THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Bosonic quantum computing with near-term devices and beyond

TIMO HILLMANN

Applied Quantum Physics Laboratory Department of Microtechnology and Nanoscience (MC2) CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2025

Bosonic quantum computing with near-term devices and beyond TIMO HILLMANN ISBN 978-91-8103-174-4

© TIMO HILLMANN, 2025

Doktorsavhandlingar vid Chalmers tekniska högskola Ny serie nr. 5632 ISSN 0346-718X

Applied Quantum Physics Laboratory Department of Microtechnology and Nanoscience (MC2) Chalmers University of Technology SE-412 96 Göteborg Sweden Telephone: +46 (0)31-772 1000

Cover:

This illustration captures the thematic intersection explored in this thesis: quantum error correction, bosonic codes, and superconducting circuits. The left side features an artistic rendition of a Tanner graph, representing the aspects of quantum error correction, while the right side shows a stylized superconducting quantum circuit. These two domains are unified by the central height profile of a Wigner function, evoking the continuous-variable nature at the heart of bosonic quantum error correction. This illustration was generated with the assistance of ChatGPT.

Chalmers Digitaltryck Göteborg, Sweden 2025 Bosonic quantum computing with near-term devices and beyond TIMO HILLMANN Applied Quantum Physics Laboratory Department of Microtechnology and Nanoscience (MC2) Chalmers University of Technology

Abstract

This thesis investigates scalable strategies for fault-tolerant quantum computation by developing and analyzing bosonic quantum codes, quantum low-density parity-check (LDPC) codes, and decoding protocols that aim to unify bosonic and discrete-variable quantum error correction.

In the continuous-variable regime, we explore the use of native nonlinearities in superconducting microwave circuits to realize a universal gate set for continuous-variable quantum computing, including the deterministic generation of a cubic phase state. Separately, we propose and analyze the dissipatively squeezed cat qubit, a noise-biased bosonic encoding that offers improved error suppression and faster gate implementations compared to standard cat qubits. To evaluate the broader viability of bosonic encodings, we study the performance of rotation-symmetric and Gottesman-Kitaev-Preskill (GKP) codes under realistic noise and measurement models, revealing important trade-offs in measurement-based approaches.

Recognizing the need to integrate bosonic codes into larger fault-tolerant architectures, we develop decoding techniques that explicitly leverage analog syndrome information from the readout of bosonic modes. These methods reduce the need for repeated measurements and enable quasi-single-shot decoding in concatenated schemes, forming a bridge between continuous-variable encodings and discrete-variable stabilizer codes.

To advance scalable discrete-variable fault tolerance, we introduce localized statistics decoding, a flexible and highly parallelizable decoding framework for general quantum LDPC codes with state-of-the-art accuracy. Based on a novel on-the-fly matrix elimination strategy, this decoder efficiently identifies and resolves local error configurations, enabling low-latency and hardware-friendly implementations. Additionally, we present quantum radial codes, a new family of single-shot quantum LDPC codes constructed from lifted products of classical quasi-cyclic codes. These codes offer low overhead, tunable parameters, and competitive performance under circuit-level noise, making them promising candidates for near-term implementation.

Finