a bias-preserving Toffoli gate between three cat qubits encoded in storage modes $\hat{a}$, $\hat{b}$ and $\hat{c}$, further to two-photon driven dissipation on the two control cat qubits, a time-dependent dissipator given by

$$\kappa_2 \mathcal{D}[\hat{c}^2 - \alpha^2 + \frac{1}{4}(1 - e^{\frac{2i\pi t}{T}})(\hat{a}\hat{b} - \alpha(\hat{a} + \hat{b}) + \alpha^2)]$$

is required. Similarly to the CNOT gate, a way to achieve this is to couple the three modes to a highly dissipative buffer mode as shown in Figure 2.8. Driving the buffer mode at five different frequencies $\omega_1 = 2\omega_c - \omega_d$, $\omega_2 = \omega_a + \omega_b - \omega_d$, $\omega_3 = (\omega_a - \omega_d)/2$, $\omega_4 = (\omega_b - \omega_d)/2$, and $\omega_5 = \omega_d$, one can engineer an effective interaction Hamiltonian of the form

$$\begin{aligned} H_{\text{Toffoli}} = (g_{cd}\hat{c}^2\hat{d}^\dagger + g_{cd}^*\hat{c}^\dagger\hat{d}) + (g_{abd}\hat{a}\hat{b}\hat{d}^\dagger + g_{abd}^*\hat{a}^\dagger\hat{b}^\dagger\hat{d}) + (g_{ad}\hat{a}\hat{d}^\dagger + g_{ad}^*\hat{a}^\dagger\hat{d}) \\ +(g_{bd}\hat{b}\hat{d}^\dagger + g_{bd}^*\hat{b}^\dagger\hat{d}) + (\epsilon_d\hat{d}^\dagger + \epsilon_d^*\hat{d}). \end{aligned}$$

Here again, all these effective terms are achieved in a parametric manner and using the 4-wave mixing property of the Josephson junction. The amplitude and phase of each interaction term can be modulated by the amplitude and phase of the associated pump. After the adiabatic elimination of the buffer mode, we achieve a dissipation operator

$$\kappa_2 \mathcal{D}[\hat{c}^2 + c_{ab}\hat{a}\hat{b} + c_a\hat{a} + c_b\hat{b} + c],$$

where κ_2 is given by $4g_{cd}^2/\kappa_d$, and the complex constants $c_{ab} = g_{abd}/g_{cd}$, $c_a = g_{ad}/g_{cd}$, $c_b = g_{bd}/g_{cd}$, $c = \epsilon_d/g_{cd}$. By varying the amplitudes and phases of the pumps in time, we obtain time-varying constants

$$\begin{aligned} c_{ab}(t) &= \frac{1}{4}\left(1 - e^{\frac{2i\pi t}{T}}\right) \\ c(t) &= -\frac{\alpha^2}{4}\left(3 + e^{\frac{2i\pi t}{T}}\right) \\ c_a(t) = c_b(t) &= -\frac{\alpha}{4}\left(1 - e^{\frac{2i\pi t}{T}}\right). \end{aligned}$$

This implements a bias-preserving Toffoli gate between the cat qubits encoded in the three modes $\hat{a}, \hat{b}$ and $\hat{c}$. Here again, it should be noted that the functions c_{ab}, c_a, c_b vanish at the beginning and at the end of the gate execution, so that each cat qubit gets back to being defined by its own local oscillators. Similarly to the CNOT gate, the pump fluctuations during the gate only result in a slight increase in the rate of phase-flip type errors, but do not lead to unsuppressed bit-flip type ones. In order to reduce the phase-flip probability induced by the non-adiabaticity, we use an additional Hamitonian

$$\hat{H} = -\frac{1}{2}\frac{\pi}{T}\frac{\hat{a} - \alpha}{2\alpha} \otimes \frac{\hat{b} - \alpha}{2\alpha} \otimes (\hat{c}^\dagger\hat{c} - |\alpha|^2) + \text{h.c.}$$

This Hamiltonian is undoubtedly the most complicated parametric Hamiltonian to realize in our scheme. While, in theory, it could be realized as previously by using a high-order multi-wave mixing with appropriate pumps, this approach would produce the required term with a very small amplitude. Instead, a better strategy to engineer this Hamiltonian could be to use a cascade of low-order multi-wave mixing such as the ones realized [88, 89]. However, due to the complexity of engineering this higher order Hamiltonian, perhaps the first generation of Toffoli gates could be implemented without this improvement.

Chapter 3

Fault-tolerant logical gates on repetition cat qubits

The focus of the precedent Chapter is the design of physical operations on the cat qubit preserving the exponentially large noise asymmetry. In this Chapter, we detail how cat qubits can be used in a repetition code protecting against the dominant phase-flip errors to produce a logical qubit, called the *repetition cat qubit*, with very low logical phase-flip and logical bit-flip error rates. Furthermore, we demonstrate that the set of bias-preserving operations of the precedent Chapter is sufficient to build a *universal* and *fault-tolerant* set of logical encoded gates for the repetition cat qubit, making this qubit a promising candidate to perform large scale quantum computations.

3.1 Fault-tolerant and universal construction

3.1.1 Repetition cat qubit

The effect of a phase-flip error, *i.e* the uncontrolled application of a Pauli Z operator, is to swap the eigenstates of the Pauli X operator $|\pm\rangle$. For the cat qubit, this is a swap of the two cat states $|\mathcal{C}_\alpha^\pm\rangle$

$$Z|\mathcal{C}_\alpha^\pm\rangle = |\mathcal{C}_\alpha^\mp\rangle.$$

Hence, the repetition code protecting against phase-flips is defined in the dual basis by repeating these fragile states. For a distance d repetition cat qubit, encoding a single logical qubit using d physical cat qubits, the logical $|+\rangle_L$ and $|-\rangle_L$ states are given by $|\pm\rangle_L := |\pm\rangle^{\otimes d} = |\mathcal{C}_\alpha^\pm\rangle^{\otimes d}$. The layout of an experimental proposal to implement a single repetition cat qubit is depicted in Figure 3.2.

Using the stabilizer formalism, the code space is defined as the $+1$ common eigenspace of the $d-1$ stabilizers

$$S_i = X_i X_{i+1}$$

with $i \in \mathbb{Z}_{d-1}$.

The prerequisite to the implementation of the repetition code that one needs to check is whether these stabilizers can be measured using the operations on the cat qubit of the previous Chapter. The quantum non demolition (QND) measurement of these operators can be implemented using an ancilla cat qubit prepared in the $|+\rangle$ state, the bias-preserving CNOT gate and the measurement of the X operator on the ancilla cat qubit, as depicted in Figure 3.1.

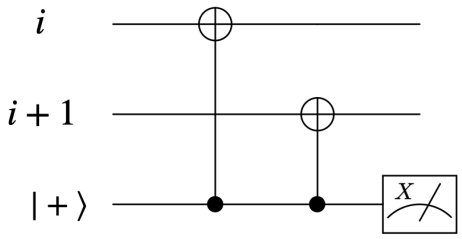


FIGURE 3.1: Joint-parity measurement $X_i X_{i+1}$ between two neighboring cat qubits i and $i+1$ of a repetition cat qubit, using one ancilla. Note that the error propagation from the ancilla cat qubit to the data ones, of the bit-flip type, is exponentially suppressed with the size of the cat.

A choice of logical Pauli operators for the repetition cat qubit of minimal weight is

$$\begin{aligned} X_L &= X_0 \\ Z_L &= \bigotimes_{i \in \mathbb{Z}_d} Z_i \\ Y_L &= i X_L Z_L. \end{aligned}$$

Note that the minimal choice $X_L = X_0$ is not unique, as actually any physical Pauli X_i operator acting on a cat qubit is implementing a logical Pauli X_L operator on the cat qubit. This reflects the fact that the repetition code protecting against phase-flips cannot detect or correct any physical bit-flip.

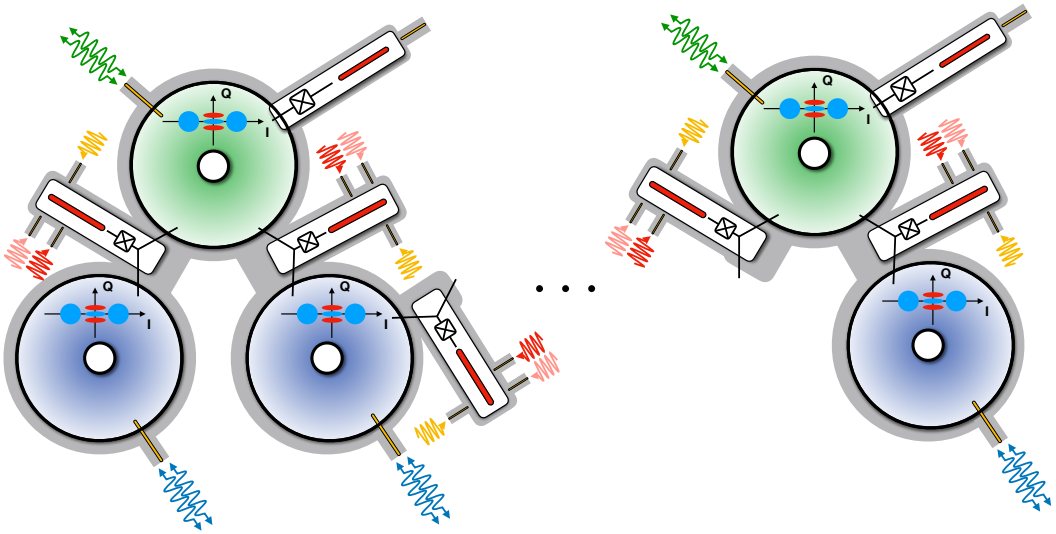


FIGURE 3.2: Lay-out of a repetition cat qubit using high-Q 3D cylindrical post-cavities [106]. Each data cat qubit (in blue cavities) is connected to a pair of ancilla cat qubits (in green cavities) for the repetition code stabilizer measurement. The results of the parity measurement are read out using the low-Q stripline resonators (in red) coupled to green cavities. Each cat qubit is continuously driven via the two-photon driven-dissipative scheme (arrows). The couplings between cavity modes are mediated by a Josephson circuit and extra microwave drives, required for bias-preserving CNOT operations as detailed in Chapter 2 and Figure 2.8. The choice of Y-shape transmons is optional, and they could be replaced by flux pumped ATS. The choice of cylindrical post-cavities is to ensure high quality factors, but a similar layout could be thought of for a 2D architecture.

The logical computational states are given by

$$|0\rangle_L = \frac{1}{(\sqrt{2})^{d-1}} \sum_{j\in\{0,1\}^d, |j| \text{ even}} |j\rangle$$
$$|1\rangle_L = \frac{1}{(\sqrt{2})^{d-1}} \sum_{j\in\{0,1\}^d, |j| \text{ odd}} |j\rangle$$

where j is a d-bit string and $|j|$ denotes the number of ones, called the *Hamming weight*, of the string j. Recalling that $|1\rangle \approx |-\alpha\rangle$, one can note that the logical information is encoded in the parity of the number of cat qubits in the state $|-\alpha\rangle$.

For a fixed value of the mean number of photons in each of the physical cat qubits, hence a fixed value of the physical noise bias $\eta = p_Z/p_X$, the distance d of the repetition code is chosen such that the probabilities p_{Z_L} of logical phase-flip, and p_{X_L} of logical bit-flip errors, are of comparable strength, thus constructing a repetition cat qubit suffering from an effective *unbiased* noise of probability $\epsilon_L \leq p_{X_L} + p_{Z_L}$.

Before we move on to the construction of logical gates, an important remark is in order about the general philosophy behind our approach. Indeed, the repetition cat qubit approach relies on two different kinds of protection. The two-photon dissipation exponentially suppresses bit-flip errors with the mean number of

photons in the cat state, while the rate of phase-flip errors increases only linearly. Then, the repetition code suppresses exponentially the phase-flip errors, provided that the phase-flip error rate of the cat qubit is below the fault-tolerance threshold of the repetition code.

This protection is similar to the one achieved by Bacon-Shor codes [111, 14], with the nice feature that the "distance" of the inner protection provided by the two-photon pumping can be increased without any further hardware overhead. However, similarly to Bacon-Shor codes, because the phase-flip error rate of the cat qubit increases linearly with the mean number of photons, there cannot be a threshold since the effective phase-flip error rate of the cat qubit will eventually exceed the threshold of the repetition code. Nonetheless, this is not an obstacle to obtaining extremely low logical error rates with this approach, by limiting the mean number of photons to a finite value for which the bit-flip error probability is extremely low, and for which the phase-flip error probability is still below the threshold of the repetition code. In our analysis, we limit the size of the cat qubits to $\bar{n} = 15$ photons and assume that the exponential suppression of the bit-flip error rate holds at least up to this cat size.

3.1.2 Quantum computing against biased noise

The idea to use a repetition code against the dominant error of a biased noise qubit has been employed as a first level of encoding in [8, 129]. In these papers, the quantum information is protected by a concatenation of two codes $\mathcal{C}_1 \triangleright \mathcal{C}_2$, where $\triangleright$ denotes code concatenation. The code $\mathcal{C}_1$ is a distance d repetition code that protects against phase-flips errors, leading to a logical qubit that suffers from an effective unbiased noise of strength $p_{L,1}$. As soon as $p_{L,1}$ is below the threshold of $\mathcal{C}_2$, arbitrarily low logical error rates $p_{L,1}$ can be achieved by the concatenated $\mathcal{C}_1 \triangleright \mathcal{C}_2$ code.

A major difference with the cat qubit based scheme that we build in this work and the previous studies investigating the performance of computing against biased noise with *regular two-level qubits* [8, 129] is that in these works, the set of physical bias-preserving gates that can be implemented are not sufficient to build a set of logical gates at the repetition code $\mathcal{C}_1$ level, because the important topological gates X, CX, and CCX are missing at the physical level. Thus, in these works, the second code $\mathcal{C}_2$ is required even if the base qubits do not suffer from bit-flip errors at all, that is in the limit of an infinite noise bias where the logical error rate $p_{L,1}$ of the repetition code $\mathcal{C}_1$ can be made arbitrarily low.

More precisely, in [8], the set of fundamental bias-preserving operations $\mathcal{S}$ is limited to

$$\mathcal{S} = \{\mathcal{P}_{|+\rangle}, \mathcal{M}_X, \mathrm{CZ}\}$$

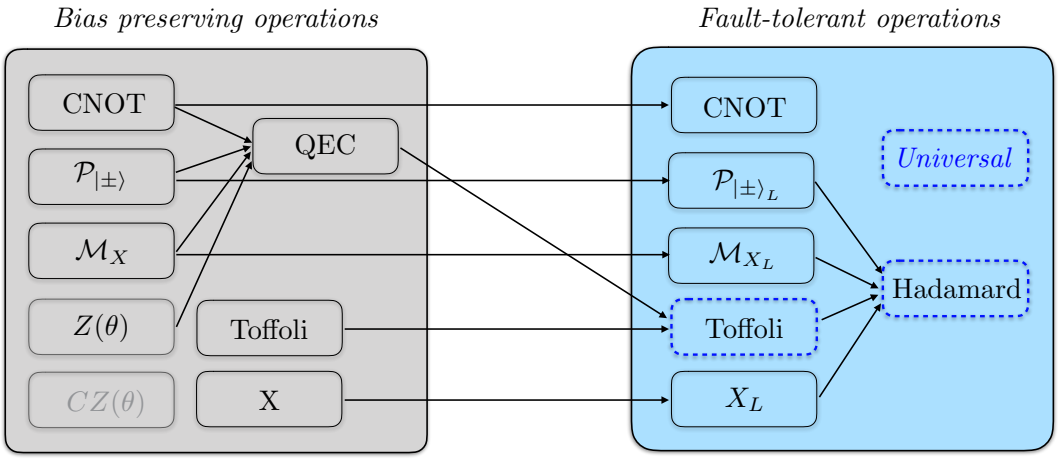


FIGURE 3.3: Overall scheme for achieving fault-tolerant universal quantum computation using repetition cat qubits. The fundamental operations inside the left-hand side box are performed on the cat qubits, in a bias-preserving manner. Fault-tolerant logical operations acting on the repetition cat qubits (right-hand side) are built out of these operations, as depicted by the arrows.

At the $\mathcal{C}_1 \triangleright \mathcal{C}_2$-logical level, the full set of Clifford gates is achieved by adding the preparation of the "magic state"

$$|+i\rangle_L = \tfrac{1}{\sqrt{2}}(|0\rangle_L + i|1\rangle_L)$$

and is made universal by appending the preparation of yet another magic state

$$|T\rangle_L = \tfrac{1}{\sqrt{2}}(|0\rangle_L + e^{i\pi/4}|1\rangle_L).$$

In [129], with the addition of $\mathrm{CZ}(\theta)$ to the set of bias-preserving operations, the authors construct new gadgets to reduce the overhead for magic state preparation and distillation.

3.1.3 Universal set of logical operations

We present the big picture for the construction of fault-tolerant gates at the level of the repetition cat qubits in Figure 3.3. We have seen in Chapter 2 that the set of all the bias-preserving operations that can be performed on cat qubits is

$$\mathcal{S}' = \{\mathcal{P}_{|+\rangle}, \mathcal{P}_{|0\rangle}, \mathcal{M}_X, \mathcal{M}_Z, Z(\theta), ZZ(\theta), ZZZ(\theta), X, \mathrm{CX}, \mathrm{SWAP}, \mathrm{CCX}\}.$$

The next step is to build fault-tolerant encoded operations at the level of the repetition code, using exclusively operations from this fundamental set. Actually, not all gates in this set are involved in the construction of the universal set of logical operations. The minimal set of bias-preserving operations required in the logical construction is

$$\mathcal{S} = \{\mathcal{P}_{|+\rangle}, \mathcal{P}_{|0\rangle}, \mathcal{M}_X, Z, X, \mathrm{CX}, \mathrm{CCX}\}.$$

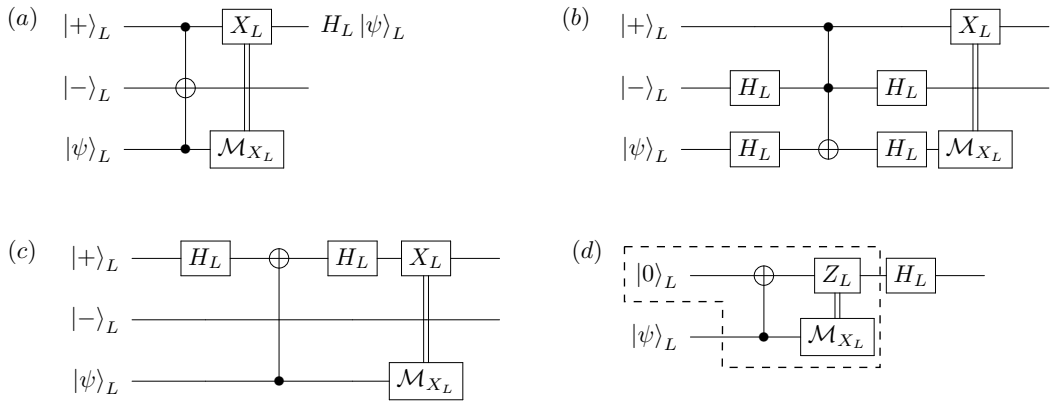


FIGURE 3.4: Logical Hadamard circuits (a). The circuits (b,c,d) are equivalent circuits that help to understand why the circuit (a) implements a logical Hadamard gate. In circuit (b), the "CNOT" gate between the repetition cat qubit 2 and 3 is replaced by the equivalent circuit where the control and target roles are switched with the addition of Hadamard gates before and after the gate. It becomes clear in circuit (b) that the second repetition cat qubit plays no role: it is initialized in $|-\rangle_L$, transformed to $|1\rangle_L$ after the first Hadamard, thus always triggers the corresponding part of the control of the Toffoli, before being converted to $|-\rangle_L$ by the second Hadamard. Actually, the second repetition cat qubit is needed only because we do not readily have a controlled-phase gate available at the repetition cat qubit level (but here we show that it can be done with the Toffoli gate plus one ancilla repetition cat qubit). The idle role of the second repetition cat qubit is represented more simply in circuit (c), where again the "CNOT" between the repetition cat qubit 1 and 3 is replaced by its equivalent circuit where the control and target roles are exchanged with the addition of Hadamard gates. Finally, in circuit (d) (where the second repetition cat qubit is omitted for clarity) the first Hadamard of the first line and the preparation of $|+\rangle_L$ are replaced by the equivalent preparation of $|0\rangle_L$ and the second Hadamard is commuted through the X_L, producing a Z_L gate. The remaining circuit in the dashed box is just a teleportation of the state $|\psi\rangle_L$ of the second repetition cat qubit to the first one. After the state has been teleported, the remaining Hadamard gate H_L is applied, thus establishing the equivalence between circuit (a) and a logical Hadamard.

The set of fault-tolerant encoded operations on repetition cat qubits is given by

$$\mathcal{S}'_L = \{\mathcal{P}_{|+\rangle_L}, \mathcal{P}_{|0\rangle_L}, \mathcal{M}_{X_L}, \mathcal{M}_{Z_L}, Z_L, \mathrm{CZ}_L, X_L, \mathrm{CX}_L, \mathrm{CCX}_L\},$$

and the smallest universal set of operations contained in $\mathcal{S}'_L$, which can be built using only operations in the minimal set of bias-preserving operations $\mathcal{S}$, is

$$\mathcal{S}_L = \{\mathcal{P}_{|+\rangle_L}, \mathcal{P}_{|0\rangle_L}, \mathcal{M}_{X_L}, Z_L, X_L, \mathrm{CCX}_L\}.$$

As will become clear in the next section, the logical CNOT gate can be implemented transversally on the repetition code, leading to great hardware simplifications compared to the logical CNOT gate construction of [8]. The universality of $\mathcal{S}_L$ is established by the fact that it contains the Toffoli gate in the computational basis together with state preparation and measurement in the dual basis. Indeed, as depicted in Figure 3.4, a logical Hadamard gate can be built out of the gates of the set

$\mathcal{S}_L$, which, together with the Toffoli gate, is universal [109]. Note that in the circuit of figure 3.4, the Toffoli gate and the logical target ancilla are used solely to implement a logical CZ_L gate. However, as will become clear in section 3.3, the logical CZ_L is actually easier to implement than the Toffoli gate such that in practice, it will be used instead of the Toffoli to implement a Hadamard gate.

3.2 Fault-tolerance via transversal construction

Let us recall the definition of a *transversal* logical operator as given by Eastin and Knill in [37]:

Definition 1 (Transversal operator) *We label as "transversal" any partition of the physical subsystems of a code such that each part contains one subsystem from each code block. Given a transversal partition of a code, an operator is called transversal if it exclusively couples subsystems within the same part. Put another way, an operator is transversal if it couples no subsystem of a code block to any but the corresponding subsystem in another code block.*

Very often, the partition of the code considered is the natural one given by the physical qubits. The transversal circuit implementing a transversal operator is fault-tolerant by construction, in the sense that they can increase the total number of locations where errors can arise (by propagation through the circuit) but the total number of errors necessary to cause a logical error is unchanged, and that the depth of the circuit (the irreducible number of time steps) is constant and does not depend on the code distance d. We quickly review the operations that can be performed transversally on the repetition code.

Preparation of $|\pm\rangle_L$ The preparation of $|\pm\rangle_L = |\pm\rangle^{\otimes d}$ can be performed transversally from the preparation of $|\pm\rangle$

$$\mathcal{P}_{|\pm\rangle_L} = (\mathcal{P}_{|\pm\rangle})^{\otimes d}.$$

followed by a quantum error correction step to reduce the preparation infidelity according to the distance of the repetition code.

Preparation of $|0\rangle_L$ In our scheme, the preparation of the state $|0\rangle_L = \frac{1}{\sqrt{2}}(|+\rangle^{\otimes d} + |-\rangle^{\otimes d})$ is needed to perform the Steane error correction step in the implementation of the logical Toffoli gate that we describe in subsection 3.3.2. This state could be prepared from the logical $|+\rangle_L$ state by applying a logical Hadamard gate as depicted in Figure 3.4. However, we have seen that the logical Hadamard gate is not transversal but rather, involves at least one additional ancilla repetition cat qubit, and a logical CZ_L, such that this way of preparing the state is quite complicated. A more simple preparation protocol consists in first preparing

the state $|0\rangle^{\otimes d} \approx |\alpha\rangle^{\otimes d}$ from the transversal preparation of all the cat qubits in the state $|\alpha\rangle$, and performing a round of error correction. This protocol works because the state $|0\rangle^{\otimes d}$ can be written as the state $|0\rangle_L$ on which a superposition of 2^{d-1} (correctable) errors are acting. Each of these 2^{d-1} is uniquely identified by a syndrome, such that the round of error correction projects the state on one of these errors. After the corresponding correction is applied, the state $|0\rangle_L$ is recovered.

Indeed, the state $|0\rangle^{\otimes d}$ can be expanded as

$$|0\rangle^{\otimes d} = \frac{1}{(\sqrt{2})^d}(I+Z)^{\otimes d}|+\rangle^{\otimes d} = \frac{1}{(\sqrt{2})^d}\Big[\sum_{j\in\{0,1\}^{\otimes d}} \bigotimes_{i\in\mathbb{Z}_{d-1}} Z_i^{j_i}\Big]|+\rangle_L$$

Factorizing in the above sum the terms for which the Hamming weight of the bit-string $|j|$ (the number of bits j_i equal to one) is superior to $(d+1)/2$ as $\bigotimes_{i\in\mathbb{Z}_{d-1}} Z_i^{j_i} = Z_L \bigotimes_{i\in\mathbb{Z}_{d-1}} Z_i^{1\oplus j_i}$, the state $|0\rangle^{\otimes d}$ writes

$$|0\rangle^{\otimes d} = \frac{1}{(\sqrt{2})^{d-1}}\Big[\sum_{\substack{j\in\{0,1\}^{\otimes d}\\ |j|\leq\frac{d+1}{2}}} \bigotimes_{i\in\mathbb{Z}_{d-1}} Z_i^{j_i}\Big]|0\rangle_L = \frac{1}{(\sqrt{2})^{d-1}}\Big[\sum_{\substack{j\in\{0,1\}^{\otimes d}\\ |j|\leq\frac{d+1}{2}}} E_j\Big]|0\rangle_L.$$

The measurement outcomes of the $d-1$ stabilizers $\{X_i\otimes X_{i+1}\}_{i\in\mathbb{Z}_{d-1}}$ on the state $|\alpha\rangle^{\otimes d}$ gives a random outcome s^j (obtained with probability $1/2^{d-1}$), and the state after the measurement is projected on $E_j|0\rangle_L$. Because E_j is a Pauli operator, the preparation of the state $|0\rangle_L$ is achieved by applying the correction E_j.

Measurement of X_L Similarly, the measurement of only one cat qubit from the repetition cat qubit using $\mathcal{M}_{X_i}$ already implements a measurement of X_L. However, to ensure fault-tolerance, $\mathcal{M}_{X_L}$ is implemented by measuring $\mathcal{M}_X$ on all the cat qubits and applying a majority vote to the measurement outcomes.

Logical CNOT gate This gate is not required in our set of universal gates and can be suppressed from the set $\mathcal{S}_L$. However, its implementation is easy and can lead to more economical circuits for realization of certain algorithms. Indeed, the logical CNOT gate is simply obtained from the physical one by performing d CNOT gates in a transversal manner, as depicted in Figure 3.5.

Before we move on to the construction of the logical Toffoli gate, which achieves universality, let us point out for the repetition code, not all Clifford gates are transversal. For instance, a logical encoded version of the controlled-Z gate CZ is not readily implemented by the transversal application of a single round of d physical CZ gates, but rather, like the Toffoli gate (see section 3.3), requires that d^2 CZ gates be applied in an all-to-all fashion. We discuss how such a circuit can be made fault-tolerant in subsection 3.3.3.

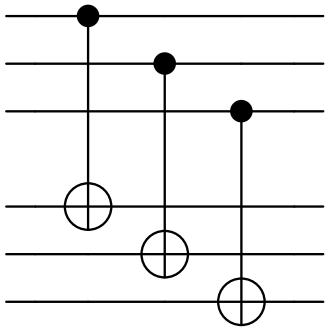


FIGURE 3.5: Transversal implementation of the logical CNOT gate on distance $d = 3$ repetition cat qubits. Here the three upper lines represent the control repetition cat qubit and the three bottom ones the target one.

3.3 Fighting the curse of the Eastin-Knill theorem: non-transversal fault-tolerance

In this section, we investigate two different strategies to build a fault-tolerant encoded version of the Toffoli gate from the physical, bias-preserving Toffoli gate. A clear consequence of the Eastin-Knill theorem is that a circuit implementing an encoded version of the Toffoli gate on the repetition code cannot be transversal. Indeed, we have seen in the previous section that all the other operations in the logical Hadamard circuit of Figure 3.4 (a), thus a transversal circuit implementing a logical Toffoli would achieve a universal and transversal gate set on the repetition code. Rather, the logical Toffoli circuit on (the distance d) repetition code requires d^2 Toffoli gate, where the d^2 factor comes from the fact that the Toffoli gate involves some Z_1Z_2 interaction between the two controls and that a (minimal) logical Z_L operator on the code $\mathcal{C}_1$ is given by the product of all the physical Z operators. Thus, a logical Toffoli circuit necessarily contains a physical Toffoli gate coupling all the qubits of the first control block to all the qubits of the second control block. Note that, for each physical Toffoli, the qubit chosen to be the target in the target block actually does not matter, given that $X_L = X_i$ for all i). This degree of freedom in the choice of the target qubit enables us to build a circuit of d blocks, called *pieces*, of d Toffoli gates applied transversally, producing a depth d circuit rather than d^2. An example of such a circuit is depicted in Figure 3.6

Mathematically, the logical Toffoli gate is implemented on the repetition code using the "round-robin" construction [130]

$$\mathrm{CCX}_L = \prod_{i,j\in\mathbb{Z}_d} \mathrm{CCX}(i,j,k(i,j))$$

where $\mathrm{CCX}(i,j,k)$ denotes a physical Toffoli gate between the i-th qubit of the first control block, the j-th qubit of the second control block, and the k-th qubit of target block. Note that $k(i,j)$ can actually be any mapping $\mathbb{Z}_d \times \mathbb{Z}_d \to \mathbb{Z}_d$, since

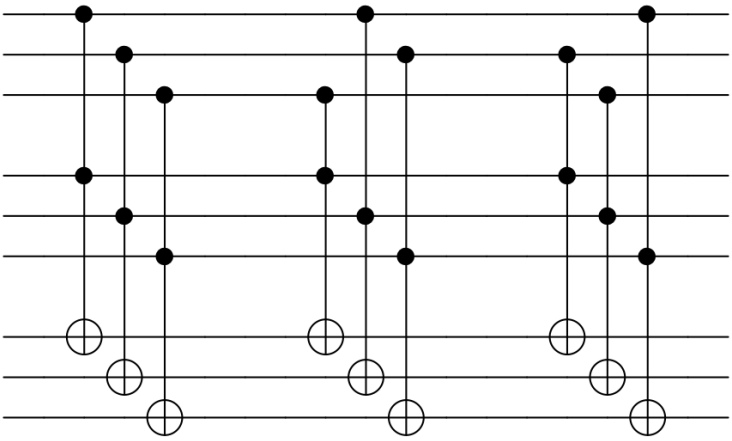


FIGURE 3.6: Example of a circuit implementing a logical encoded Toffoli gate on distance $d = 3$ repetition cat qubits, using $d^2 = 9$ physical Toffoli gates. The depth of this circuit is 3 and supposes that any triplet of physical qubits can be coupled by a Toffoli gate.

the gates $\mathrm{CCX}(i, j, k_1)$ and $\mathrm{CCX}(i, j, k_2)$ act identically on the code spaces of the three logical qubits, and we chose $k(i, j) = j$ for the rest of this manuscript. In the next two subsections, we detail the reasons why the raw circuit of Figure 3.6 is not fault-tolerant and we propose two different schemes to fix this issues. In the next Chapter, we present a third scheme to implement a fault-tolerant Toffoli gate in a very different manner, by fault-tolerantly preparing a magic state associated to this gate and injecting it into the code.

The first scheme (subsection 3.3.1) achieves fault-tolerance by using concatenation [130]. The idea of code concatenation is to build a hierarchy of codes within codes iteratively, by replacing all the physical gates in a logical circuit by their logical versions (see e.g [91]). We argue in Chapter 4 that with experimentally reasonable physical error rates, the logical error rate of the Toffoli circuit is well below the *pseudo-threshold*, *i.e* the logical error probability of the logical encoded Toffoli gate is lower than the physical error probability of the Toffoli gate, thus making it possible to use code concatenation.

The second scheme (subsection 3.3.2) avoids concatenation and achieves a higher phase-flip threshold and an improved scaling, but comes at the expense of a more complex error correction circuit based on three ingredients. First, the accumulation of non-propagating errors is prevented using the pieceable fault-tolerant protocol described in [130, 60]. Second, this pieceable fault-tolerant protocol requires the measurement of the stabilizers of the code in the middle of the logical Toffoli circuit, at a point where these stabilizers are no longer Pauli operators, but have evolved to Clifford operators under the action of the non-Clifford pieces of the Toffoli circuit. Last, Steane-style error detection [113] decoded with a majority vote on the

target block is required instead of the usual stabilizer measurements decoded with the MWPM decoder used everywhere else in this work.

3.3.1 Fault-tolerant Toffoli circuit: scheme 1

There are two reasons the circuit of Figure 3.6 is not fault-tolerant as such.

Propagation of errors: a consequence of non-transversality First, because the circuit of Figure 3.6 is not transversal and, by construction, any qubit of the target block is connected to *all* the qubits of the first control block, a single Z error acting on a qubit of the target block can propagate to many different qubits within the same control block, eventually leading to a logical failure, with a probability that is not exponentially suppressed with the code distance. For example, a Z error occurring on a qubit of the target block before the circuit is executed propagates to the same Z error on the target block, plus a logical CZ gate between the two logical control qubits. This first problem can be solved following the pieceable fault-tolerant method of [130]. More precisely, we split the circuit containing d^2 physical Toffoli gates into d transversal pieces of d Toffoli gates each. Choosing a circular permutation on the qubit of the first control block, the k-th piece of the circuit can be written

$$P_k = \prod_{i=0}^{d-1} \mathrm{CCX}(i-k+1, i, i)$$

where all the indices are taken modulo the distance d of the repetition code. Between two pieces, a round of error correction is inserted on the target block to catch errors before they spread to the control blocks, as depicted in Figure 3.7. Importantly, because the target X operator commutes with the CCX gate, the stabilizers of the target block $\{X_i X_{i+1}, i \in \mathbb{Z}_{d-1}\}$ are left unchanged by the CCX gates of the circuit and can be measured at any point in the circuit with the circuit of Figure 3.1.

The logical Toffoli circuit is executed as follows: after each of the first $d-1$ transversal pieces of CCX gates, a single round of stabilizer measurement is performed on the target block. After each of these pieces, say the k-th one, the k outcomes from all the previous measurement rounds are decoded together using a minimum weight matching decoder (see Chapter 4). The corresponding correction is applied before the $(k+1)$-th piece of the circuit is executed. After the last piece is executed, the usual error correction, composed of d rounds of stabilizer measurements and correction, is performed on the three code blocks. The fact that a single round of stabilizer measurement is enough during the intermediate error correcting steps can be intriguing. Indeed, since the measurements themselves are faulty, they usually need to be repeated a certain number of times, that scales linearly with the code distance, before the outcome of the first measurement, decoded together with the ones following, can be trusted. Here, a single round is executed independent of the code size, but is decoded using the full history of the previous measurement

outcomes. The history of the Z correction applied on the target block is also kept in memory. Thus, after all the d pieces have been executed, a final decoding on all d syndrome outcomes is performed and it becomes possible to know *a posteriori* which target Z errors have propagated to CZ errors between control qubits and to correct the corresponding CZ before the final QEC round on the control blocks. This is possible because the Z corrections performed on the target block anti-commute with the constant stabilizers of the target block, thus travel without being projected and can be undone later if needed.

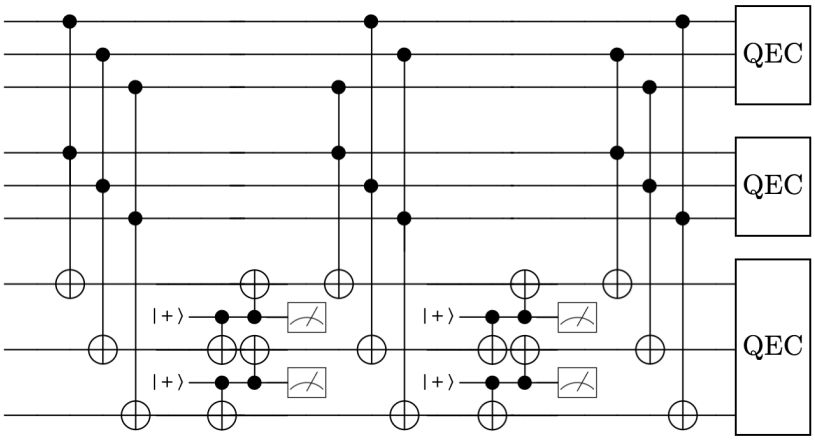


FIGURE 3.7: Logical Toffoli circuit for distance 3 repetition cat qubits. After each round of transversal Toffoli gates, a single round of stabilizer measurement is performed on the target block. The outcome of the measurement is decoded together with the history of all outcomes and an appropriate correction is applied. After the circuit, a full error correction stage is performed on all three blocks.

Accumulation of non-propagating errors: a consequence of increasing circuit depth The second reason the circuit is not fault-tolerant is because of the accumulation of non-propagating errors on the control qubits. Indeed, as a consequence of the fact that the circuit depth increases linearly with the code distance d, each qubit of the two control blocks undergo d gates without the stabilizers of these qubits being measured. Therefore, without further considerations, when increasing the code distance d, the probability of Z errors on these control blocks increases and eventually exceeds the fault-tolerance threshold of the repetition code. One way to handle this problem is by concatenation with another repetition code. The existence of a reasonable *pseudo-threshold*, *i.e* a physical error probability for which one can find a code distance d yielding a lower logical error probability proves the existence of a concatenation phase-flip threshold. We postpone the precise analysis of the value of this pseudo-threshold to the numerical study of the circuits of Chapter 4. Note that concatenating a distance d repetition code with itself produces a repetition code of distance d^2. The circuit implementing a logical Toffoli on the concatenation of these two codes is thus very similar to the one depicted in Figure 3.7, except that it now includes error correcting steps on the control blocks every d steps. The distance can

be further increased by raising the number of levels of concatenation (the concatenation of k repetition code produces a distance d^k repetition code), while the number of steps between two rounds of error correction remains constant and equal to d.

3.3.2 Fault-tolerant Toffoli circuit: scheme 2

For the circuit of Figure 3.7 to exhibit a phase-flip threshold without any concatenation, it is necessary to place additional rounds of error correction on the control blocks in such a way that the number of time-steps between two rounds of error detection does not increase with the code distance. Ideally, we would like to perform a round of error correction on all three logical blocks after each of the transversal pieces of the circuit. This task is complicated by the fact that the stabilizers of the control blocks are not constant throughout the circuit. We label the three logical qubits A, B and C, where C is the logical target block and denote by X_i^A the X Pauli operator acting on the i-th physical qubit of block A, where all subscripts are taken modulo the code distance d.

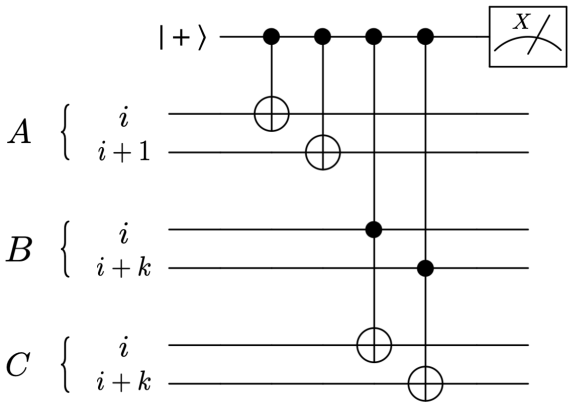


FIGURE 3.8: Measurement circuit of the Clifford stabilizer $X_i^A X_{i+1}^A \mathrm{CX}^{B,C}(i,i)\mathrm{CX}^{B,C}(i+k,i+k)$.

Let us have a look at the value of the non-constant stabilizers of the two control code blocks $X_i^O X_{i+1}^O$, $O \in \{A,B\}$, $i \in [\![0,d-1]\!]$, after k pieces of the circuit have been executed. Noting

$$U_k = \prod_{j \in [\![1,k]\!]} P_j$$

the stabilizers of the two controls blocks A and B become under conjugation by this unitary

$$S_{i,k}^A = U_k X_i^A X_{i+1}^A U_k^\dagger = X_i^A X_{i+1}^A \mathrm{CX}^{B,C}(i,i)\mathrm{CX}^{B,C}(i+k,i+k)$$
$$S_{i,k}^B = U_k X_i^B X_{i+1}^B U_k^\dagger = X_i^B X_{i+1}^B \times \prod_{j=0}^{k-1} \mathrm{CX}^{A,C}(i-j,i)\mathrm{CX}^{A,C}(i+1-j,i+1)$$

where $\mathrm{CX}^{R,S}(i,j)$ denotes the CX gate between the i-th qubit of block R acting as the control and the j-th qubit of block S acting as the target. Note that the stabilizers of the control block C are constant throughout this evolution

$$S^C_{i,k} = U_k X^C_i X^C_{i+1} U_k^\dagger = X^C_i X^C_{i+1}.$$

The evolution of the stabilizers of the control blocks leads to a few issues that need to be handled carefully, to ensure the existence of a phase-flip threshold.

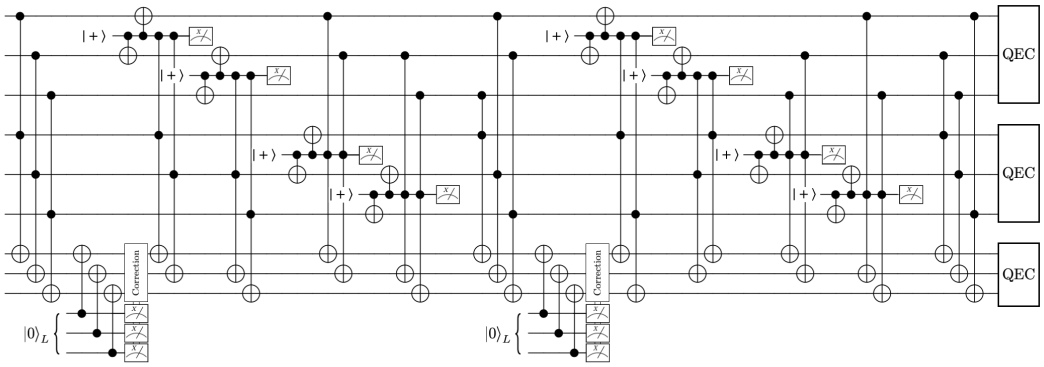


FIGURE 3.9: A fault-tolerant Toffoli circuit without concatenation. After each of the first $d-1$ pieces of the circuit (here, $d=3$), a round of Steane error correction is performed on the target block, followed by the measurement of the Clifford stabilizers on the control blocks.

First, the unitary U_k does not belong to the Clifford group, but to the third level of the Clifford hierarchy [55] (see Appendix B). It maps the Pauli stabilizers of the control blocks to Clifford operators. Nevertheless, these Clifford stabilizers can be measured using CCX gates and Clifford gates. The stabilizer $S^A_{i,k}$ can be measured in the standard way using one ancilla qubit with the circuit depicted in Figure 3.8. Importantly, the measurement of the non-constant stabilizers is bias-preserving as the CX and CCX gates possess this property.

Second, the weight of the stabilizers of control block A, $S^A_{i,k}$, is constant at all intermediate steps of the circuit. Unfortunately, this is not the case for the stabilizers $S^B_{i,k}$ of control block B, whose weights grow linearly with the number of pieces k. The asymmetry between the two control blocks is a consequence of the particular choice of ordering for the physical Toffoli gates. A symmetric ordering causes the weight of the stabilizers of both logical blocks to grow linearly with the code distance, but unfortunately it is not possible to order the gates such that the weights of all stabilizers be bounded by a constant. This implies that an increasing depth-k circuit might be needed to measure these stabilizers in the same fashion as in Figure 3.8. This scaling of the measurement time with the code distance d prevents the existence of a phase-flip threshold.

A possible solution to this problem is to measure a different set of Clifford observables of constant weight instead of the stabilizers of block B. We choose these

observables in such a way that the action of the circuit on the code space is not modified by their measurement, while their measurement still reveals the value of the actual stabilizers. We call these Clifford observables, the "modified stabilizers". One further trick here is to first perform a round of error correction on the target block C before measuring the non-constant stabilizers of block A and the modified "B-stabilizers". Let us first assume that we can perform a perfect error correction on the target register, mapping the state of the target block C back to the code space. After this perfect error correction, we have $X_i^C X_{i+1}^C = +1$ for all i. Note that

$$\prod_{j=0}^{k-1} \mathrm{CX}^{A,C}(i-j,i)\mathrm{CX}^{A,C}(i+1-j,i+1) =$$

$$\mathrm{CX}^{A,C}(i+1-k,i)\mathrm{CX}^{A,C}(i+1,i+1) \times \prod_{j=1}^{k-1} \mathrm{CX}^{A,C}(i+1-j,i)\mathrm{CX}^{A,C}(i+1-j,i+1)$$

and that

$$\mathrm{CX}^{A,C}(i+1-j,i)\mathrm{CX}^{A,C}(i+1-j,i+1) = \frac{1}{2}(I + Z_{i+1-j}^A) + \frac{1}{2}(I - Z_{i+1-j}^A) X_i^C X_{i+1}^C.$$

As $X_i^C X_{i+1}^C = +1$, we have

$$S_{i,k}^B = X_i^B X_{i+1}^B \mathrm{CX}^{A,C}(i+1-k,i)\mathrm{CX}^{A,C}(i+1,i+1),$$

which admits a constant weight now. It is important to note that these constant-weight "modified B-stabilizers" only commute with the A-stabilizers if the state of the block C is in the code space.

The remaining question is whether this procedure still works when the error correction step on the target block is imperfect, thus mapping imperfectly the state of the logical block C to the code space. In this case, the set of "modified B-stabilizers" may not commute with the A-stabilizers, thereby forbidding a simultaneous measurement of these two sets. Indeed, in the current error correction approach, the imperfection of the C-stabilizer measurements is compensated by the repetition of these measurements and a MWPM decoder. This procedure however requires to repeat the measurements a number of times that scales linearly with d and during which we cannot measure the A and B stabilizers. The final trick to get around this issue is to replace the current error correction procedure of the target block by a single round of Steane-style error correction. Indeed, while the Steane-style error correction step can still be faulty, the output errors are not correlated to the input errors. This means that the measurements of the subsequent A and B stabilizers might be faulty, but these errors remain independent and therefore one can still hope to achieve a phase-flip threshold. The full circuit for the logical Toffoli gate, including the different error correction steps, is depicted in Figure 3.9.

3.3.3 Pieceable fault-tolerant CZ gate

We argued that the logical CZ_L gate is not required for universality, as it can be straightforwardly synthesized using a logical Toffoli gate, together with a logical ancilla prepared in the $|-\rangle_L$ state. However, we will see in Chapter 5 an alternative construction of the logical Toffoli gate with better architectural properties, which requires a logical CZ_L gate. For this reason, and because the logical CZ_L construction is just an easier variation of the logical Toffoli circuit presented above, we quickly present how a logical CZ_L gate can be realized.

Similar to the Toffoli gate, the CZ_L gate is not transversal on the repetition code because of the logical $Z_L Z_L$ interaction. This, together with the fact that the logical Z_L operator is implemented by the transversal application of all the physical Z operators, implies that an all-to-all coupling is required between the physical cat qubits of the two logical blocks. Here, the pieceable fault-tolerant method of [130] can also be applied. More precisely, the logical gate is split as in the Toffoli case into d transversal pieces of d physical CZ gates, using the round-robin construction

$$CZ_L = \prod_{k=1}^{d} P_k = \prod_{k=1}^{d} [\prod_{i=0}^{d-1} \mathrm{CZ}(i-k+1, i)]. \tag{3.1}$$

There are two major simplifications with respect to the logical Toffoli gate. The first is that the phase-flips errors commute with the gate, such that the issue of the spreading of phase-flip errors from the target block to the control blocks does not exist here. Rather, the pieceable construction allows to prevent the accumulation of the non-propagating errors.

This second is that, for the specific ordering considered in (3.1), the (non-constant) stabilizers of both logical blocks remain Pauli operators of constant weight, which simplifies greatly their measurement. Indeed, noting again

$$U_k = \prod_{j \in [\![1,k]\!]} P_j$$

the unitary implemented by the first k pieces of the circuit, the non-constant stabilizers of the two code blocks $X_i^O X_{i+1}^O$, $O \in \{A, B\}$, $i \in [\![0, d-1]\!]$ become

$$\begin{aligned} S_{i,k}^A &= U_k X_i^A X_{i+1}^A U_k^\dagger = X_i^A X_{i+1}^A Z_i^B Z_{i+k}^B \\ S_{i,k}^B &= U_k X_i^B X_{i+1}^B U_k^\dagger = X_i^B X_{i+1}^B Z_{i-k}^A Z_{i+1}^B. \end{aligned}$$

The full circuit for the implementation of a pieceable fault-tolerant CZ gate on distance 3 repetition cat qubits is depicted in Figure 3.10.

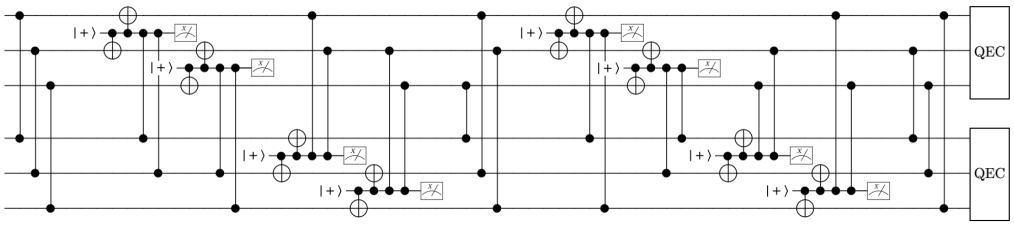


FIGURE 3.10: Non-transversal implementation of the pieceable fault-tolerant logical CZ gate on distance $d = 3$ repetition cat qubits.

Chapter 4

Error rates and overheads: a numerical study

This chapter covers the work of the pre-print [59].

4.1 Assumptions and methodology

4.1.1 The phase-flip threshold

The repetition cat qubit relies on two different kinds of protection. The two-photon dissipation exponentially suppresses bit-flip errors with the mean number of photons in the cat state, while the rate of phase-flip errors increases only linearly. Next, the repetition code suppresses exponentially the phase-flip errors, provided that the phase-flip error rate of the cat qubit is below the fault-tolerance threshold of the repetition code.

We argued in Chapter 1 that this protection is similar to the one achieved by Bacon-Shor codes [111, 14], with the nice feature that the "distance" of the inner protection provided by the two-photon pumping can be increased without any further

hardware overhead. However, similarly to Bacon-Shor codes, because the phase-flip error rate of the cat qubit increases linearly with the mean number of photons, there cannot be a threshold since the effective phase-flip error rate of the cat qubit will eventually exceed the threshold of the repetition code. Nonetheless, this is not an obstacle to obtaining extremely low logical error rates with this approach, by limiting the mean number of photons to a finite value for which the bit-flip error probability is extremely low, and for which the phase-flip error probability is still below the threshold of the repetition code. In our analysis, we limit the size of the cat qubits to $\bar{n} = 15$ photons and assume that the exponential suppression of the bit-flip error rate holds at least up to this cat size.

All the circuits presented in this work are built to implement logical gates on repetition cat qubits, while being fault-tolerant to phase-flip errors only. Indeed, any single bit-flip error occurring on any data qubit during the execution of a circuit can cause a logical bit-flip error. The resulting logical bit-flip error rate can therefore be bounded by simply counting the number of single locations in the circuits where a bit-flip can occur. The numerical simulations of the logical circuits is devoted to estimating the logical phase-flip error rate only, without taking into account the bit-flip errors. For fault-tolerant circuits with respect to phase-flip errors, we define the "phase-flip threshold" to be the highest value of the physical phase-flip error probability p_{th} for which the logical phase-flip error probability decreases upon an increase of the code distance d.

4.1.2 Error models and thresholds

The accuracy threshold of a given quantum code is an important figure. Its actual value is highly dependent on the assumptions made about the noise, and in the case of a direct simulation of the quantum code, on the decoder used to interpret the measurement outcomes, the various hypothesis made in the simulations, etc. The error models considered in the literature usually fall into one of three classes that we briefly describe below, each of which being suited for studying a particular feature of the code. The corresponding allocation of errors to each of these error models is illustrated in the case of the repetition code in Figure 4.1.

Code capacity error model The code capacity error model (Figure 4.1 (a)) corresponds to the case where errors can affect the data qubits, but the syndrome extraction circuit is perfect. This model is useful to study the intrinsic properties of a quantum code, for instance, how the surface code can be tailored when the base qubits are suffering from biased noise [120]. In this case, the threshold of the 2D surface code for a depolarizing error model (where all single qubit Pauli errors are equally likely) is $p_{th} = 10.3\%$ using a maximum-likelihood decoder [124] and an optimal decoder increases this value to slightly above $p_{th} = 10.9\%$ [104, 98]. The code capacity threshold for the repetition code is $p_{th} = 50\%$. In this model, the stabilizer measurement does not need to be repeated.

Phenomenological error model The phenomenological error model (Figure 4.1 (b)) includes the errors of the code capacity model, and furthermore adds some errors on the measurement outcome of the stabilizers (which fail with some probability that is modelled phenomenologically). The stabilizer measurement are repeated to ensure fault-tolerance and the history of all the measurement outcomes is interpreted using a decoder. In this model, and for depolarizing noise, the threshold of the surface code falls to around 2.9% [124] (and 3.3% using an optimal decoder [97]), and that of the repetition code falls to about 10% [115].

Circuit-level error model The circuit-level error model includes errors at every location in the circuit, including the locations where qubits are idle. The interest of this error model is to give precise estimations of what can actually be expected in a real world experiment, and the value of the accuracy threshold in this model gives (roughly) the target to reach to implement error correcting codes. The surface code threshold for the circuit-level error model (depolarizing) is around 1% (see e.g [127] among many others) while that of the repetition code is about 3% [115].

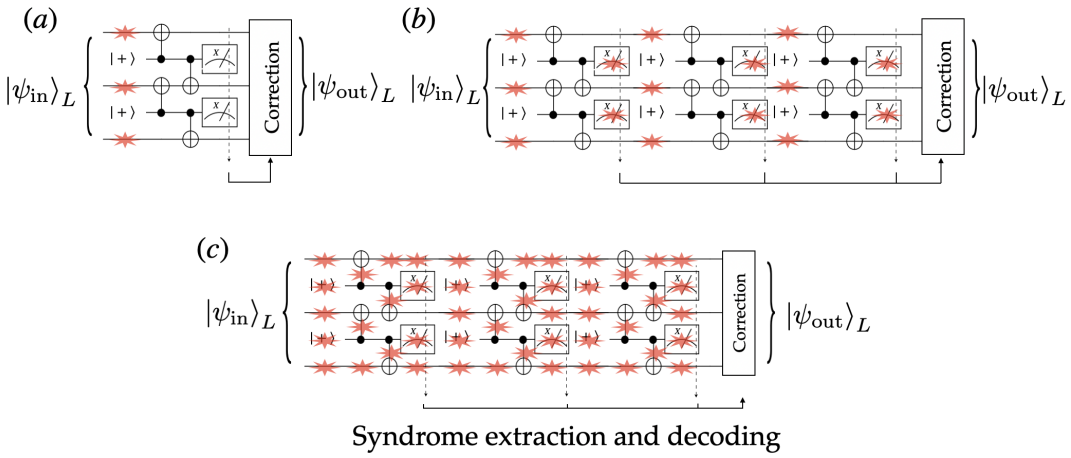


FIGURE 4.1: Location of errors in the stabilizer circuit measurement for (a) Code capacity noise (b) Phenomenological noise (c) Circuit-level noise.

Description of the simulation We now describe how the logical phase-flip error rate can be computed through numerical simulations of the circuits. The error model that we choose is the circuit-level error model, in order to get estimates of the logical error rates and overheads as close as possible to one could realistically expect in an experimental setting.

Every noisy gate is modelled as a perfect gate, followed by a stochastic error. These error models account for non-adiabatic errors and for the effect of single photon loss at rate κ_1. The parameter p that characterizes the "strength" of the physical noise is the error probability of a physical phase-flip during the typical gate time T, given by $p = \bar{n}\kappa_1 T$, where $\bar{n}$ is the average number of photons. For the gate time T^*

that maximizes the CNOT and Toffoli gate fidelities, this probability is given by

$$p = \frac{1}{2\sqrt{\pi}}\sqrt{\kappa_1/\kappa_2}.$$

For each gate, we summarize the Z-type errors and the corresponding probability that we have used in the Monte Carlo simulations in Table 4.1. We also assume both the ancilla preparation and measurement to be faulty with probability p.

I		Z		CZ		CNOT		Toffoli	
Error	Prob.	Error	Prob.	Error	Prob.	Error	Prob.	Error	Prob.
I	$1-p$	I	$1-p$	I	$1-2p$	I	$1-4p$	I	$1-6p$
Z	p	Z	p	Z_1	p	Z_1	$3p$	Z_1	p
				Z_2	p	Z_2	$p/2$	Z_2	p
						Z_1Z_2	$p/2$	Z_3	$p/2$
								CZ_{12}	$3p$
								$CZ_{12}Z_3$	$p/2$

TABLE 4.1: Error models of each gate used in the simulations.

For simplicity, we fix the duration of these time-steps to be the same as T, the duration of CNOT and Toffoli gates. When a qubit is acted upon by a gate at a given time-step, the applied probabilistic error is drawn from the corresponding error model, otherwise, the error is drawn from the identity error model (which corresponds to a phase-flip probability of $p = \bar{n}\kappa_1 T$).

In order to simulate a given circuit, we fix the code distance d and the value of the physical noise strength p and run the noisy circuit N times. We detail in subsection 4.1.3 how the noisy circuits are simulated efficiently. For each trajectory, the output of the syndrome measurements is decoded using a minimum weight perfect matching (MWPM) decoder (see subsection 4.1.4) and a final perfect recovery operation [44]. After the recovery operation, mapping the state back to the code space, we check whether a logical error has occurred. We then define the logical error probability of the circuit $p_L(d, p)$ as

$$p_L = \frac{N_{\text{fail}}}{N}$$

where N_{fail} is the number of times a logical error occurred during the N runs. In order to obtain a constant relative error on the value of p_L, the circuits are run continuously until a minimum of $N_{\text{fail}} = 500$ logical failures are observed, which ensures that the relative error on p_L is less than 9% with probability 95%.

The numerical computations were performed in parallel using the cluster of Inria Paris, composed of 68 nodes for a total of 1244 cores. The nodes are divided in a few hardware generations: 28 bi-processors Intel Xeon X5650 of 6 cores, 12 bi-processors E5-2650v4 2.20 of 12 cores, 16 bi-processors XeonE5-2670 of 10 cores, 8 bi-processors E5-2695 v4 of 18 cores, 4 bi-processors E5-2695 v3 of 14 cores. Some

data points for the logical Toffoli circuits corresponding to the largest distances and lowest logical error probabilities, for which $\sim 10^8 - 10^9$ trajectories were simulated per point, required up to a week (real time) of computation.

4.1.3 Efficient simulation of non-Clifford circuits using CHP algorithm

The numerical determination of the logical error probability is a schizophrenic game. For each run of the circuit, one generates random errors at random locations in the circuit, and keeps them stored somewhere. Then, once the location of errors (and the type of errors, as multi-qubit gates can produce different errors) are drawn, one needs to *simulate* the circuit in order to get the outcomes of the stabilizer measurement. Then, observing only the stabilizer measurement outcomes, one tries to guess the error that occurred to decide which correction needs to be applied.

In this subsection, we detail how the measurement outcomes can be obtained once the location of errors has been drawn. There are actually two different cases to consider: the stabilizer circuits and the non-stabilizer circuits. A circuit is called a stabilizer circuit, or a Clifford circuit, if it is composed exclusively of quantum gates belonging to the Clifford group, and state preparations and measurements in the computational basis only. The Gottesman-Knill theorem establishes that such circuits can be simulated efficiently on a classical computer [53]. Most of the logical circuits that we considered in Chapter 3 actually fall in this class: the preparation of the logical states $|\pm\rangle_L$, the measurement of the logical Pauli X_L operator, the error correction circuit, and the logical transversal CNOT gate. The non-stabilizer circuits that we introduced were the two circuits for the logical Toffoli gate, as they involved using physical Toffoli gates.

The original idea behind the Gottesman-Knill theorem is to use an operator representation of the quantum state rather than the wavefunction, *i.e* to track the evolution of a quantum state through a stabilizer circuit in the Heisenberg picture rather than in the Schrödinger picture. There are two points to address: how does one represent a stabilizer state in the Heisenberg picture? How can one track efficiently the evolution of this representation through a stabilizer circuit?

The first answer is actually closely related to the stabilizer formalism of quantum error correction. The goal is to represent the *stabilizer states*, *i.e* the set of all possible states that can be produced by a stabilizer circuit (the states that one can reach by applying Clifford gates to the all 0 state). The key point is that an n-qubit stabilizer state is the unique (up to a global phase) $+1$ eigenstate of exactly 2^n Pauli operators, that form a group generated by n Pauli operators. For instance, the 1-qubit state $|0\rangle$ is the $+1$ eigenstate of the set of two Pauli operators $\{I, Z\}$ generated by a single Pauli operator $\{I, Z\} = \langle Z\rangle$, the 2-qubits state $\frac{1}{\sqrt{2}}(|0+\rangle - |1-\rangle)$ is the $+1$ eigenstates of the Pauli operators in the set $\{II, -XZ, ZX, -YY\} = \langle -XZ, ZX\rangle$, the

3-qubit state $|+1-\rangle$ is the common $+1$ eigenstate of the Pauli subgroup generated by $\langle XII, -IZI, -IIX\rangle$, etc.

Now, why is this representation useful in designing an efficient simulation algorithm ? The strategy behind the efficient simulation algorithm of stabilizer circuits is to keep track of the n Pauli generators of the group representing the state, rather than keeping track of the 2^n complex amplitudes of n-qubit state. The crucial point is that the Pauli operators representing the state *remain* Pauli operators upon the application of gates in the Clifford group (by definition of the Clifford group). Thus, one merely needs to update the Pauli operators of the generating set to keep track of the evolution of the state, which can be achieved efficiently [1].

Toffoli circuits simulation We now describe how this efficient algorithm can be used for our purpose. Let us begin by the less trivial case, the simulation of the two logical Toffoli circuits proposed in the Chapter 3. The simulation of non-Clifford circuits is classically hard (not surprinsingly, as the interest of building a quantum computer would otherwise be very limited). In our case, however, there are a few specific features that enable us to perform the Monte Carlo simulations of all the non-Clifford circuits presented in this work in a classically efficient manner. The first important thing to note is that while the circuits contain non-Clifford Toffoli gates, the propagation of errors in the circuits can never produce non-Clifford errors, as would be the case in all generality. To see this, recall from the error models of Table 4.1 that all the 'bare' errors produced by any gates in the Toffoli circuits are of the following form: a Pauli Z error on any data qubit of any logical block, a Pauli correlated Z_1Z_2 error on any two pair of data qubits of the two control blocks, a controlled-phase gate CZ_{12} between any two data qubit of the control blocks, or a correlated controlled-phase gate and Z error $\mathrm{CZ}_{12}Z_3$ on two data qubits of the control block and one data qubit of the target block.

Crucially, none of these errors can become non-Clifford through the Toffoli circuits: the Pauli Z errors of the control blocks commute with the Toffoli gates, while a Z error on the target block evolves through a Toffoli gate as a Z error on the target block together with a controlled-phase error between the two controls:

$$\mathrm{CCX}_{1,2,3} \times \mathrm{Z}_3 = \mathrm{CZ}_{1,2}\mathrm{Z}_3 \times \mathrm{CCX}_{1,2,3}.$$

Thus, the only Clifford error that can ever appear anywhere in the circuits is a controlled-phase between any pair of two qubits of the control blocks, either produced by a Toffoli gate error or by propagation of a Z error on the target block through another Toffoli gate. Once these errors appear, however, they can never propagate further to non-Clifford errors as a controlled-phase gate on the control qubits of a Toffoli gate commutes with the Toffoli gate:

$$\mathrm{CCX}_{1,2,3} \times \mathrm{CZ}_{1,2} = \mathrm{CZ}_{1,2} \times \mathrm{CCX}_{1,2,3}.$$

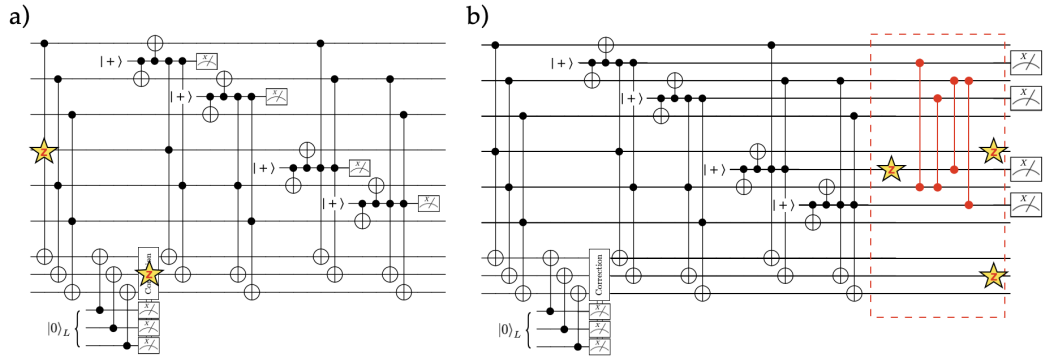


FIGURE 4.2: Noisy piece of a Toffoli circuit (a) where the errors are drawn randomly from the errors models of Table 4.1. Here, the first physical Toffoli produces a Z error on the second qubit of the second control block and the Steane error correction produced a Z error on the second qubit of the target block, depicted by the stars. This noisy circuit is equivalent to the one depicted in (b), where the Toffoli circuit is now perfect (error less) and is followed by a circuit containing the noise (in the red dashed box), but composed exclusively of Clifford gates.

Thus, several specific ingredients ensure that all the errors we deal with are at most Clifford errors:

- the noise bias, such that we deal only with phase-flip types of errors,
- the bias-preserving property of all the gates, thus preserving the phase-flip nature of errors,
- the specific orientation of the Toffoli gates where the target qubit of any physical Toffoli gate always belong to the same code block.

The third point is actually a key point. If we were to use two Toffoli gates in a circuit with the target qubits belonging to two different logical blocks, then a CZ error produced by the first Toffoli gate could evolve to a non-Clifford CCZ error by propagation through the second Toffoli:

$$\mathrm{CCX}_{1,2,3} \times \mathrm{CZ}_{1',3} = \mathrm{CCZ}_{1,1',2} \times \mathrm{CZ}_{1',3} \times \mathrm{CCX}_{1,2,3}.$$

With these facts in mind, we now detail how the logical error probability of a noisy Toffoli circuit can be simulated efficiently. Once the locations and nature of errors have been drawn, the errors are propagated through the circuit up until the point where they meet a measurement. At this point, the circuit has been decomposed in two different circuits: the first one is perfect, and contains the non-Clifford Toffoli gates, and it is followed by a second one that consists of the errors only, and contains exclusively Clifford operations. We depict in Figure 4.2 one example of this circuit decomposition, for the first piece of the circuit of Figure 3.9. The left hand-side of the circuit, which is error free, is non-Clifford but its effect on the value of the stabilizers is trivial. Actually, since the operators that we measure are "compatible" with the Toffoli pieces of the circuit, in the absence of errors, the results of the measurement

of these operators is $+1$ (deterministic). The only thing that needs to be simulated numerically to get the correct probability distribution for the measurement outcomes is the effect of the error circuit, depicted in the red box of Figure 4.2 (b), on the measurement results and the remaining errors after the measurements have been executed. Note that since this error circuit is Clifford, it can be efficiently simulated using the CHP algorithm [1]. Note that the errors on the target block are always simple Pauli Z errors, even after propagation through any gate of the circuit. Thus, the logical error rate of the target block can actually be simulated separately from the rest, using a simple array of 0, 1 integers.

Stabilizer circuits simulation Given that the CHP algorithm handles efficiently stabilizer circuits, all the remaining circuits in this work are within the scope of this algorithm. However, such simulations can be further simplified for the same reasons mentioned above. Indeed, we saw in the case of the Toffoli circuits that what is needed to be simulated to produce the correct probability distribution for the measurement outcomes were actually the *errors* themselves, propagated through the circuits, while the effect of the (error free) circuit themselves was trivial. In the case of the Toffoli circuits, the errors could take the form of Clifford unitaries (the CZ error), while for all of the stabilizer circuits the errors actually remain Pauli. Thus, since the basis states that we consider are only the $|\pm\rangle$ states and the only errors that can happen are merely causing flips between these states, the effect of the randomly drawn errors in the circuits can be tracked classically even in the Schrödinger picture, which turned out to be the most efficient way to simulate the logical phase-flip error rates for all of the stabilizer circuits in this work.

4.1.4 Efficient decoding of the repetition code using the MWPM decoder

The problem of deciding what is the most likely set of errors that occurred given a pattern of stabilizer measurement outcomes (and hence the correction that needs to be applied) is known as the *decoding problem*, and the specific algorithm that solves this task is the *decoder*. Typically, in most current proposed schemes it is assumed that the decoding algorithm will be executed on classical hardware running alongside the quantum computer. An important question is whether this computation needs to be performed in real time, at the pace of the quantum computer, or if the measurement outcomes merely need to be recorded to be interpreted later. There are actually two cases to consider: when the quantum algorithm can be implemented using exclusively stabilizer circuits, Pauli errors remain Pauli as they propagate in the circuit. In this case, it is enough to simply detect the errors without correcting them, and to let them propagate until the end of the circuit where they can be corrected. Actually, in this case, the errors do not even need to be corrected but rather their effects can be taken into account when interpreting the final output of the quantum algorithm (another equivalent way to state this is to say that the errors can be corrected in software, by updating the Pauli frame [72]). However,

a quantum algorithm providing a quantum speed-up will necessarily involve non-Clifford gates in its implementation (otherwise it is possible to efficiently simulate it on a classical hardware according to the Gottesman-Knill theorem). The propagation of Pauli errors through these non-Clifford gates results in errors of increasing complexity, as a Pauli error that propagates through k non-Clifford gates might in general become an error belonging to the $(k+1)$-th level of the Clifford hierarchy. Tracking the effect of these errors as they propagate becomes classically intractable, and in this case, errors need to be *corrected* before the execution of non-Clifford gates. It is then crucial that the entire error correction procedure, including the stabilizer measurements, the classical decoding of the measurement outcomes, and the correction step, can be performed in a time similar to the typical clock cycle time of the quantum computer. The design of an efficient decoder for the stabilizer codes, and most particularly for the surface code, is an important topic of research that has been extensively studied in parallel to the development of the quantum hardware itself. One of the leading decoders for the surface code is known as the "minimum weight perfect matching" (MWPM) decoder [46, 44, 47, 45, 42, 126], and it relies on a mapping of the decoding problem onto a well known problem of graph theory. The graph problem consists in finding a pairing of nodes of a weighted undirected graph (a matching) that minimizes the sum of the weights of the selected edges (the minimum weight), and such that all the nodes are matched (a perfect matching). The "Blossom" algorithm [38, 16] solves this problem efficiently, providing an efficient decoder of stabilizer codes.

In the numerical study presented in this work, the efficiency of the decoding algorithm is convenient to perform efficient simulations of circuits involving a few tens of cat qubits. Indeed, an alternative way to decode the measurement outcomes is the direct simulation of the full density matrix of the system, which contains all of the information about the system and can therefore be used to compute numerically the most likely state of the system given an observed pattern of measurement outcomes. However, this requires the computation of a complex-valued matrix of size $2^n \times 2^n$, which quickly becomes intractable for classical computers. Instead, the MWPM decoder focuses on the pattern of errors, which is typically very sparse when the errors are rare, independent of the code distance. We now describe how the observed pattern of measurement outcomes can be mapped (efficiently) on the graph problem mentioned above, for the specific case of the repetition code that we use in this work, and for the simple QEC circuit (that is for the case of the memory). Note that this decoder has been used to decode the first experimental implementation of a quantum repetition code with transmon qubits [67].

Repetition code with perfect stabilizer measurement Let us first consider the ideal case where the stabilizer operators $X_i X_{i+1}$ are measured with unit fidelity. In this case, all the operations of the stabilizer circuits are assumed to be perfect (the

ancilla preparation and measurement, the CNOT gates used to copy errors from data qubits to ancilla qubits, and idle operations on data qubits are all assumed to be error-free), such that we only consider the errors on data qubits at the input of the circuit (say, from the circuit implementing a logical gate just before, or from a logical state preparation, etc). In this case, since the outcomes of the stabilizer measurements are perfectly reliable, and the measurement circuits themselves do not introduce additional errors, the circuit does not need to be repeated.

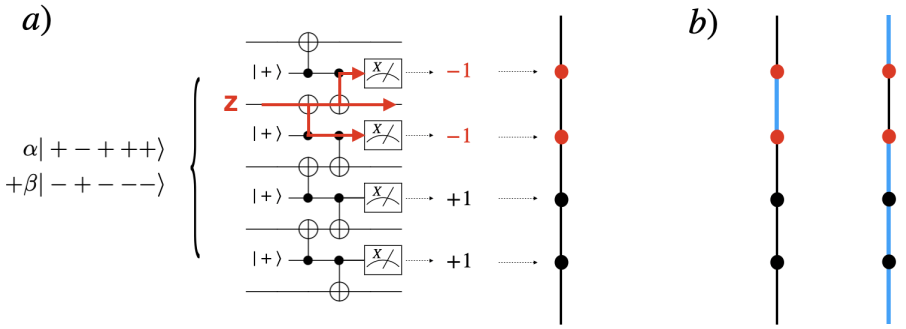


FIGURE 4.3: a) A single round of perfect stabilizer measurement is performed on a distance-5 repetition cat qubit. To decode the outcomes, the detection events are put on a graph where the edges represent the physical qubits and the nodes the detection events. b) The Blossom algorithm matches every detection event in pair while minimizing the sum of the weight of the edges. Here, there are two possible solutions to the observed syndrome (-1,-1,+1,+1) corresponding to either the second qubit having phase-flipped (red nodes matched together) or the all of the other qubits having phase-flipped (each node matched to a physical boundary).

Consider a distance-5 repetition code encoding an arbitrary logical state

$$|\psi\rangle_L = \alpha|+\rangle_L + \beta|-\rangle_L = \alpha|+\rangle^{\otimes 5} + \beta|-\rangle^{\otimes 5}.$$

Suppose that the second physical cat qubit of this code suffers from a Z_1 error just before the round of stabilizer measurements, as depicted in Figure 4.3 a). This error is detected as expected by the two stabilizers X_0X_1 and X_1X_2 that are supported on this qubit, which can be visualized directly on the circuit by noting that the Z_1 errors propagates through the CNOT gates of the corresponding stabilizer measurements and flip the $|+\rangle$ state of the corresponding ancilla to $Z|+\rangle = |-\rangle$, resulting in a deterministic -1 outcome for the X measurement of the ancilla. The collection of measurement outcomes of all the stabilizers, called the *syndrome*, is given by $(-1,-1,+1,+1)$. In order to *decode* this syndrome using a MWPM decoder, the measurement outcomes are represented on a graph as depicted in Figure 4.3 where the vertical vertices represent the qubits and the nodes in between two qubit represent the measurement outcome. Actually, a node represents a *detection event*, that is a node is selected (depicted in red) if the corresponding measurement outcome differs from the one of the previous round (here, the two -1 outcomes

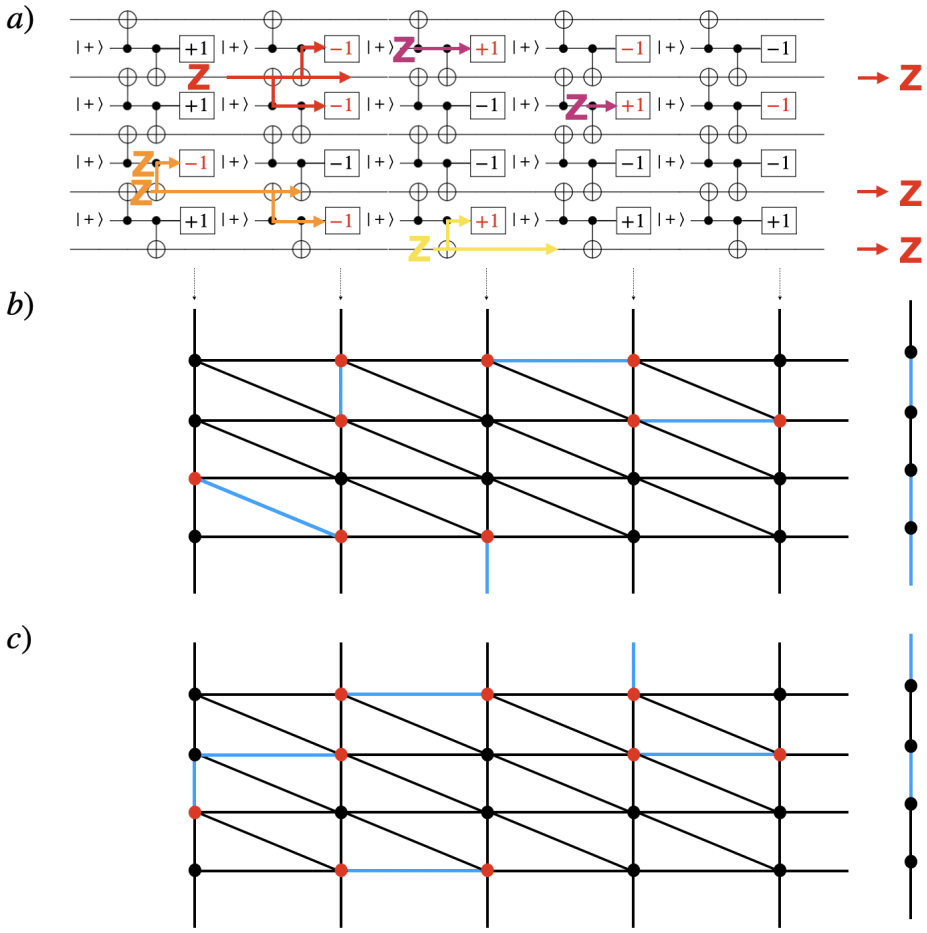


FIGURE 4.4: MWPM decoding of the distance-5 repetition code for circuit-level error model. a) The stabilizer measurement circuit is repeated $d = 5$ times before the syndrome are decoded together. While some errors occur on the data qubits (red and yellow errors) and need to be corrected, other errors (purple) affect the ancilla qubit and cause a wrong measurement outcome to be recorded, but leave no error on the data qubit, and some correlated errors occur on both (orange). b) The detection events (change of a measurement outcome with respect to its previous value) are put on a graph and matched in pairs. A selected edge by the matching (in blue) that has a vertical component indicates a physical error on the corresponding data qubit which needs to be corrected. c) A different matching, leading to a different interpretation of the measurement outcomes pattern. This matching is of higher weight and is rightfully not selected by the decoder, as it would lead to a logical error after correction.

result in two detection events at the corresponding locations assuming the previous syndrome was the all $+1$ syndrome corresponding to no error). The edges of the graph are weighted, where the weights of all the vertices are pre-computed using the error models of the operations. This step has to be performed only once and can be done before the quantum computer is run, but it requires to have a solid understanding of the noise models of all the operations of the quantum computer.

Given the set of nodes corresponding to detection events (red nodes), and the weights of the graph, the blossom algorithm determines a matching of these nodes of minimal weight, where the weight of the matching is the sum of the weights of the selected edges. In Figure 4.3 b), there are actually only two possible matchings of these two nodes: either these two nodes are matched together (with the corresponding selected edge marked in blue) or each of these nodes is matched with the boundaries of the graph. Suppose that each qubit at the input may have suffered from a phase-flip error with constant probability p and therefore the weight of each edges is $-\ln(p)$. The total weight of the first matching is $-\ln(p)$, while that of the second is $-4\ln(p) = -\ln(p^4)$. As $0 < p < 1$, the minimum weight matching is the first one where only one edge is selected. Indeed, in the case of a perfect measurement of the syndrome, there are always two different solutions for the matching problem that are complementary to one another: the two states

$$\begin{aligned} Z_1|\psi_L\rangle &= \alpha|+-+++\rangle + \beta|-+---\rangle \\ Z_L Z_1|\psi_L\rangle = Z_0Z_2Z_3Z_4|\psi_L\rangle &= \alpha|-+---\rangle + \beta|+-+++\rangle \end{aligned}$$

produce the same syndrome $(-1,-1,+1,+1)$. However, if one assumes that each qubit may have flipped with the same probability p, the probability to get the first state is p whereas the second one requires four independent errors of probability p, which is what is captured by the matching algorithm.

Repetition code with imperfect stabilizer measurement The assumption that the stabilizer operators can be measured with perfect accuracy, and without introducing errors on the system while the measurements are executed, is not realistic. Indeed, all the parts of the measurement circuit are themselves prone to errors. Some of these errors only cause the measurement outcome to be wrong, without having an effect on the data qubits. This is the case for the errors in the ancilla state preparation or measurement, or for the phase-flip errors on the ancilla (control) qubit introduced by the CNOT gate. Because of these "readout" errors, the outcome that one gets when measuring a stabilizer may differ from the real value of the stabilizer, and one cannot trust a single measurement outcome to decide which correction should be applied. Indeed, if one proceeds as such, a single erroneous syndrome results in applying a wrong correction, which can lead to a high weight errors on the data caused by the correction itself.

In order to avoid this, the measurement of the stabilizers can be repeated a certain number of times before a correction is applied, where the applied correction depends on the entire history of the past measurement outcomes. The difficulty in achieving this is that other errors, like the phase-flip errors on the data (target) qubit introduced by the CNOT gate, or the phase-flip errors on the data while idling during the ancilla preparation or measurement, actually cause the state of the

system to evolve in time. We depict schematically in Figure 4.4 how the MWPM decoder works in this case. Consider the circuit that repeatedly measures the stabilizers of the distance-5 repetition code, where phase-flip errors may occur at every locations in the circuit. In 4.4 a), a particular instance where five particular errors happen at random locations is shown, and the corresponding ± 1 outcomes that one gets in this particular case are given in the measurement boxes. Unlike the case of the perfect stabilizer measurement, where the solution to the decoding problem is obvious, here it is much more difficult to guess what the state of the system might be by just looking at the syndromes, that keep changing at every round. The first step to decode this syndrome is to actually identify when a stabilizer measurement outcome *differs* from the outcome of the previous round. The change of measurement outcomes, called a detection event, indicates that an error has occurred. In general, for the repetition code, a single error always triggers two neighbouring detection events (except at the physical or temporal boundaries, where a physical boundary is an edge of the repetition code and a temporal boundary denotes the last round of measurement performed in time). Depending on the location and nature of the error, the two detection events can be aligned *in time*, in the case of a measurement error (purple), or aligned *in space*, in the case of an error affecting a data qubit between two different measurement rounds (red), or in both direction (orange) for some very specific locations (a phase-flip on the target qubit of a CNOT of the first round of CNOT gates, or a correlated ZZ errors on both qubits of a CNOT of the second round). Errors that occur at the boundaries (yellow) only cause a single detection event. To decode the history of syndromes corresponding to a particular Monte Carlo instance, the detection events are put on the *detection graph* at the corresponding locations. The detection graph is precomputed (only once, outside the Monte Carlo loop) using the errors models of the gates. For a given circuit, and errors models for all the operations in the circuit, we build this graph in two steps. First, a node is added for every possible location of a detection event, *i.e* for every measurement in the circuit. Second, the edges and the corresponding weights are added. In order do to this, for every possible error, the two detection events (or the single detection event) that this error will trigger are identified and the weight of the corresponding edge is updated to take into account the corresponding probability.

Once this graph is computed, the Monte Carlo simulation proceeds as follows. For each instance (*i.e* for each particular draw of the random set of errors), the complete subgraph to feed to the matching algorithm is computed by calling Dijkstra's algorithm (shortest path finding algorithm [34]) for each pair of detection events. Then, this complete subgraph is fed to the Blossom V implementation of MWPM algorithm [16]. Once the detection events have been matched (possibly with a physical/temporal boundary), all that remains to be done is to determine the correction to apply, given the particular matching of detection events. We depict in Figure 4.4 b) how this is done. The detection events (red nodes) corresponding to the locations of

the circuits where the measurement outcomes differ from that of the previous round, are matched in pairs (blue edges) such that the sum of the weights of the matching is minimal. The horizontal edges of the matching actually identify measurement errors, for which no error actually happened on the data and hence no correction needs to be applied. The vertical edges of the matching (or the diagonal edges), however, identify pairs of detection events corresponding to errors that occurred on the data qubits and that need to be corrected. Tracking all of the pairs of matched nodes that account for an error on a data qubit gives the correction to apply to remove these errors. Note that in the depicted example, the decoder correctly finds that three of the five qubits have actually suffered from phase-flips (as depicted on the right). Another possible matching of the detection event is the one depicted in Figure 4.4 c), leading to guessing that actually only two qubits have been phase-flipped (and which would lead to a logical error after correction), but this matching has a higher weight (a detailed look at the error models and the weights of the edges is required to check this, however one can readily note that the matching of c) requires six edges instead of five for the minimum weight matching of b)).

4.2 Performance of the quantum memory and transversal operations

4.2.1 The memory

In this section, we investigate the performance of a repetition cat qubit used as a quantum memory. Here, the error correction is applied to extend the lifetime of a quantum bit of information. In this case, the logical circuit (implementing the logical encoded version of identity operation) simply consists of the error correction step. The $d-1$ stabilizer operators are measured using $d-1$ ancilla qubits. To make the procedure fault-tolerant, the measurements are repeated d times before they are decoded with the MWPM decoder. Since all the operations are bias-preserving and do not convert X and Z errors, we estimate separately the error probabilities p_{Z_L} and p_{X_L} of logical Z_L and X_L errors occurring per cycle of error correction. We then bound the global logical error probability by $p_L = p_{Z_L} + p_{X_L}$.

Logical X error probability p_{X_L} The repetition code does not provide any protection against bit-flip errors. Hence, a single bit-flip occurring on any qubit during the execution of the circuit will cause a logical X_L error. For all the bias-preserving operations presented in Chapter 2, the bit-flip rate is exponentially suppressed with the mean number of photons in the cat state (experimentally observed in [83] for the identity operation). However, the exact value of this probability depends on the operation that is applied to the cat qubit.

The value of the bit-flip error probability induced by each gate is determined numerically. The most important source of bit-flip errors is found to be induced by the CNOT gate, such that the bit-flip errors induced by idling data qubits are actually negligible with respect to the bit-flips occurring when a CNOT gate is performed. Here, we assume pessimistically that any bit-flip type of error for the CNOT gate will result in a logical bit-flip error (which is not always the case, as for instance a correlated XX error on both the control and the target of a CNOT gate of the first round propagates through the second CNOT as a $X_i X_{i+1}$ error on the data, which is a stabilizer and hence does not induce an error). In this case, the bit-flip error rate for the CNOT is very well approximated by the numerical fit [26]

$$p_X^{\mathrm{CX}} = (5.58\sqrt{\frac{\kappa_1}{\kappa_2}} + 1.68\frac{\kappa_1}{\kappa_2})e^{-2\bar{n}}$$

where p_X^{CX} is the sum of all the probabilities of the 12 errors that contain some bit-flip (X_1, X_1X_2, X_1Y2, X_1Z_2, X_2, etc). A full cycle of QEC requires $2d(d-1)$ CNOT gates. Assuming pessimistically that any single bit-flip error will result in a logical X_L error, and assuming p_X^{CX} is small, the resulting logical error probability is simply bounded by

$$p_{X_L} = 2d(d-1)p_X^{\mathrm{CX}}.$$

Logical Z error probability p_{Z_L} To estimate p_{Z_L}, we perform Monte Carlo simulations of the QEC circuit depicted in Figure 4.1-(c) where we assume there are no physical bit-flip errors, as they are accounted for separately. Here and in the following simulations, we assume that the classical processing of the measurement outcomes is instantaneous, so no errors are induced on the data qubits while the decoding is performed. In the memory case, we also assume the correction step is perfect, because note that as discussed previously, in this case the correction does not need to be physically applied but rather can be performed in software by updating the Pauli frame [72].

For each run, the repetition cat qubit is initialized in a code word $|\psi_{in}\rangle_L$. The stabilizers of the code are measured d times as depicted in Figure 4.1-(c). A last round of perfect stabilizer measurements is performed and the history of measurement outcomes is decoded together with this last perfect measurement outcome. This ensures that, after the perfect correction, the output state $|\psi_{out}\rangle_L$ is back in the code space, either $|\psi_{out}\rangle_L = |\psi_{in}\rangle_L$, in which case the error correction was successful, or a Z_L error occurred, $|\psi_{out}\rangle_L = Z_L|\psi_{in}\rangle_L$. We plot in Figure 4.5 the probability Z_L that a logical error occurred for various code distances d and values of the physical noise strength p. The phase-flip threshold for this circuit is $p_{\mathrm{th}} = 1.9\%$, which corresponds to a ratio between the two-photon dissipation rate and a single photon loss rate $\kappa_2/\kappa_1 = 220$, close to the value achieved in [118]. For a typical cavity lifetime of 1ms and cat qubits of size $\bar{n} = 5 - 15$ photons, this phase-flip threshold of 1.9% corresponds to a CNOT

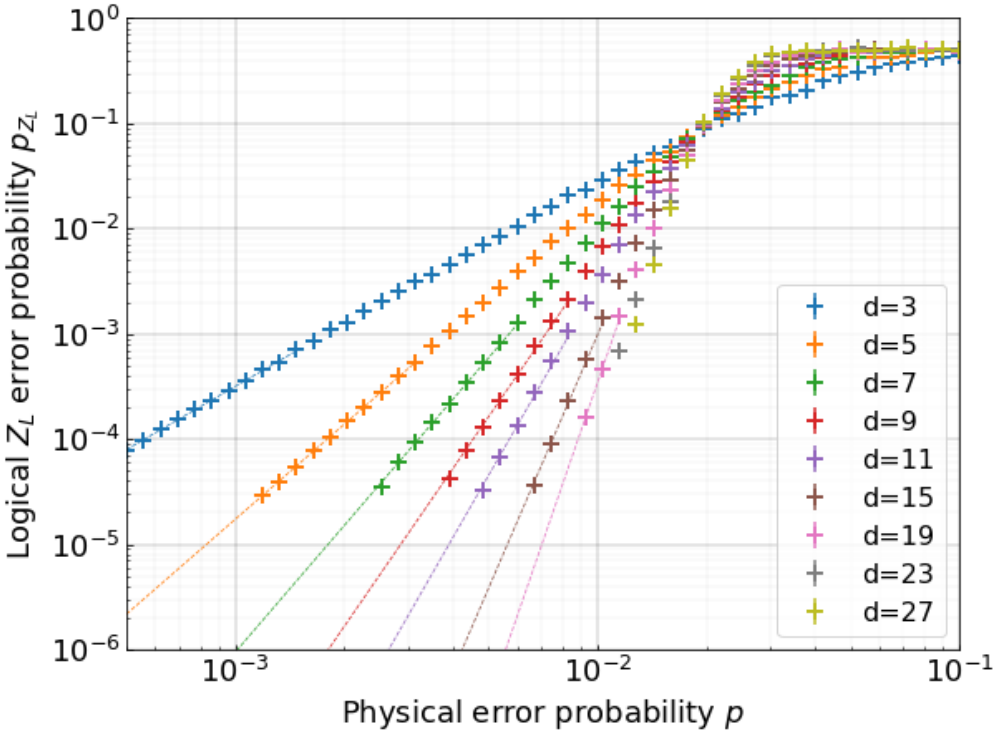


FIGURE 4.5: Probability that the error correction circuit of Figure 4.1-(c) induces a logical Z_L error on the repetition cat qubit after the correction is performed. The dotted lines correspond to the asymptotic regime and fit the empirical scaling formula $p_{Z_L} = A(\frac{p}{p_{th}})^{\frac{d+1}{2}}$.

gate time of about $4\mu s - 1.3\mu s$, respectively. It is experimentally reasonable to think that all the other operations could be performed as fast.

Note that the phase-flip threshold for the CNOT error probability is about 92%. For a depolarizing model where idle qubits, state preparation and measurement, and the CNOT gate all fail with probability p, and where the CNOT error model is balanced $p_{Z_1} = p_{Z_2} = p_{Z_1Z_2} = p/3$, the fault-tolerance threshold is slightly above 3% [115], which corresponds to a CNOT error probability around 97%. Here, a higher CNOT gate error probability is tolerated because the phase-flips errors of the CNOT mostly occur on the ancilla cat qubits used for the stabilizer measurement.

Logical error rate and resource overhead Combining the logical X_L and Z_L errors, the minimum number of data cat qubits and the minimum number of photons per cat qubit to achieve a target logical error rate p_L for a quantum memory can be estimated. In Figure 4.6, we present this physical overhead as a function of the physical error probability p. Physical error probabilities of about 1% (corresponding to a CNOT fidelity of 96%) are enough to achieve very low logical error probabilities of order 10^{-10} per QEC cycle using a modest number of 70 modes per repetition cat qubit (twice as much including the ancillary modes) and for cat sizes of about 15-16

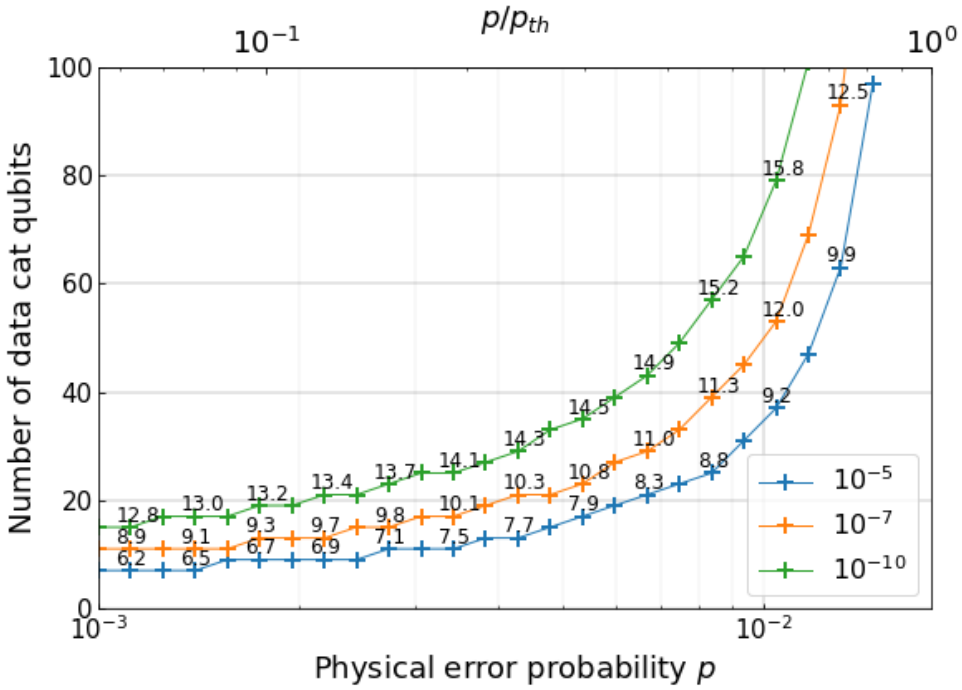


FIGURE 4.6: Estimated number of cat qubits per repetition cat qubit used as a quantum memory, versus the physical noise probability p, also given in units of the phase-flip threshold value p_{th}. The different plots correspond to different values of the target logical error probability per QEC cycle, and the numbers on the curves correspond to the mean number of photons $\bar{n} = |\alpha|^2$ in the cat qubits.

photons. Furthermore, with the specific gate realizations of Chapter 2, this physical error probability of 1% can for instance be achieved with a two-photon dissipation rate of 125kHz, a cavity mode lifetime of about 1ms and a gate time of about 0.6μs.

4.2.2 Transversal operations

All logical operations that admit a transversal implementation exhibit a similar performance to the quantum memory. This includes the measurement of X_L, the preparation of the logical $|\pm\rangle_L$ states, and the logical CNOT gate. The measurement of the X_L operator is done by measuring all the cat qubits in the X basis, followed by a majority vote on the measurement outcomes. The fault-tolerant preparation of the state $|\pm\rangle_L$ consists in preparing all the cat qubits in the $|+\rangle$ state, and performing a full round of error correction. The phase-flip threshold for this preparation is therefore the same as the quantum memory. The logical CNOT gate is implemented on the code space by performing a physical CNOT gate between each pair of cat qubits of two different logical code blocks, followed by a separate round of error correction on each logical block. As it can be seen from Figure 4.7, the logical phase-flip error probability of a logical CNOT gate is similar to that of a quantum memory.

The set of fault-tolerant gates that can be used to perform universal quantum computation using repetition cat qubits is $\mathcal{S}_L = \{\mathcal{P}_{|\pm\rangle_L}, \mathcal{M}_{X_L}, X_L, \text{CNOT}_L, \text{Toffoli}_L\}$.

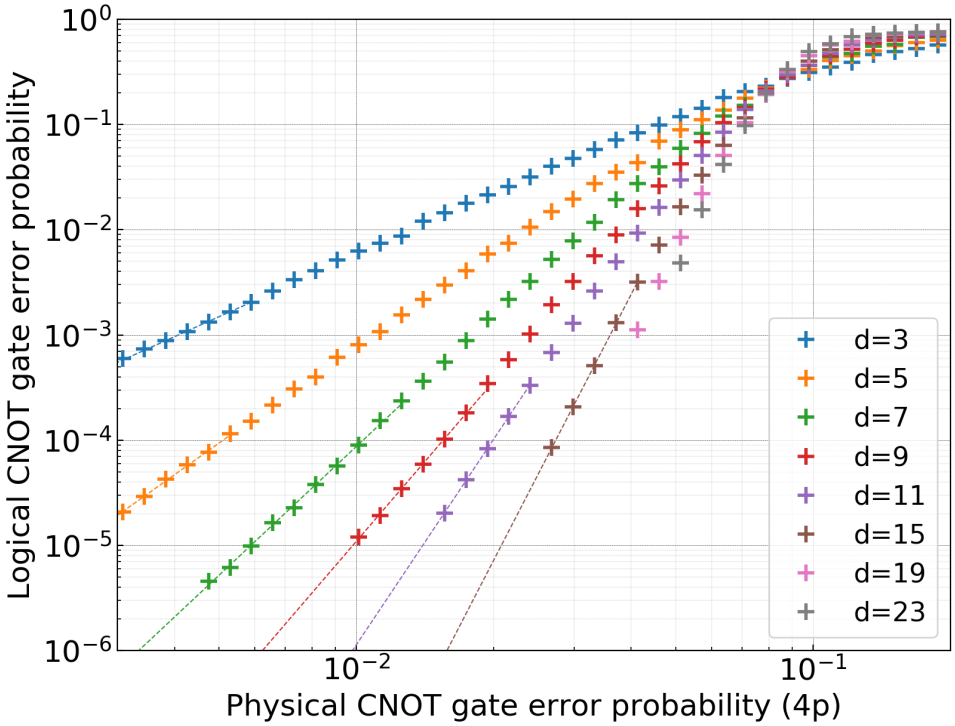


FIGURE 4.7: Error probability of a transversal logical CNOT gate as a function of the physical CNOT gate error probability given by $4p$. The asymptotic dotted curves are fits to the empirical scaling formula $A(\frac{p}{p_{th}})^{\frac{d+1}{2}}$.

In this section we showed that all these gates except the logical Toffoli can be implemented with very high fidelities for modest code sizes. In the next section, we investigate the performance of two schemes proposed in Chapter 3 to implement this non-Clifford gate fault-tolerantly using pieceable fault-tolerant circuits.

4.3 Performance of the Toffoli gate

4.3.1 Scheme 1 (with concatenation)

In Figure 4.8, we simulate the circuit of Figure 3.7 with a circuit-based error model. As mentioned above, the absence of a phase-flip threshold can be explained by the accumulation of non-propagating errors: for all values of the physical error probability p, there is a finite optimal value of the code distance that achieves a minimum logical error probability. We also plot in Figure 4.8 the identity line to visualize the "break-even" point below which code concatenation becomes possible. The phase-flip threshold for this circuit used with concatenation is slightly below 2%.

In the case where p is small and for large enough code distance d, the infidelity of the circuit is dominated by the accumulation of non-propagating errors on the control blocks. The probability that the circuit fails due to the errors in the control

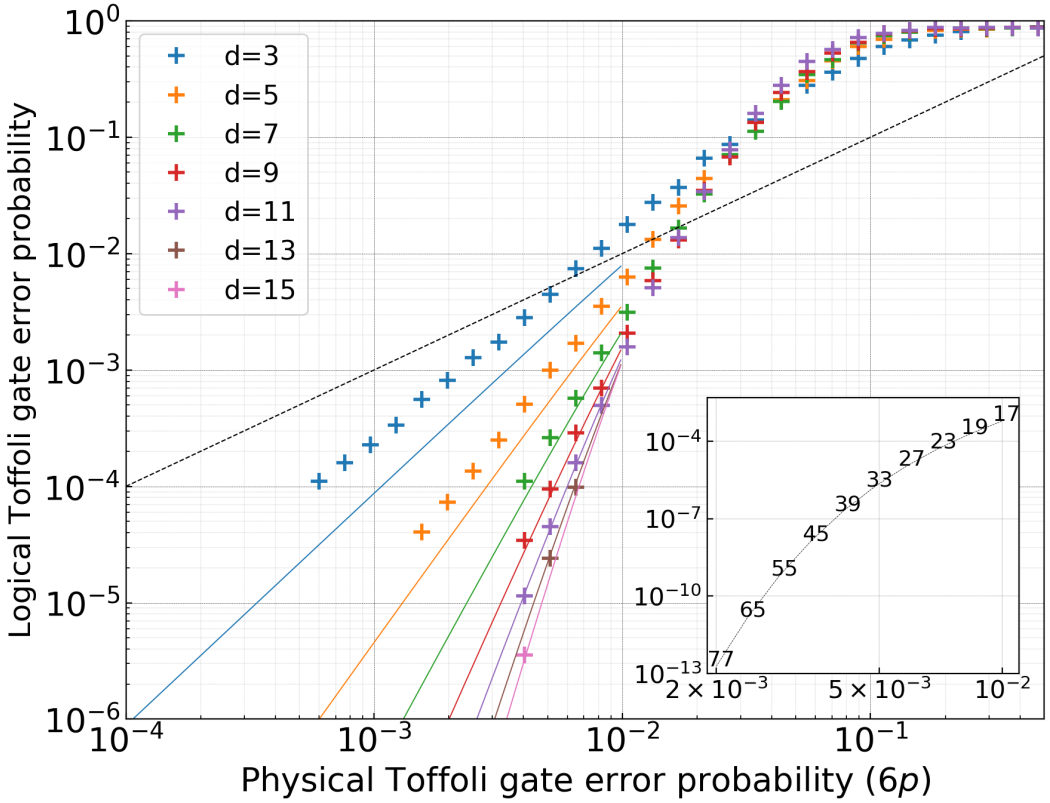


FIGURE 4.8: Error probability of the logical Toffoli gate implemented by the circuit of Figure 3.7 as a function of the error probability of physical cat-qubit Toffoli gates. The colored cross are numerical simulation results, the plain lines are computed analytically and correspond to the error probability of this circuit in the case where the final QEC stage is perfect (see main text). The dotted black line is the identity and serves as a guide to the eye to visualize the "break-even point" below which the error probability of the logical Toffoli circuit is smaller than that of the physical Toffoli gate. Inset: Minimal value of the logical Toffoli error probability achievable for a given physical error probability. The optimal distance realizing this minimum is indicated on the curve.

blocks exceeds by several orders of magnitude both the failure probability due to the target block errors or the failure probability of error correction blocks. A good estimate of this optimal code distance can be obtained by assuming the QEC steps are perfect. In this case, the probability of a logical Z_L error on either of the control blocks is simply given by the probability of accumulating more than $\lfloor d/2 \rfloor$ errors

$$p_{Z_L} = \sum_{k=\lfloor d/2 \rfloor + 1}^{d} \binom{d}{k} p'^k (1-p')^{d-k}$$

where $p' \approx dp$ is the probability that a given physical qubit of a logical control block is corrupted by a Z error during the circuits execution (the approximation $p' \approx dp$ is valid as far as $dp \ll 1$). This infidelity is plotted in plain lines, and, as expected, fits well the numerical values in the regime where the physical error probability p is small and the code distance d is large, for which the logical error is entirely set by

the accumulation of non-propagating errors.

We used this asymptotic formula to estimate the optimal code distance d and the associated logical error probability for a given physical Toffoli error probability, as it is precisely the region of the curves where the formula fits very well the numerical values. The results are plotted in the inset of Figure 4.8. As it can be observed, even without concatenation, a physical error probability of about .25% per Toffoli gate on cat-qubits yields logical error rates of about 10^{-10} with as few as 60 modes. Now, if the physical error probability per cat-qubit Toffoli gate is about 1%, the same logical error probability of 10^{-10} can be achieved with one level of concatenation, concatenating a 9-mode repetition code with a 60-mode one. Following the error model of the appendix achieved for the Toffoli implementation of [60], and assuming a two-photon dissipation rate of $\kappa_2/2\pi = 1$MHz, this physical error probability can be achieved for a cavity lifetime of 1ms. With the recent progress in 3D superconducting cavities, this long lifetime can be typically achieved with cylindrical postcavities [106].

4.3.2 Scheme 2 (without concatenation)

We perform the Monte-Carlo simulations of the circuit 3.9, using a circuit-based error model including the error models provided in the Table 4.1. The simulation results are plotted in Figure 4.9. These simulations indicate the existence of a threshold corresponding to a physical Toffoli error probability sightly below 3%. A typical physical error probability of 1% that can be achieved with the parameters of the previous subsection should result in a logical Toffoli error probability of 10^{-10} with as few as 90 data modes. This important overhead reduction, with respect to the previous concatenated case, comes at the expense of a fault-tolerant preparation of logical $|0\rangle_L$ states that will be consumed by the Steane EC protocol. This logical preparation can be performed by initializing each mode in the coherent state $|\alpha\rangle$ followed by d rounds of X_iX_{i+1} parity measurements and correction by MWPM. This requires to allocate a memory register in which the states are constantly prepared, maintained by EC, and consumed by logical Toffoli gates when needed.

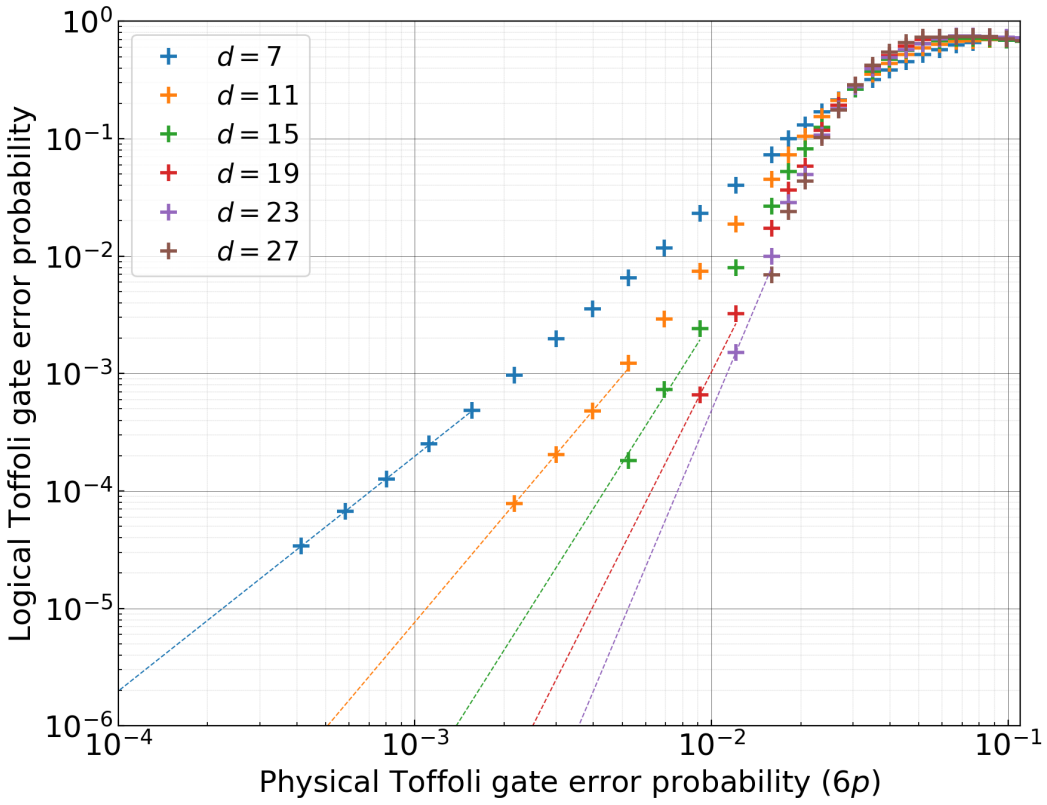


FIGURE 4.9: Monte Carlo simulations of the circuit of Figure 3.9 using a circuit-based error model. Here, we plot the error probability of the logical Toffoli gate as a function of the error probability of the physical cat qubit Toffoli gate. In the asymptotic regime where the physical error probability is small, the logical error probability now with $\lceil \frac{d}{4} \rceil$ instead of the usual $\lceil \frac{d}{2} \rceil$ that we get in the memory case. This is a consequence of the fact that the errors of a single physical qubit of the target block can spread to two different physical qubits within the same logical control block through the stabilizer measurements. Here, the curves are fit to the empirical scaling formula $A(\frac{p}{p_{th}})^{\frac{d+1}{4}}$.

Chapter 5

Conclusions and perspectives

5.1 Prior versus current state of the art

The work presented in this manuscript is the natural extension of the works on dissipative cat qubits initiated around 2012 in a collaboration between Yale university and the Quantic team of Inria Paris. The first proposal to use superpositions of coherent states as a qubit to protect against bit-flips [30] (1998), was adapted to the superconducting circuits [80] (2012), and then supplemented with a proposal to stabilize this encoding in a dissipative manner [87] (2013). In the foundational paper of our work, "Dynamically protected cat-qubits: a new paradigm for universal quantum computation" [87], it was shown how *operations* could be performed on these autonomously stabilized cat qubits (2013), opening the path to building a quantum computer based on these cat qubits. This proposal demonstrated how a universal set of gates could be performed on the cat qubits, necessarily including some operations that were not compatible with the noise bias of the cat qubits (see Chapter 2, subsection 2.1). In 2014, the autonomous stabilization of the cat qubit was experimentally demonstrated for the first time [78], followed in 2017 by the first demonstration [118] of a quantum gate on this cat qubit, the Rabi oscillations around the Z axis of the Bloch sphere.

In the large family of the bosonic codes, the cat qubit encoding stands out because the resulting qubit suffers from a highly biased noise. In 2007, a generic study demonstrated how fault-tolerant quantum computation could be performed on biased noise qubits [8]. By generic, we mean that this study is implementation-agnostic, *i.e* does not rely on a specific type of qubit but rather on a specific structure of the noise. The strategy follows in this work consists in embedding the noise-bias

qubits into a repetition code protecting exclusively against the dominant error. The distance of this repetition code is chosen such that the resulting logical qubit suffers from unbiased noise. Using this (very good) logical qubit to perform quantum computations requires to design a universal set of gates at the level of a repetition code. However, the set of physical operations that one shall use to design logical gates must crucially be restricted to *bias-preserving* operations, to comply with the particular demand of fault-tolerance of this scheme. The study of [8] being hardware-agnostic, the aforementioned restricted set of bias-preserving gates was limited to gates that either commuted with phase-flips, or operations that were trivially biased (see Chapter 2). This restricted set was actually not sufficient to build a universal set of gates at the level of the repetition code, such that concatenating with an additional CSS code and appending magic state preparation, distillation and injection was necessary in this case to construct a universal set of gates. This work was followed by others, including [129] which showed how the cost of magic state preparation could be reduced in the context of biased noise.

The combination of the generic path proposed in these works together with the construction of the biased noise cat qubits is actually a perfectly valid proposal to build a fault-tolerant and universal quantum computer, because all the bias-preserving operations required at the base level of [8] were included in the proposal of [87]. The main focus of this thesis has been to tailor the generic construction to the specific features of the dissipative cat qubits, with the following metrics in mind: reducing the overall hardware overhead, increasing the thresholds that need to be reached in experiments, and using building bricks that have either already been demonstrated experimentally, or are (hopefully) within reach in the next couple of years. The major simplification proposed in this work is the addition to a new class of bias-preserving topological gates on the dissipative cat qubits, obtained through a continuous deformation of the code space of the cat qubits. The idea behind this gate was first introduced in the context of the CNOT gate for the Kerr-cat qubits [103]. We adapted this idea to the particular dissipative implementation and extended it to the non-Clifford Toffoli gate (Chapter 2). Using the now larger set of bias-preserving gates, we constructed a universal set of logical gates at the level of the repetition code, without having to further concatenate. Also, in an attempt to circumvent the costly magic state preparation required in the reference work [8], we introduced a fault-tolerant logical Toffoli circuit as the logical non-Clifford resource (Chapter 3). The design of a non-transversal, yet fault-tolerant, logical gate turns out to be quite involved. Last, in Chapter 4, we performed a thorough numerical investigation of the performance of the entire construction. Here, the goal is two-fold. First, the thorough study of the repetition cat qubit as a quantum memory (and of transversal operations) sets the experimental requirements to meet within the next few years to demonstrate a fully protected repetition cat qubit, and (transversal) fault-tolerant logical gates on

repetition cat qubit. Second, the particular focus on the most costly gate, the logical non-Clifford Toffoli gate, sets the cost of the overall approach and gives a rough idea of the demanding requirements that need to be matched within the next decade.

Despite all of the theoretical and experimental efforts of the past decade in building dissipative cat qubits, including this work, the proposal remains extremely challenging from an engineering point of view. Yet, the first two decades of the 21st century, and more especially the past 5 to 10 years, have seen many private companies start a research activity in quantum computing. Among the companies relying on superconducting qubits, the leading architecture is undeniably the surface code with transmon qubits. However, the field of superconducting bosonic qubits is quickly evolving, triggering interest in proposals similar to this work. In particular, the start-up Alice&Bob was founded in 2020 to collaborate with academic groups on the specific effort towards a large scale quantum computer based on the repetition cat qubits proposed in this thesis. Very recently, a thorough comprehensive architectural study based on repetition cat qubits was proposed by the team of the AWS Center for Quantum Computing [26] and was presented as the blueprint for their experimental work in the following years. This massive theoretical work has filled in some of the gaps of our own work. Because of the important overlap with our own work, it would deserve a detailed comparison, which we unfortunately do not have time to include. Rather, we simply tried to include and acknowledge the contributions to the analytical errors models for our gates, and postpone the detailed comparison of the high-level proposal to later analysis.

5.2 Perspectives

In this section, we discuss the perspectives for improving the performance of the repetition cat qubit approach. Two main axis of research that needs to be addressed are the implementation of the physical gates and the design of an architecture compatible with a 2D implementation.

The implementation of the gates is of crucial importance to the overall performance of the scheme. Specifically, reducing the errors of the CNOT gate used for error correction will automatically result in a higher threshold, or, equivalently reduce the hardware overhead. In its current form, the CNOT implementation relies on both a continuous conditional deformation of the code space of the target cat qubit, with the addition of a feed-forward Hamiltonian to increase the fidelity of the gate. Unfortunately, the desired Hamiltonian that would exactly compensate the non-adiabatic phase-flip errors is not an experimentally feasible Hamiltonian. Therefore, an important axis of research is how well one can approximate this Hamiltonian. We note that because of the similarities between the CNOT and the Toffoli gate, improving the former might readily lead to a similar improvement of

the latter.

In this work, we have not considered the physical restrictions imposed by the particular experimental implementation of the scheme. Typically, a realistic scheme for large scale fault-tolerant quantum computation should possess the following features: a high accuracy threshold such that error rates well below this value can be achieved in the experiments, a universal set of logical gates that can be implemented with a reasonable resource overhead, and an architecture that can be scaled up to a size where the logical error rates match those needed for the targeted computation. The first and the last points are very strong assets of the surface code approach. Indeed, this code combines the advantage of a high accuracy threshold around 1% for a depolarizing noise model [105] and a 2D spatial arrangement of the physical qubits requiring only low-weight stabilizer measurements between nearest neighbours. In our approach, the transversal Clifford operations presented in Chapter 3 are compatible with a 2D architecture, using only couplings between neighbouring qubits. However, the two logical Toffoli circuits proposed in this work require an all-to-all coupling between the data cat qubits of the two logical control blocks. Within the particular circuit QED framework that we have in mind for the experimental implementation of repetition cat qubits, this kind of connectivity is not practical and poses a major challenge. Yet, it is worth noting that we anticipate very low logical error rates with only a few tens of cat qubits per logical qubit, which is a drastically lower overhead than those usually envisioned in other QEC schemes. Therefore, the general constraints on the connectivity graph of the physical qubits may be easier to satisfy for near term experiments involving a small number of cat qubits, yet achieving low logical error rates.

The optimal layout of a large scale quantum computer based on repetition cat qubits was not addressed in this work, so we conclude by mentioning two possible solutions in the next two subsections.

5.2.1 Towards a 2D architecture: SWAP gates

The connectivity graph for the first logical Toffoli circuit (3.3.1) can be made local by swapping the data qubits of the first control block appropriately, as depicted in Figure 5.1. Each physical cat qubit of a given repetition cat qubit is now coupled to a single cat qubit of another repetition cat qubit only. Yet, the intermediate rounds of error correction on the target block are still needed to prevent the propagation of errors. The particular ordering of the physical Toffoli gates in Figure 5.1 differs from the circular permutations previously considered. This particular choice corresponds to a permutation that can be implemented with parallel SWAP gates in two steps, independent of the code distance.

One may wonder whether the same trick can be applied to the second Toffoli circuit proposed in subsection 3.3.2, to produce a circuit both compatible with a 2D

architecture and with a phase-flip threshold without concatenation. The existence of a Toffoli circuit ordering that allows us both to measure constant weight Clifford operators for the intermediate error correction steps on the logical control blocks and that can be implemented using a constant depth circuit of SWAP gates is an open problem and requires further investigation.

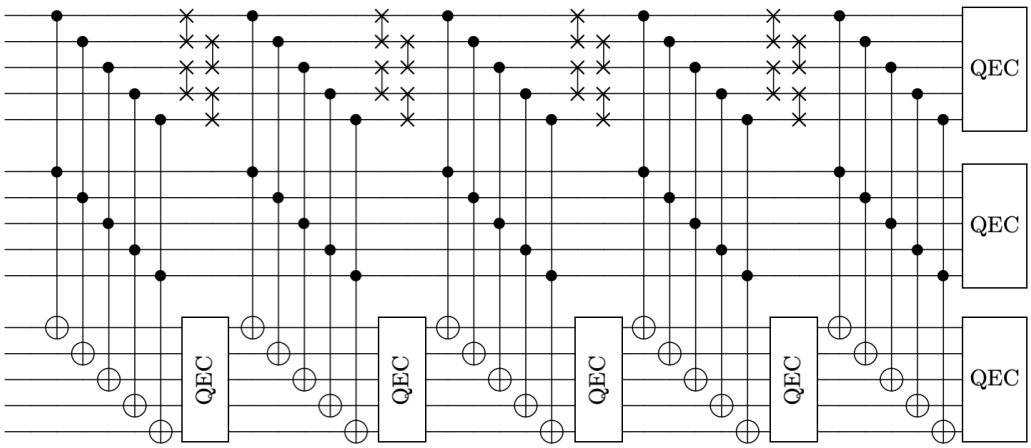


FIGURE 5.1: Logical Toffoli circuit for distance 5 repetition cat qubits including physical SWAP gates on the first control block during error correcting stage on the target. The physical SWAP gates ensure a great simplification of the connectivity graph for the implementation of the logical Toffoli gate.

5.2.2 Towards a 2D architecture: fault-tolerant magic state preparation and injection

The results presented in this subsection (unpublished) are partial results of the masters internship of François-Marie Le Régent [77] that I co-supervised with Mazyar Mirrahimi. A similar proposal is included in the AWS paper [26].

We have seen in Chapter 3 how to construct directly a logical Toffoli gate using physical bias-preserving Toffoli gates. However, the additional error correction steps required to make this non-transversal circuit fault-tolerant are quite involved, and can only be made local for the scheme achieving fault-tolerance with concatenation. Here, we review a different strategy to work around the Eastin-Knill theorem is to use non-unitary operations, *e.g* state injection based on quantum measurement induced teleportation. In particular, we emphasize the fact that this approach can be made compatible with a 2D architecture.

The protocol was first introduced in [55] (in the context of a general noise structure) and it contains two distinct steps:

- the preparation of a high fidelity "magic state" corresponding to the target logical gate

- the teleportation of the logical gate ("injection") into the code, consuming the magic state.

Usually, the teleportation step can be realized using fault-tolerant Clifford gates, which, in many schemes, can be implemented transversally. For this reason, the hard step in this approach is usually concentrated in the preparation of a magic state of high fidelity. In order to do so, this step is quite often divided in two sub-steps:

- the preparation of many copies of low fidelity magic states
- the "distillation" of these states, producing a single copy of the magic state of high fidelity by consuming the low fidelity magic states.

Here the specific bias-preserving features of our construction simplify considerably this scheme. We begin by describing the injection of a Toffoli magic state before discussing its preparation.

Toffoli magic state injection Following the recipe given in [55], a logical Toffoli gate can be injected in the code by consuming a "Toffoli magic state" [77] (see also [26]) denoted $|\mathrm{CCX}\rangle$ using the circuit depicted in Figure 5.2. We emphasize the fact that, in this circuit, all the operations are *logical* operations (every single line is representing a repetition cat qubit).

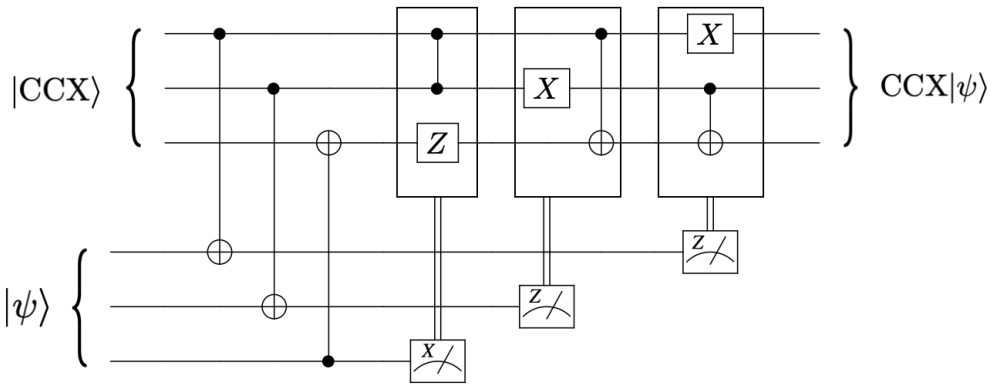


FIGURE 5.2: Injection circuit of a Toffoli magic state $|\mathrm{CCX}\rangle$ into a Toffoli gate, using logical Clifford operations (see *e.g* Exercise 10.68 p.488 of [91] for a detailed explanation of the construction of this circuit). The operation in the boxes are performed conditionnaly on the measurement outcomes being -1.

One can check by inspecting this circuit, and using the logical operation constructions of Chapter 3, that every gate needed in the injection circuit is transversal, thus compatible with a 2D architecture, except for the CZ gate. Similar to the Toffoli gate, there are two ways to make the logical CZ gate compatible with a 2D implementation: either by making the logical CZ circuit of Chapter 3 with concatenation local using SWAP gates, or by teleporting a logical CZ gate using the injection circuit

depicted in Figure 5.3, and the CZ magic state

$$|\mathrm{CZ}\rangle = \frac{1}{\sqrt{2}}(|0+\rangle + |1-\rangle).$$

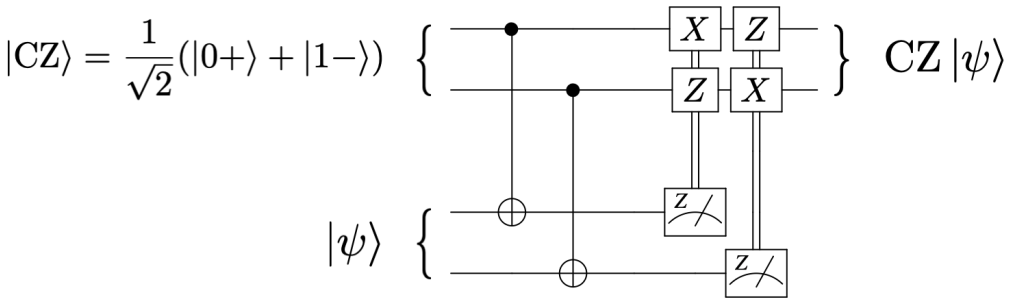


FIGURE 5.3: Injection of a CZ magic state producing a transversal logical circuit for the CZ gate on the repetition code.

The injection circuit of Figure 5.3 is composed exclusively of transversal gates, which is compatible with a 2D implementation. The fault-tolerant preparation of the $|CZ\rangle$ state is achieved with a QND measurement of the logical $X_L Z_L$ operator applied to the logical input state $|0+\rangle_L$, for which the preparation scheme is included in the logical operations of Chapter 3. The QND measurement of $X_L Z_L$ is achieved exactly like the QND measurement of $X_L \mathrm{CX}_L$ needed for the fault-tolerant preparation of the $|CCX\rangle$ state that we discuss below. More precisely, the measurement of the logical $X_L Z_L$ is obtained from Figure 5.5 by replacing every physical Toffoli gate by a physical CZ gate.

Toffoli magic state preparation The gates and operations used in the injection circuit are all Clifford operations. Thus, the non-Clifford nature of the whole circuit is inherited from the particular Toffoli magic state $|\mathrm{CCX}\rangle$. The Toffoli magic state is given by

$$|\mathrm{CCX}\rangle = \tfrac{1}{2}(|000\rangle + |010\rangle + |100\rangle + |111\rangle).$$

To understand why this state can be used as a non-Clifford resource, it is useful to look at the stabilizers of this state. The $|\mathrm{CCX}\rangle$ is the $+1$-eigenstate of the three Clifford operators

$$\begin{aligned} S_1 &= X_1 \mathrm{CX}_{2,3} \\ S_2 &= X_2 \mathrm{CX}_{1,3} \\ S_3 &= Z_3 \mathrm{CZ}_{1,2} \end{aligned}$$

and a natural method to prepare this state is to perform a QND measurement of these three stabilizers. Because these stabilizers are Clifford operators, the *measurement* of these operators is actually in the third level of the Clifford hierarchy,

thus, a non-Clifford operation.

The fault-tolerant preparation of the non-Clifford state $|\text{CCX}\rangle$ boils down to the fault-tolerant measurement of its stabilizers. Because these stabilizers are transversal, the generic method of [55] can be applied. Let us recall the idea behind this protocol to emphasize where the use of repetition cat qubits greatly simplifies this protocol.

The quantum non demolition measurement of a logical Hermitian operator U_L, admitting a transversal decomposition on the repetition code $U_L = \bigotimes_i U_i$ can be performed with an extra ancilla qubit and physical controlled-U_i operations as depicted in Figure 5.4 a).

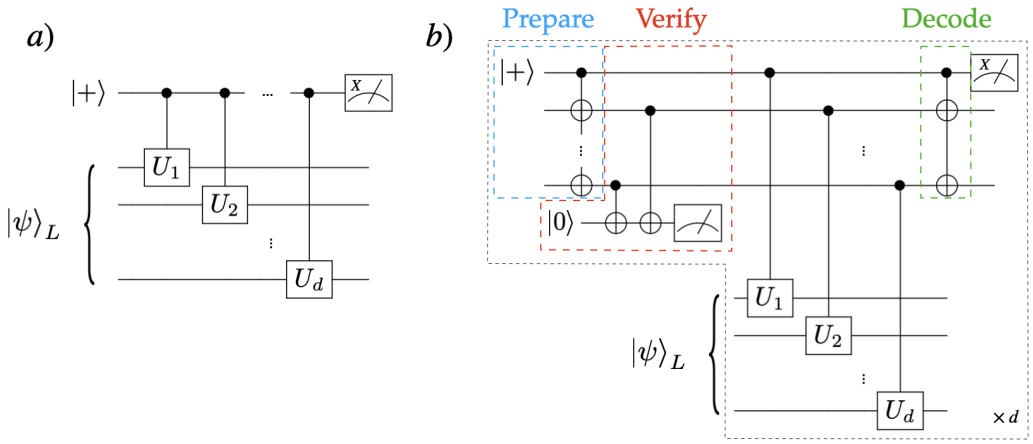


FIGURE 5.4: a) QND measurement circuit of the transversal Hermitian operator $U_L = \bigotimes_i U_i$. b) Fault-tolerant version of the circuit.

Because the ancilla qubit used is coupled to all of the data qubits of the code, this circuit is not fault-tolerant. The reason behind this is that a single X error can be converted to many (possibly d) U_i errors on the logical block. In general, this circuit can be made fault-tolerant by following the following steps [91], depicted in Figure 5.4 b). The single ancilla qubit is replaced by a block of ancilla qubit, prepared in the so-called CAT state (here we follow the usual denomination of the literature for this state, but we note that this must not be confused with the cat states defining the code space for the cat qubits)

$$|\text{CAT}\rangle = \frac{1}{\sqrt{2}}(|0\rangle^{\otimes d} + |1\rangle^{\otimes d}).$$

This can be achieved *e.g* by preparing the first ancilla qubit in the $|+\rangle$ state and applying $d-1$ CNOT gates between this ancilla qubit as the control and the others $d-1$ ancilla qubits of the block as the targets (blue box of Figure 5.4 b)). This preparation is not fault-tolerant, because a single error here can propagate

to many errors on the CAT state. Thus, before coupling this ancilla state with the system, an additional verification step is needed (only one of these checks is shown in the red box of Figure 5.4 b)). Whenever this verification step fails (one check measurement outcome is -1), the CAT state is discarded and the preparation is restarted. When the CAT state passes the verification step, the measurement is realized using transversal controlled-U_i operations. Then, the last step is to decode the output of the measurement (green box of Figure 5.4 c)). While this procedure ensures that a single error in the ancilla block cannot spread to more than one error on the data block, the fidelity of this logical measurement is typically rather low. Because the fidelity of this logical measurement directly impacts the fidelity of the logical magic state being prepared with this measurement (and hence, the fidelity of the logical gate that is teleported when consuming this logical magic state), it is crucial that the fidelity of this measurement can be made very reliable. Fortunataly, because the measurement of Figure 5.4 c) is both fault-tolerant and QND, its fidelity can be increased to the required accuracy by simply repeating the measurement d times.

We now detail how the use of cat qubits and bias-preserving gates simplifies significantly this construction. One can readily note that the circuit of Figure 5.4 a) is not fault-tolerant because a single bit-flip error of the ancilla can spread to many errors on the encoded data block. However, when this ancilla is a cat qubit, the probability of such an error is exponentially suppressed with the mean number of photons, while the remaining phase-flip error does not spread to the encoded block as it commutes with the controlled-U_i gates. This, combined with the fact that the controlled-U_i gates are themselves bias-preserving, ensures that the circuit in Figure 5.4 a) is already fault-tolerant for a repetition cat qubit.

However, the use a of an ancilla block encoded in the CAT state is still desirable to implement the controlled-U_i operations in a transversal manner. We emphasize that, in our scheme, the transversality is not required to prevent a spread of errors but rather, to benefit from the fact that the controlled-U_L can now be executed in a single time step, thus avoiding undesirable idling locations for the data cat qubits. Because the bias-preserving property ensures that errors cannot spread, the circuit using an ancilla block in the CAT state of Figure 5.4 b) is simplified. In particular, the verification step is not needed anymore, such that the protocol in now deterministic as it is not necessary to post-select the CAT state preparation. Furthermore, we note that the decoding step on the CAT state is actually unneeded, as one can equivalently perform an X measurement on all of the ancilla cat qubit and decode the collection of outcomes by computing the parity of the number of $+1$ outcomes. This follows

from the fact that the CAT state can be rewritten in the dual basis $|\pm\rangle$ as

$$|\mathrm{CAT}\rangle = \frac{1}{(\sqrt{2})^{d-1}} \sum_{j\in\{+,-\}^d, \#\{+1\} \text{ even}} |j\rangle.$$

In the case of a single ancilla qubit (5.4 a)), the state of the ancilla is phase-flipped for each data qubit i that is in a -1 eigenspace of the corresponding operator U_i. Equivalently, for the ancilla block qubit case (5.4 b)), it is the *parity* of the number of $+1$ outcomes which is flipped for each data qubit i that is in a -1 eigenspace of the corresponding operator U_i.

Putting the pieces back together, we depict in Figure 5.5 the circuit to fault-tolerantly prepare the Toffoli magic state $|\mathrm{CCX}\rangle$.

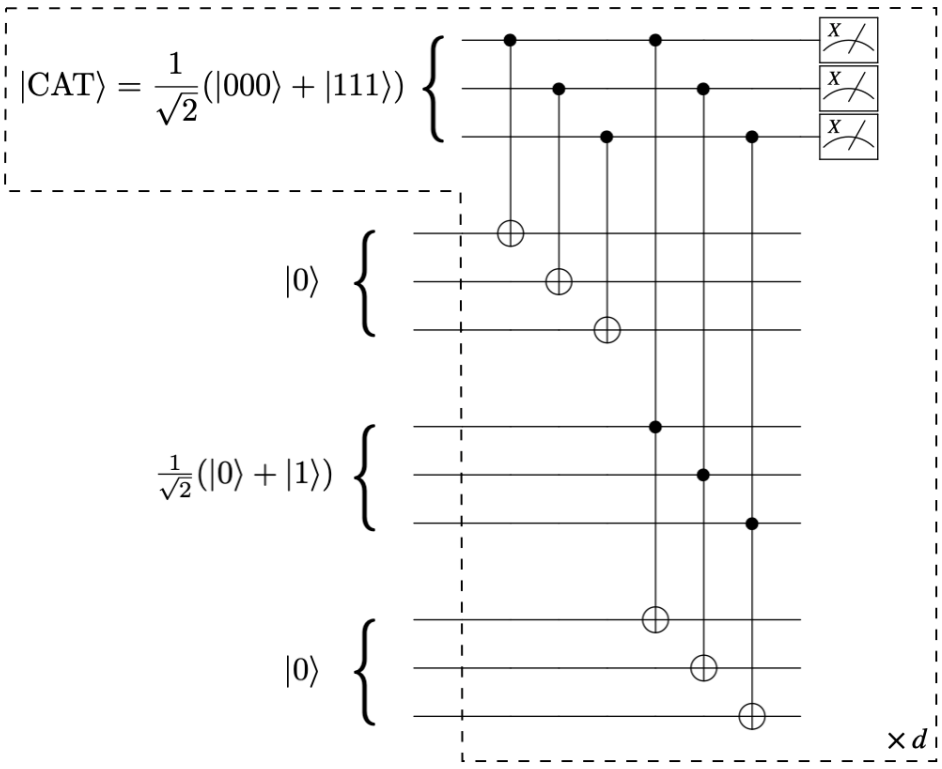


FIGURE 5.5: Fault-tolerant preparation of the $|\mathrm{CCX}\rangle$ state.

The specific choice in Figure 5.5 of the state on which the stabilizer S_1 is measured is because this state is both easy to prepare and can be decomposed as

$$|0\rangle|+\rangle|0\rangle = \tfrac{1}{\sqrt{2}}(|\mathrm{CCX}\rangle + Z_1|\mathrm{CCX}\rangle)$$

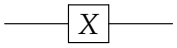
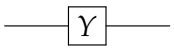
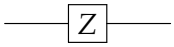
such that the measurement of S_1 yields either the $|\mathrm{CCX}\rangle$ state or the $Z_1|\mathrm{CCX}\rangle$ state (with equal probability).

The detailed analysis of the error rates of the logical Toffoli gate implemented via magic state, and the overhead reduction one could expect, is left to future work (the design of magic state factories based on this scheme is discussed in [26]). In particular, the design of a universal set of logical gates compatible with an experimentally scalable layout will be a major problem to address in the next few years.

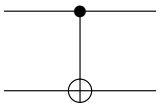
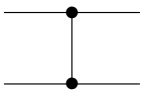
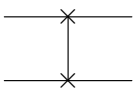
Appendix A

Quantum gates

Pauli gates

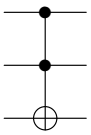
Gate	Circuit notation	Matrix representation
X gate		$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
Y gate		$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$
Z gate		$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

Clifford gates

Hadamard gate		$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
CNOT (CX) gate		$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$
Controlled-Z (CZ) gate		$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$
SWAP		$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

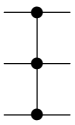
Non-Clifford gates

Toffoli (CCX)



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

CCZ



$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

Appendix B

Clifford hierarchy

The *Clifford hierarchy* was first introduced in [55]. This hierarchy is useful to classify quantum operations in a growing inclusion. The first two-levels (the Pauli and Clifford groups, respectively) contain operations that are not enough for universal quantum computation. Rather, by the Gottesman-Knill theorem, a quantum circuit that is composed entirely of operations in the first two levels can be efficiently simulated classically. The operations in the third level or higher levels (the non-Clifford operations), on the other hand, unlock the computational power of quantum computing. The exact structure of the hierarchy is not yet fully understood and is still the topic of current research [32, 107, 100].

The Clifford hierarchy is defined recursively. The first level, denoted $\mathcal{C}_1$, is the Pauli group. Then, a unitary U is said to be in the k-th level of the Clifford hierarchy ($k \leq 2$) if it maps Pauli operators to operators in the $(k-1)$-th level:

$$\mathcal{C}_k = \{U, U\mathcal{C}_1U^\dagger \subseteq \mathcal{C}_{k-1}\}.$$

The second level of the Clifford hierarchy, $\mathcal{C}_2$, contains the operators that map Pauli operators to Pauli operators (the so-called Clifford operators). The first two-levels of the Clifford hierarchy are the only one that have a group structure. Rather, the gates in the third or higher level of the Clifford hierarchy can "bootstrap" themselves up the Clifford hierarchy.

Examples the operations in the k-th level include:

- Rotations around a cardinal axis of the Bloch sphere of an angle $\frac{2\pi}{2^k}$
- Controlled-U operations with $U \in \mathcal{C}_{k-1}$
- Measurement of U with $U \in \mathcal{C}_{k-1}$.

The last fact is key in many magic states based schemes. In this thesis, it was illustrated by arguing that the Toffoli gate can be realized by measuring the Clifford operator $X \otimes CX$ (thus achieving the preparation of the Toffoli magic state) together with Clifford operations.

Appendix C

A no-go theorem for bias-preserving gates

The extra degree freedom associated to the complex amplitude α in the cat-qubits enables us to perform a set of non-trivial operations such as the CNOT or the Toffoli gate in bias-preserving manner (see Chapter 2). In this appendix, we will show the crucial role played by this extra degree of freedom. Indeed, we will prove that such gates cannot be performed in a bias-preserving manner with two-level systems. The analysis of this appendix should be extendable to the case of qubits encoded in qudits with a finite number of levels. However, as the dimension of the space in which the qubit is encoded increases, the bias could be approximately preserved.

We will focus on the case of the CNOT gate but a similar analysis can be performed for the Toffoli gate. Throughout this section, we will call $\mathcal{U}(4)$ the Lie group of unitary operators on two two-level systems and $\mathfrak{su}(4)$ the associated Lie algebra. We also assume that we are dealing with qubits that are only susceptible to phase-flip errors. We define a bias-preserving gate to be a gate that does not transform phase-flip errors to bit-flip ones. Here is a more precise definition.

Definition 2 *We call a unitary operation $U \in \mathcal{U}(4)$ bias-preserving, if*

$$[UZ_{1,2}U^\dagger, Z_{1,2}] = 0.$$

Indeed, the operators iZ_1 and iZ_2 and similarly $iUZ_{1,2}U^\dagger$ are members of the Lie algebra $\mathfrak{su}(4)$ and therefore can be written in the basis of two-qubit Pauli operators. It is therefore easy to see that the above condition is equivalent to saying that $UZ_{1,2}U^\dagger$ is a linear combination of the three Pauli operators Z_1, Z_2, Z_1Z_2. This e.g. means that

$$UZ_1U^\dagger = c_1Z_1 + c_2Z_2 + c_{12}Z_1Z_2 \Rightarrow UZ_1 = (c_1Z_1 + c_2Z_2 + c_{12}Z_1Z_2)U.$$

Therefore, a Z_1 error before the unitary operation can only lead to Z_1, Z_2 and Z_1Z_2 errors after the operation.

We note that, the CNOT or Toffoli gates are members of this set of bias-preserving operations. We will see however, that they cannot be realized in a bias-preserving manner. We have the following Lemma:

Lemma 1 *The set*

$$\mathcal{B} = \{U \in \mathcal{U}(2) \mid [U^\dagger Z_{1,2} U, Z_{1,2}] = 0\}$$

is a Lie subgroup of $\mathcal{U}(2)$.

Proof. We start by proving that $\mathcal{B}$ is a group. It clearly includes the identity operator. Also if $U_1, U_2 \in \mathcal{B}$, we have

$$U_1 Z_1 U_1^\dagger = c_1 Z_1 + c_2 Z_2 + c_{12} Z_1 Z_2.$$

Therefore,

$$\begin{aligned} U_2 U_1 Z_1 U_1^\dagger U_2^\dagger &= c_1 U_2 Z_1 U_2^\dagger + c_2 U_2 Z_2 U_2^\dagger + c_{12} U_2 Z_1 Z_2 U_2^\dagger \\ &= c_1 U_2 Z_1 U_2^\dagger + c_2 U_2 Z_2 U_2^\dagger + c_{12} U_2 Z_1 U_2^\dagger U_2 Z_2 U_2^\dagger \\ &= \tilde{c}_1 Z_1 + \tilde{c}_2 Z_2 + \tilde{c}_{12} Z_1 Z_2. \end{aligned}$$

Therefore $U_1 U_2 \in \mathcal{B}$. We only need to prove that if $U \in \mathcal{B}$, then $U^\dagger \in \mathcal{B}$. We have

$$\begin{aligned} U Z_1 U^\dagger &= r_1 Z_1 + r_2 Z_2 + r_{12} Z_1 Z_2, \\ U Z_2 U^\dagger &= s_1 Z_1 + s_2 Z_2 + s_{12} Z_1 Z_2. \end{aligned}$$

We note that r_1, r_2, r_{12} cannot simultaneously vanish (similarly for s_1, s_2, s_{12}). Also, we note that the two vectors (r_1, r_2, r_{12}) and (s_1, s_2, s_{12}) are necessarily orthogonal. In order to see this, we note that by multiplying the above equations and taking the trace of both sides we get

$$r_1 s_1 + r_2 s_2 + r_{12} s_{12} = 0.$$

Now we multiply the above equations from left by $U^\dagger$ and from right by U:

$$\begin{aligned} Z_1 &= r_1 U^\dagger Z_1 U + r_2 U^\dagger Z_2 U + r_{12} U^\dagger Z_1 Z_2 U, \\ Z_2 &= s_1 U^\dagger Z_1 U + s_2 U^\dagger Z_2 U + s_{12} U^\dagger Z_1 Z_2 U. \end{aligned}$$

Furthermore, the product of the above equations give

$$Z_1 Z_2 = (r_2 s_{12} + s_2 r_{12}) Z_1 + (r_1 s_{12} + s_1 r_{12}) Z_2 + (r_1 s_2 + s_1 r_2) Z_1 Z_2.$$

We note that this can not be a linear combination of Z_1 and Z_2, which means that the vector $(r_2 s_{12} + s_2 r_{12}, r_1 s_{12} + s_1 r_{12}, r_1 s_2 + s_1 r_2)$ is linearly independent from the orthogonal vectors (r_1, r_2, r_{12}) and (s_1, s_2, s_{12}). This means the matrix provided by these vectors can be inverted and therefore the operators $U^\dagger Z_1 U$ and $U^\dagger Z_2 U$ can also be written as a linear combination of $Z_1, Z_2, Z_1 Z_2$. Thus, $U^\dagger \in \mathcal{B}$ and we have therefore shown that $\mathcal{B}$ is a sub-group of $\mathcal{U}(4)$.

In order to prove that it is a Lie sub-group, we note that $f(U) = \Big([U^\dagger Z_j U, Z_k]\Big)_{j,k=1,2}$ is a continuous function of U. Furthermore $\mathcal{B}$ is defined as the

pre-image of the set $\{(0,0,0,0)\}$ which is a closed set. Therefore $\mathcal{B}$ is topologically closed. A topologically closed sub-group of a Lie group is a Lie sub-group (Cartan's theorem) and therefore the proof is complete. □

We now follow ideas that are very similar to the analysis of [37]. The Lie group $\mathcal{B}$ can be partitioned into cosets of the connected component of the identity that we call $\mathcal{C}$. $\mathcal{C}$ is itself a Lie sub-group of $\mathcal{B}$. This set of cosets is the quotient group $\mathcal{B}/\mathcal{C}$. The main result of this appendix can be resumed in the following theorem:

Theorem 6 *The unitary operator CNOT is not a member of $\mathcal{C}$. This means that CNOT cannot be continuously obtained from identity in a bias-preserving process.*

Proof. As $\mathcal{C}$ is a connected Lie group, any element $C \in \mathcal{C}$ can be written as $C = \Pi_k e^{iD_k}$, where D_k is in $\mathfrak{c}$, the Lie algebra of $\mathcal{C}$. Now, note that for any $\epsilon \in \mathbb{R}$ and any $D \in \mathfrak{c}$, the operator $e^{i\epsilon D}$ is also in $\mathcal{C}$ and therefore satisfies

$$[e^{i\epsilon D} Z_{1,2} e^{-i\epsilon D}, Z_{1,2}] = 0.$$

Taking the derivative with respect to ϵ at $\epsilon = 0$, we get

$$[[D, Z_j], Z_k] = 0, \qquad j,k = 1,2.$$

Noting that D is necessarily a linear combination to two-qubit Pauli operators, it is the same for $[D, Z_j]$ and therefore

$$\begin{aligned}[D, Z_1] &= r_1 Z_1 + r_2 Z_2 + r_{12} Z_1 Z_2,\\ [D, Z_2] &= s_1 Z_1 + s_2 Z_2 + s_{12} Z_1 Z_2.\end{aligned}$$

The only possibility for such a combination is that all the coefficients vanish. Therefore $[D, Z_{1,2}] = 0$, or equivalently

$$D = c_0 I + c_1 Z_1 + c_2 Z_2 + c_{12} Z_1 Z_2.$$

Therefore, the Lie algebra $\mathfrak{c}$ is spanned by $I, Z_1, Z_2, Z_1 Z_2$, which means that the associated Lie group does not include CNOT. □

Bibliography

[1] Scott Aaronson and Daniel Gottesman. "Improved simulation of stabilizer circuits". In: *Phys. Rev. A* 70 (5 Nov. 2004), p. 052328. DOI: 10.1103/PhysRevA.70.052328. URL: https://link.aps.org/doi/10.1103/PhysRevA.70.052328.

[2] Dorit Aharonov and Michael Ben-Or. "Fault-Tolerant Quantum Computation with Constant Error Rate". In: *SIAM Journal on Computing* 38.4 (Jan. 2008), pp. 1207–1282. DOI: 10.1137/s0097539799359385. URL: https://doi.org/10.1137/s0097539799359385.

[3] Charlene Ahn, Andrew C. Doherty, and Andrew J. Landahl. "Continuous quantum error correction via quantum feedback control". In: *Physical Review A* 65.4 (Mar. 2002). DOI: 10.1103/physreva.65.042301. URL: https://doi.org/10.1103/physreva.65.042301.

[4] Victor V. Albert. *Lindbladians with multiple steady states: theory and applications*. 2018. arXiv: 1802.00010 [quant-ph].

[5] Victor V. Albert and Liang Jiang. "Symmetries and conserved quantities in Lindblad master equations". In: *Physical Review A* 89.2 (Feb. 2014). DOI: 10.1103/physreva.89.022118. URL: https://doi.org/10.1103/physreva.89.022118.

[6] Victor V. Albert et al. "Geometry and Response of Lindbladians". In: *Physical Review X* 6.4 (Nov. 2016). DOI: 10.1103/physrevx.6.041031. URL: https://doi.org/10.1103/physrevx.6.041031.

[7] Victor V. Albert et al. "Performance and structure of single-mode bosonic codes". In: *Physical Review A* 97.3 (Mar. 2018). DOI: 10.1103/physreva.97.032346. URL: https://doi.org/10.1103/physreva.97.032346.

[8] Panos Aliferis and John Preskill. "Fault-tolerant quantum computation against biased noise". In: *Physical Review A* 78.5 (Nov. 2008). DOI: 10.1103/physreva.78.052331. URL: https://doi.org/10.1103/physreva.78.052331.

[9] P Aliferis et al. "Fault-tolerant computing with biased-noise superconducting qubits: a case study". In: *New Journal of Physics* 11.1 (Jan. 2009), p. 013061. DOI: 10.1088/1367-2630/11/1/013061. URL: https://doi.org/10.1088/1367-2630/11/1/013061.

[10] Jonas T. Anderson, Guillaume Duclos-Cianci, and David Poulin. "Fault-Tolerant Conversion between the Steane and Reed-Muller Quantum Codes". In: *Physical Review Letters* 113.8 (Aug. 2014). DOI: 10.1103/physrevlett.113.080501. URL: https://doi.org/10.1103/physrevlett.113.080501.

[11] J. Pablo Bonilla Ataides et al. *The XZZX Surface Code*. 2020. arXiv: 2009.07851 [quant-ph].

[12] R. Azouit, A. Sarlette, and P. Rouchon. "Adiabatic elimination for open quantum systems with effective Lindblad master equations". In: *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, Dec. 2016. DOI: 10.1109/cdc.2016.7798963. URL: https://doi.org/10.1109/cdc.2016.7798963.

[13] R Azouit et al. "Towards generic adiabatic elimination for bipartite open quantum systems". In: *Quantum Science and Technology* 2.4 (Sept. 2017), p. 044011. DOI: 10.1088/2058-9565/aa7f3f. URL: https://doi.org/10.1088/2058-9565/aa7f3f.

[14] Dave Bacon. "Operator quantum error-correcting subsystems for self-correcting quantum memories". In: *Physical Review A* 73.1 (Jan. 2006). DOI: 10.1103/physreva.73.012340. URL: https://doi.org/10.1103/physreva.73.012340.

[15] P. Bertet et al. "Direct Measurement of the Wigner Function of a One-Photon Fock State in a Cavity". In: *Phys. Rev. Lett.* 89 (20 Oct. 2002), p. 200402. DOI: 10.1103/PhysRevLett.89.200402. URL: https://link.aps.org/doi/10.1103/PhysRevLett.89.200402.

[16] "Blossom V: a new implementation of a minimum cost perfect matching algorithm". In: *Mathematical Programming Computation* 1.1 (2009), pp. 43–67. DOI: 10.1007/s12532-009-0002-8. URL: https://doi.org/10.1007/s12532-009-0002-8.

[17] Héctor Bombın. "Dimensional jump in quantum error correction". In: *New Journal of Physics* 18.4 (Apr. 2016), p. 043038. DOI: 10.1088/1367-2630/18/4/043038. URL: https://doi.org/10.1088/1367-2630/18/4/043038.

[18] Héctor Bombın. "Gauge color codes: optimal transversal gates and gauge fixing in topological stabilizer codes". In: *New Journal of Physics* 17.8 (Aug. 2015), p. 083002. DOI: 10.1088/1367-2630/17/8/083002. URL: https://doi.org/10.1088/1367-2630/17/8/083002.

[19] Sergey Bravyi and Alexei Kitaev. "Universal quantum computation with ideal Clifford gates and noisy ancillas". In: *Physical Review A* 71.2 (Feb. 2005). DOI: 10.1103/physreva.71.022316. URL: https://doi.org/10.1103/physreva.71.022316.

[20] Peter Brooks and John Preskill. "Fault-tolerant quantum computation with asymmetric Bacon-Shor codes". In: *Physical Review A* 87.3 (Mar. 2013). DOI: 10.1103/physreva.87.032310. URL: https://doi.org/10.1103/physreva.87.032310.

[21] Benjamin J. Brown. "A fault-tolerant non-Clifford gate for the surface code in two dimensions". In: *Science Advances* 6.21 (May 2020), eaay4929. DOI: 10.1126/sciadv.aay4929. URL: https://doi.org/10.1126/sciadv.aay4929.

[22] Benjamin J. Brown, Naomi H. Nickerson, and Dan E. Browne. "Fault-tolerant error correction with the gauge color code". In: *Nature Communications* 7.1 (July 2016). DOI: 10.1038/ncomms12302. URL: https://doi.org/10.1038/ncomms12302.

[23] P. Campagne-Ibarcq et al. "Quantum error correction of a qubit encoded in grid states of an oscillator". In: *Nature* 584.7821 (Aug. 2020), pp. 368–372. DOI: 10.1038/s41586-020-2603-3. URL: https://doi.org/10.1038%2Fs41586-020-2603-3.

[24] Christopher Chamberland and Michael E. Beverland. "Flag fault-tolerant error correction with arbitrary distance codes". In: *Quantum* 2 (Feb. 2018), p. 53. ISSN: 2521-327X. DOI: 10.22331/q-2018-02-08-53. URL: https://doi.org/10.22331/q-2018-02-08-53.

[25] Christopher Chamberland and Andrew W. Cross. "Fault-tolerant magic state preparation with flag qubits". In: *Quantum* 3 (May 2019), p. 143. DOI: 10.22331/q-2019-05-20-143. URL: https://doi.org/10.22331/q-2019-05-20-143.

[26] Christopher Chamberland et al. *Building a fault-tolerant quantum computer using concatenated cat codes*. 2020. arXiv: 2012.04108 [quant-ph].

[27] Rui Chao and Ben W. Reichardt. "Quantum Error Correction with Only Two Extra Qubits". In: *Phys. Rev. Lett.* 121 (5 Aug. 2018), p. 050502. DOI: 10.1103/PhysRevLett.121.050502. URL: https://link.aps.org/doi/10.1103/PhysRevLett.121.050502.

[28] Lilian Childress and Ronald Hanson. "Diamond NV centers for quantum computing and quantum networks". In: *MRS Bulletin* 38.02 (Feb. 2013), pp. 134–138. DOI: 10.1557/mrs.2013.20. URL: https://doi.org/10.1557/mrs.2013.20.

[29] Isaac L. Chuang, Debbie W. Leung, and Yoshihisa Yamamoto. "Bosonic quantum codes for amplitude damping". In: *Physical Review A* 56.2 (Aug. 1997), pp. 1114–1125. DOI: 10.1103/physreva.56.1114. URL: https://doi.org/10.1103/physreva.56.1114.

[30] P. T. Cochrane, G. J. Milburn, and W. J. Munro. "Macroscopically distinct quantum-superposition states as a bosonic code for amplitude damping". In: *Physical Review A* 59.4 (Apr. 1999), pp. 2631–2634. DOI: 10.1103/physreva.59.2631. URL: https://doi.org/10.1103/physreva.59.2631.

[31] Joachim Cohen. "Autonomous quantum error correction with superconducting qubits". Theses. PSL Research University, Feb. 2017. URL: https://tel.archives-ouvertes.fr/tel-01545186.

[32] Shawn X. Cui, Daniel Gottesman, and Anirudh Krishna. "Diagonal gates in the Clifford hierarchy". In: *Physical Review A* 95.1 (Jan. 2017). DOI: 10.1103/physreva.95.012329. URL: https://doi.org/10.1103/physreva.95.012329.

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

[34] E. W. Dijkstra. "A note on two problems in connexion with graphs". In: *Numerische Mathematik* 1.1 (Dec. 1959), pp. 269–271. DOI: 10.1007/bf01386390. URL: https://doi.org/10.1007/bf01386390.

[35] Guillaume Duclos-Cianci and David Poulin. "Reducing the quantum-computing overhead with complex gate distillation". In: *Phys. Rev. A* 91 (4 Apr. 2015), p. 042315. DOI: 10.1103/PhysRevA.91.042315. URL: https://link.aps.org/doi/10.1103/PhysRevA.91.042315.

[36] Guillaume Duclos-Cianci and Krysta M. Svore. "Distillation of nonstabilizer states for universal quantum computation". In: *Physical Review A* 88.4 (Oct. 2013). DOI: 10.1103/physreva.88.042325. URL: https://doi.org/10.1103/physreva.88.042325.

[37] Bryan Eastin and Emanuel Knill. "Restrictions on Transversal Encoded Quantum Gate Sets". In: *Physical Review Letters* 102.11 (Mar. 2009). DOI: 10.1103/physrevlett.102.110502. URL: https://doi.org/10.1103/physrevlett.102.110502.

[38] Jack Edmonds. "Paths, Trees, and Flowers". In: *Canadian Journal of Mathematics* 17 (1965), pp. 449–467. DOI: 10.4153/CJM-1965-045-4.

[39] Laird Egan et al. *Fault-Tolerant Operation of a Quantum Error-Correction Code*. 2020. arXiv: 2009.11482 [quant-ph].

[40] C. Flühmann et al. "Encoding a qubit in a trapped-ion mechanical oscillator". In: *Nature* 566.7745 (Feb. 2019), pp. 513–517. DOI: 10.1038/s41586-019-0960-6. URL: https://doi.org/10.1038/s41586-019-0960-6.

[41] Austin G. Fowler. "Coping with qubit leakage in topological codes". In: *Physical Review A* 88.4 (Oct. 2013). DOI: 10.1103/physreva.88.042308. URL: https://doi.org/10.1103/physreva.88.042308.

[42] Austin G. Fowler. *Minimum weight perfect matching of fault-tolerant topological quantum error correction in average $O(1)$ parallel time*. 2014. arXiv: 1307.1740 [quant-ph].

[43] Austin G. Fowler, Simon J. Devitt, and Cody Jones. "Surface code implementation of block code state distillation". In: *Scientific Reports* 3.1 (June 2013). DOI: 10.1038/srep01939. URL: https://doi.org/10.1038/srep01939.

[44] Austin G. Fowler, Adam C. Whiteside, and Lloyd C. L. Hollenberg. "Towards Practical Classical Processing for the Surface Code". In: *Phys. Rev. Lett.* 108 (18 May 2012), p. 180501. DOI: 10.1103/PhysRevLett.108.180501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.108.180501.

[45] Austin G. Fowler, Adam C. Whiteside, and Lloyd C. L. Hollenberg. "Towards practical classical processing for the surface code: Timing analysis". In: *Physical Review A* 86.4 (Oct. 2012). DOI: 10.1103/physreva.86.042313. URL: https://doi.org/10.1103/physreva.86.042313.

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

[47] Austin G. Fowler et al. "Topological Code Autotune". In: *Phys. Rev. X* 2 (4 Oct. 2012), p. 041003. DOI: 10.1103/PhysRevX.2.041003. URL: https://link.aps.org/doi/10.1103/PhysRevX.2.041003.

[48] Kosuke Fukui, Akihisa Tomita, and Atsushi Okamoto. "Analog Quantum Error Correction with Encoding a Qubit into an Oscillator". In: *Physical Review Letters* 119.18 (Nov. 2017). DOI: 10.1103/physrevlett.119.180507. URL: https://doi.org/10.1103/physrevlett.119.180507.

[49] Kosuke Fukui et al. "High-Threshold Fault-Tolerant Quantum Computation with Analog Quantum Error Correction". In: *Physical Review X* 8.2 (May 2018). DOI: 10.1103/physrevx.8.021054. URL: https://doi.org/10.1103/physrevx.8.021054.

[50] Yvonne Y. Gao et al. "Programmable Interference between Two Microwave Quantum Memories". In: *Physical Review X* 8.2 (June 2018). DOI: 10.1103/physrevx.8.021073. URL: https://doi.org/10.1103/physrevx.8.021073.

[51] K. Geerlings et al. "Demonstrating a Driven Reset Protocol for a Superconducting Qubit". In: *Physical Review Letters* 110.12 (Mar. 2013). DOI: 10.1103/physrevlett.110.120501. URL: https://doi.org/10.1103/physrevlett.110.120501.

[52] Craig Gidney and Austin G. Fowler. "Efficient magic state factories with a catalyzed |CCZ to 2|T transformation". In: *Quantum* 3 (Apr. 2019), p. 135. DOI: 10.22331/q-2019-04-30-135. URL: https://doi.org/10.22331/q-2019-04-30-135.

[53] Daniel Gottesman. "The Heisenberg Representation of Quantum Computers". In: *arXiv e-prints*, quant-ph/9807006 (July 1998), quant–ph/9807006. arXiv: quant-ph/9807006 [quant-ph].

[54] Daniel Eric Gottesman. "Stabilizer Codes and Quantum Error Correction". en. PhD thesis. 1997. DOI: 10.7907/RZR7-DT72. URL: https://resolver.caltech.edu/CaltechETD:etd-07162004-113028.

[55] Daniel Gottesman and Isaac L. Chuang. "Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations". In: *Nature* 402.6760 (Nov. 1999), pp. 390–393. DOI: 10.1038/46503. URL: https://doi.org/10.1038/46503.

[56] Daniel Gottesman, Alexei Kitaev, and John Preskill. "Encoding a qubit in an oscillator". In: *Physical Review A* 64.1 (June 2001). DOI: 10.1103/physreva.64.012310. URL: https://doi.org/10.1103/physreva.64.012310.

[57] A. Grimm et al. "Stabilization and operation of a Kerr-cat qubit". In: *Nature* 584.7820 (Aug. 2020), pp. 205–209. DOI: 10.1038/s41586-020-2587-z. URL: https://doi.org/10.1038/s41586-020-2587-z.

[58] Arne L. Grimsmo, Joshua Combes, and Ben Q. Baragiola. "Quantum Computing with Rotation-Symmetric Bosonic Codes". In: *Physical Review X* 10.1 (Mar. 2020). DOI: 10.1103/physrevx.10.011058. URL: https://doi.org/10.1103/physrevx.10.011058.

[59] Jérémie Guillaud and Mazyar Mirrahimi. *Error Rates and Resource Overheads of Repetition Cat Qubits*. 2020. arXiv: 2009.10756 [quant-ph].

[60] Jérémie Guillaud and Mazyar Mirrahimi. "Repetition Cat Qubits for Fault-Tolerant Quantum Computation". In: *Physical Review X* 9.4 (Dec. 2019). DOI: 10.1103/physrevx.9.041053. URL: https://doi.org/10.1103/physrevx.9.041053.

[61] Jeongwan Haah and Matthew B. Hastings. "Codes and Protocols for Distilling T, controlled-S, and Toffoli Gates". In: *Quantum* 2 (June 2018), p. 71. DOI: 10.22331/q-2018-06-07-71. URL: https://doi.org/10.22331/q-2018-06-07-71.

[62] Lisa Hänggli, Margret Heinze, and Robert König. "Enhanced noise resilience of the surface–Gottesman-Kitaev-Preskill code via designed bias". In: *Physical Review A* 102.5 (Nov. 2020). DOI: 10.1103/physreva.102.052408. URL: https://doi.org/10.1103/physreva.102.052408.

[63] Aram W. Harrow, Benjamin Recht, and Isaac L. Chuang. "Efficient discrete approximations of quantum gates". In: *Journal of Mathematical Physics* 43.9 (Sept. 2002), pp. 4445–4451. DOI: 10.1063/1.1495899. URL: https://doi.org/10.1063/1.1495899.

[64] L. Hu et al. "Quantum error correction and universal gate set operation on a binomial bosonic logical qubit". In: *Nature Physics* 15.5 (Feb. 2019), pp. 503–508. DOI: 10.1038/s41567-018-0414-3. URL: https://doi.org/10.1038/s41567-018-0414-3.

[65] Tomas Jochym-O'Connor and Raymond Laflamme. "Using Concatenated Quantum Codes for Universal Fault-Tolerant Quantum Gates". In: *Physical Review Letters* 112.1 (Jan. 2014). DOI: 10.1103/physrevlett.112.010505. URL: https://doi.org/10.1103/physrevlett.112.010505.

[66] Atharv Joshi, Kyungjoo Noh, and Yvonne Y. Gao. *Quantum information processing with bosonic qubits in circuit QED*. 2020. arXiv: 2008.13471 [quant-ph].

[67] J. Kelly et al. "State preservation by repetitive error detection in a superconducting quantum circuit". In: *Nature* 519 (Mar. 2015), 66 EP -. URL: https://doi.org/10.1038/nature14270.

[68] Gerhard Kirchmair et al. "Observation of quantum state collapse and revival due to the single-photon Kerr effect". In: *Nature* 495.7440 (Mar. 2013), pp. 205–209. DOI: 10.1038/nature11902. URL: https://doi.org/10.1038/nature11902.

[69] A. Yu. Kitaev. "Quantum computations: algorithms and error correction". In: *Uspekhi Matematicheskikh Nauk* 52.6 (1997), pp. 53–112. DOI: 10.4213/rm892. URL: https://doi.org/10.4213/rm892.

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

[71] Morten Kjaergaard et al. "Superconducting Qubits: Current State of Play". In: *Annual Review of Condensed Matter Physics* 11.1 (Mar. 2020), pp. 369–395. DOI: 10.1146/annurev-conmatphys-031119-050605. URL: https://doi.org/10.1146/annurev-conmatphys-031119-050605.

[72] E. Knill. "Quantum computing with realistically noisy devices". In: *Nature* 434.7029 (Mar. 2005), pp. 39–44. DOI: 10.1038/nature03350. URL: https://doi.org/10.1038/nature03350.

[73] E. Knill. "Resilient Quantum Computation". In: *Science* 279.5349 (Jan. 1998), pp. 342–345. DOI: 10.1126/science.279.5349.342. URL: https://doi.org/10.1126/science.279.5349.342.

[74] Emanuel Knill, Raymond Laflamme, and Wojciech H. Zurek. "Resilient quantum computation: error models and thresholds". In: *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 454.1969 (Jan. 1998), pp. 365–384. DOI: 10.1098/rspa.1998.0166. URL: https://doi.org/10.1098/rspa.1998.0166.

[75] P. Krantz et al. "A quantum engineer's guide to superconducting qubits". In: *Applied Physics Reviews* 6.2 (June 2019), p. 021318. DOI: 10.1063/1.5089550. URL: https://doi.org/10.1063/1.5089550.

[76] Aleksander Kubica and Michael E. Beverland. "Universal transversal gates with color codes: A simplified approach". In: *Physical Review A* 91.3 (Mar. 2015). DOI: 10.1103/physreva.91.032330. URL: https://doi.org/10.1103/physreva.91.032330.

[77] François-Marie Le Régent. *Quantum Operations on repetition cat qubits*. Master Theses. June 2020.

[78] Z. Leghtas et al. "Confining the state of light to a quantum manifold by engineered two-photon loss". In: *Science* 347.6224 (Feb. 2015), pp. 853–857. DOI: 10.1126/science.aaa2085. URL: https://doi.org/10.1126/science.aaa2085.

[79] Zaki Leghtas et al. "Deterministic protocol for mapping a qubit to coherent state superpositions in a cavity". In: *Physical Review A* 87.4 (Apr. 2013). DOI: 10.1103/physreva.87.042315. URL: https://doi.org/10.1103/physreva.87.042315.

[80] Zaki Leghtas et al. "Hardware-Efficient Autonomous Quantum Memory Protection". In: *Physical Review Letters* 111.12 (Sept. 2013). DOI: 10.1103/physrevlett.111.120501. URL: https://doi.org/10.1103/physrevlett.111.120501.

[81] Raphaël Lescanne. "Engineering Multi-Photon Dissipation in Superconducting Circuits for Quantum Error Correction". Theses. Ecole Normale Supérieure de Paris, Feb. 2020.

[82] Raphaël Lescanne et al. "Escape of a Driven Quantum Josephson Circuit into Unconfined States". In: *Physical Review Applied* 11.1 (Jan. 2019). DOI: 10.1103/physrevapplied.11.014030. URL: https://doi.org/10.1103/physrevapplied.11.014030.

[83] Raphaël Lescanne et al. "Exponential suppression of bit-flips in a qubit encoded in an oscillator". In: *Nature Physics* 16.5 (Mar. 2020), pp. 509–513. DOI: 10.1038/s41567-020-0824-x. URL: https://doi.org/10.1038/s41567-020-0824-x.

[84] Daniel Litinski. "Magic State Distillation: Not as Costly as You Think". In: *Quantum* 3 (Dec. 2019), p. 205. DOI: 10.22331/q-2019-12-02-205. URL: https://doi.org/10.22331/q-2019-12-02-205.

[85] L. G. Lutterbach and L. Davidovich. "Method for Direct Measurement of the Wigner Function in Cavity QED and Ion Traps". In: *Phys. Rev. Lett.* 78 (13 Mar. 1997), pp. 2547–2550. DOI: 10.1103/PhysRevLett.78.2547. URL: https://link.aps.org/doi/10.1103/PhysRevLett.78.2547.

[86] Marios H. Michael et al. "New Class of Quantum Error-Correcting Codes for a Bosonic Mode". In: *Physical Review X* 6.3 (July 2016). DOI: 10.1103/physrevx.6.031006. URL: https://doi.org/10.1103/physrevx.6.031006.

[87] Mazyar Mirrahimi et al. "Dynamically protected cat-qubits: a new paradigm for universal quantum computation". In: *New Journal of Physics* 16.4 (Apr. 2014), p. 045014. DOI: 10.1088/1367-2630/16/4/045014. URL: https://doi.org/10.1088%2F1367-2630%2F16%2F4%2F045014.

[88] S O Mundhada et al. "Generating higher-order quantum dissipation from lower-order parametric processes". In: *Quantum Science and Technology* 2.2 (May 2017), p. 024005. DOI: 10.1088/2058-9565/aa6e9d. URL: https://doi.org/10.1088/2058-9565/aa6e9d.

[89] S.O. Mundhada et al. "Experimental Implementation of a Raman-Assisted Eight-Wave Mixing Process". In: *Phys. Rev. Applied* 12 (5 Nov. 2019), p. 054051. DOI: 10.1103/PhysRevApplied.12.054051. URL: https://link.aps.org/doi/10.1103/PhysRevApplied.12.054051.

[90] John Napp and John Preskill. "Optimal Bacon-Shor codes". In: *Quantum Inf. Comput.* 13 (2013), pp. 490–510.

[91] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2009. DOI: 10.1017/cbo9780511976667. URL: https://doi.org/10.1017/cbo9780511976667.

[92] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 10th. New York, NY, USA: Cambridge University Press, 2011. ISBN: 1107002176, 9781107002173.

[93] D. Nigg et al. "Quantum computations on a topologically encoded qubit". In: *Science* 345.6194 (June 2014), pp. 302–305. DOI: 10.1126/science.1253742. URL: https://doi.org/10.1126/science.1253742.

[94] Kyungjoo Noh and Christopher Chamberland. "Fault-tolerant bosonic quantum error correction with the surface–Gottesman-Kitaev-Preskill code". In: *Phys. Rev. A* 101 (1 Jan. 2020), p. 012316. DOI: 10.1103/PhysRevA.101.012316. URL: https://link.aps.org/doi/10.1103/PhysRevA.101.012316.

[95] Joe O'Gorman and Earl T. Campbell. "Quantum computation with realistic magic-state factories". In: *Physical Review A* 95.3 (Mar. 2017). DOI: 10.1103/physreva.95.032338. URL: https://doi.org/10.1103/physreva.95.032338.

[96] N. Ofek et al. "Extending the lifetime of a quantum bit with error correction in superconducting circuits". In: *Nature* 536 (2016), p. 441.

[97] Takuya Ohno et al. "Phase structure of the random-plaquette gauge model: accuracy threshold for a toric quantum memory". In: *Nuclear Physics B* 697.3 (Oct. 2004), pp. 462–480. DOI: 10.1016/j.nuclphysb.2004.07.003. URL: https://doi.org/10.1016/j.nuclphysb.2004.07.003.

[98] Masayuki Ohzeki. "Locations of multicritical points for spin glasses on regular lattices". In: *Physical Review E* 79.2 (Feb. 2009). DOI: 10.1103/physreve.79.021129. URL: https://doi.org/10.1103/physreve.79.021129.

[99] Adam Paetznick and Ben W. Reichardt. "Universal Fault-Tolerant Quantum Computation with Only Transversal Gates and Error Correction". In: *Physical Review Letters* 111.9 (Aug. 2013). DOI: 10.1103/physrevlett.111.090505. URL: https://doi.org/10.1103/physrevlett.111.090505.

[100] Tefjol Pllaha et al. "Un-Weyl-ing the Clifford Hierarchy". In: *Quantum* 4 (Dec. 2020), p. 370. DOI: 10.22331/q-2020-12-11-370. URL: https://doi.org/10.22331/q-2020-12-11-370.

[101] David Poulin. "Stabilizer Formalism for Operator Quantum Error Correction". In: *Physical Review Letters* 95.23 (Dec. 2005). DOI: 10.1103/physrevlett.95.230504. URL: https://doi.org/10.1103/physrevlett.95.230504.

[102] John Preskill. "Reliable quantum computers". In: *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 454.1969 (Jan. 1998), pp. 385–410. DOI: 10.1098/rspa.1998.0167. URL: https://doi.org/10.1098/rspa.1998.0167.

[103] Shruti Puri et al. "Bias-preserving gates with stabilized cat qubits". In: *Science Advances* 6.34 (Aug. 2020), eaay5901. DOI: 10.1126/sciadv.aay5901. URL: https://doi.org/10.1126/sciadv.aay5901.

[104] S. L. A. de Queiroz. "Location and properties of the multicritical point in the Gaussian and±JIsing spin glasses". In: *Physical Review B* 79.17 (May 2009). DOI: 10.1103/physrevb.79.174408. URL: https://doi.org/10.1103/physrevb.79.174408.

[105] Robert Raussendorf and Jim Harrington. "Fault-Tolerant Quantum Computation with High Threshold in Two Dimensions". In: *Physical Review Letters* 98.19 (May 2007). DOI: 10.1103/physrevlett.98.190504. URL: https://doi.org/10.1103/physrevlett.98.190504.

[106] Matthew Reagor et al. "Quantum memory with millisecond coherence in circuit QED". In: *Phys. Rev. B* 94 (1 July 2016), p. 014506. DOI: 10.1103/PhysRevB.94.014506. URL: https://link.aps.org/doi/10.1103/PhysRevB.94.014506.

[107] Narayanan Rengaswamy, Robert Calderbank, and Henry D. Pfister. "Unifying the Clifford hierarchy via symmetric matrices over rings". In: *Physical Review A* 100.2 (Aug. 2019). DOI: 10.1103/physreva.100.022304. URL: https://doi.org/10.1103/physreva.100.022304.

[108] S. Shankar et al. "Autonomously stabilized entanglement between two superconducting quantum bits". In: *Nature* 504.7480 (Nov. 2013), pp. 419–422. DOI: 10.1038/nature12802. URL: https://doi.org/10.1038/nature12802.

[109] Yaoyun Shi. "Both Toffoli and controlled-NOT Need Little Help to Do Universal Quantum Computing". In: *Quantum Info. Comput.* 3.1 (Jan. 2003), pp. 84–92. ISSN: 1533-7146. URL: http://dl.acm.org/citation.cfm?id=2011508.2011515.

[110] P. W. Shor. "Fault-Tolerant Quantum Computation". In: FOCS '96. USA: IEEE Computer Society, 1996, p. 56.

[111] Peter W. Shor. "Scheme for reducing decoherence in quantum computer memory". In: *Physical Review A* 52.4 (Oct. 1995), R2493–R2496. DOI: 10.1103/physreva.52.r2493. URL: https://doi.org/10.1103/physreva.52.r2493.

[112] M. D. Shulman et al. "Demonstration of Entanglement of Electrostatically Coupled Singlet-Triplet Qubits". In: *Science* 336.6078 (Apr. 2012), pp. 202–205. DOI: 10.1126/science.1217692. URL: https://doi.org/10.1126/science.1217692.

[113] A.M. Steane. "Space, Time, Parallelism and Noise Requirements for Reliable Quantum Computing". In: *Fortschritte der Physik* 46.4-5 (June 1998), pp. 443–457. ISSN: 1521-3978. DOI: 10.1002/(sici)1521-3978(199806)46:4/5<443::aid-prop443>3.0.co;2-8. URL: http://dx.doi.org/10.1002/(SICI)1521-3978(199806)46:4/5%3C443::AID-PROP443%3E3.0.CO;2-8.

[114] L. Sun et al. "Tracking photon jumps with repeated quantum non-demolition parity measurements". In: *Nature* 511.7510 (July 2014), pp. 444–448. DOI: 10.1038/nature13436. URL: https://doi.org/10.1038/nature13436.

[115] Yasunari Suzuki, Keisuke Fujii, and Masato Koashi. "Efficient Simulation of Quantum Error Correction Under Coherent Error Based on the Nonunitary Free-Fermionic Formalism". In: *Phys. Rev. Lett.* 119 (19 Nov. 2017), p. 190503. DOI: 10.1103/PhysRevLett.119.190503. URL: https://link.aps.org/doi/10.1103/PhysRevLett.119.190503.

[116] Ryuji Takagi, Theodore J. Yoder, and Isaac L. Chuang. "Error rates and resource overheads of encoded three-qubit gates". In: *Physical Review A* 96.4 (Oct. 2017). DOI: 10.1103/physreva.96.042302. URL: https://doi.org/10.1103/physreva.96.042302.

[117] B M Terhal, J Conrad, and C Vuillot. "Towards scalable bosonic quantum error correction". In: *Quantum Science and Technology* 5.4 (July 2020), p. 043001. DOI: 10.1088/2058-9565/ab98a5. URL: https://doi.org/10.1088/2058-9565/ab98a5.

[118] S. Touzard et al. "Coherent Oscillations inside a Quantum Manifold Stabilized by Dissipation". In: *Physical Review X* 8.2 (Apr. 2018). DOI: 10.1103/physrevx.8.021005. URL: https://doi.org/10.1103/physrevx.8.021005.

[119] S. Touzard et al. "Gated Conditional Displacement Readout of Superconducting Qubits". In: *Phys. Rev. Lett.* 122 (8 Feb. 2019), p. 080502. DOI: 10.1103/PhysRevLett.122.080502. URL: https://link.aps.org/doi/10.1103/PhysRevLett.122.080502.

[120] David K. Tuckett, Stephen D. Bartlett, and Steven T. Flammia. "Ultrahigh Error Threshold for Surface Codes with Biased Noise". In: *Phys. Rev. Lett.* 120 (5 Jan. 2018), p. 050505. DOI: 10.1103/PhysRevLett.120.050505. URL: https://link.aps.org/doi/10.1103/PhysRevLett.120.050505.

[121] David K. Tuckett et al. "Fault-Tolerant Thresholds for the Surface Code in Excess of 5% under Biased Noise". In: *Phys. Rev. Lett.* 124 (13 Mar. 2020), p. 130501. DOI: 10.1103/PhysRevLett.124.130501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.124.130501.

[122] David K. Tuckett et al. "Tailoring Surface Codes for Highly Biased Noise". In: *Physical Review X* 9.4 (Nov. 2019). DOI: 10.1103/physrevx.9.041031. URL: https://doi.org/10.1103/physrevx.9.041031.

[123] Christophe Vuillot et al. "Quantum error correction with the toric Gottesman-Kitaev-Preskill code". In: *Physical Review A* 99.3 (Mar. 2019). DOI: 10.1103/physreva.99.032344. URL: https://doi.org/10.1103/physreva.99.032344.

[124] Chenyang Wang, Jim Harrington, and John Preskill. "Confinement-Higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory". In: *Annals of Physics* 303.1 (2003), pp. 31–58. ISSN: 0003-4916. DOI: https://doi.org/10.1016/S0003-4916(02)00019-2. URL: http://www.sciencedirect.com/science/article/pii/S0003491602000192.

[125] Chen Wang et al. "A Schrödinger cat living in two boxes". In: *Science* 352.6289 (2016), pp. 1087–1091. DOI: 10.1126/science.aaf2941. eprint: http://science.sciencemag.org/content/352/6289/1087.full.pdf. URL: http://science.sciencemag.org/content/352/6289/1087.

[126] D. S. Wang et al. *Threshold error rates for the toric and surface codes*. 2009. arXiv: 0905.0531 [quant-ph].

[127] David S. Wang, Austin G. Fowler, and Lloyd C. L. Hollenberg. "Surface code quantum computing with error rates over 1%". In: *Physical Review A* 83.2 (Feb. 2011). DOI: 10.1103/physreva.83.020302. URL: https://doi.org/10.1103/physreva.83.020302.

[128] T. F. Watson et al. "A programmable two-qubit quantum processor in silicon". In: *Nature* 555.7698 (Feb. 2018), pp. 633–637. DOI: 10.1038/nature25766. URL: https://doi.org/10.1038/nature25766.

[129] Paul Webster, Stephen D. Bartlett, and David Poulin. "Reducing the overhead for quantum computation when noise is biased". In: *Physical Review A* 92.6 (Dec. 2015). DOI: 10.1103/physreva.92.062309. URL: https://doi.org/10.1103/physreva.92.062309.

[130] Theodore J. Yoder, Ryuji Takagi, and Isaac L. Chuang. "Universal Fault-Tolerant Gates on Concatenated Stabilizer Codes". In: *Phys. Rev. X* 6 (3 Sept. 2016), p. 031039. DOI: 10.1103/PhysRevX.6.031039. URL: https://link.aps.org/doi/10.1103/PhysRevX.6.031039.

RÉSUMÉ

La construction d'un ordinateur quantique est un défi technologique extrêmement difficile à cause de la fragilité des états quantiques qui servent de support de calcul. La réalisation d'une telle machine nécessite de construire un grand nombre de systèmes quantiques suffisamment protégés du bruit inévitable induit par l'environnement, afin que la durée de vie de l'information quantique encodée dans ces systèmes soit suffisamment grande devant le temps d'exécution typique d'un algorithme quantique. Paradoxalement, l'implémentation d'algorithmes suppose aussi la capacité de manipuler l'information. La théorie de la correction d'erreur quantique établit qu'il est possible de construire des systèmes quantiques de taille macroscopique, et pourtant arbitrairement bien protégés contre le bruit induit par l'environnement. En pratique, deux obstacles majeurs s'opposent à la réalisation physique de la correction d'erreur quantique. Le premier obstacle à surmonter est de parvenir à construire un système quantique ``de base'' pour lequel le niveau du bruit est déjà suffisamment bas, un résultat connu sous le nom de ``théorème du seuil''. Le second obstacle concerne la taille de l'ordinateur quantique: en effet, aussi bien le code correcteur d'erreur que la capacité à manipuler l'information logique sont responsables d'un important surcoût en matériel.

L'objet de cette thèse est la construction et l'analyse d'un schéma particulier pour la réalisation d'un ordinateur quantique basé sur les qubits de chat répétés. La protection contre les erreurs est assurée en deux temps. D'abord, les qubits de chats utilisés sont arbitrairement bien protégés contre les erreurs dites de ``bit-flip'', lorsque le nombre moyen de photons dans les états de chat utilisés est suffisamment grand. Ensuite, un code de répétition contre les erreurs dites de ``phase-flip'' est construit à partir de ces qubits de chat. Le qubit logique résultant de cette construction est appelé le qubit de chat répété. Afin de réaliser du calcul quantique avec ce qubit, un ensemble universel de portes logiques pour les qubits de chat répétés est proposé dans cette thèse. La construction de ces portes logiques respecte la structure de la protection, afin de la préserver: les portes physiques construites au niveau des qubits de chat ont la propriété d'être ``bias-preserving'', c'est-à-dire qu'elles préservent la protection naturelle contre les erreurs de ``bit-flip''. Les propositions d'implémentation de ces portes ont été adaptées au maximum à la réalité des expériences contemporaines, afin que le schéma proposé puisse être raisonnablement implémenté d'ici quelques années. Enfin, à partir de ces portes physiques, un ensemble universel de portes logiques est construit au niveau du qubit de chat répété.

Nous espérons que le schéma ainsi proposé, et les idées développées pour sa construction, seront utiles dans la réalisation d'un ordinateur quantique.

MOTS CLÉS

Correction d'erreur quantique, information quantique, calcul quantique universel, qubits supraconducteurs

ABSTRACT

The construction of a quantum computer is an extremely challenging task, because the states of the quantum system used to carry out the computation are typically far too fragile. A necessary condition to build such a computer is to design a system in which a large number of quantum bits are protected from the devastating effect of their environment to withstand the quantum information for a sufficiently long time. At the same time, performing a computation supposes the ability to control the quantum states to process the information they encode. The theory of quantum error correction opens the path towards the realization of macroscopically large quantum systems with, in theory, arbitrary good protection against the noise induced by the environment. The bottleneck of the implementation of quantum error correction is twofold. First, it requires to build quantum systems for which the levels of noise are below a constant value called the accuracy threshold. Second, the quantum error correcting code, and the processing of the encoded information, result in a large physical hardware overhead.

In this thesis, we propose and analyze a scheme based on repetition cat qubits to perform large scale quantum computation. The protection against the environment induced noise is achieved in two steps. First, the two-photon pumped cat qubits are arbitrarily well protected against bit-flip errors with the average number of photons in the cat state. Second, a repetition code protecting against phase-flips is implemented using cat qubits, thus producing a ``repetition cat qubit'' with very low logical error rate. In order to perform quantum computation with this protected qubit, we design a universal set of logical gates acting on the repetition cat qubit, that is compatible with the structure of the scheme. More precisely, the construction is achieved in two steps: first, the design of physical operations acting on cat qubits. These operations must be bias-preserving in order to preserve the natural protection against bit-flip errors. A particular attention has been devoted to proposing operations that could be experimentally realized in the next few years, within the framework of circuit quantum electrodynamics. Then, from the set of bias-preserving physical operations, we construct a universal set of logical operations on the repetition cat qubits.

We hope that the resulting scheme, and the ideas developed for its construction, will prove useful for the construction of a large-scale quantum computer.

KEYWORDS

Quantum error correction, quantum information, universal quantum computation, quantum fault-tolerance, superconducting qubits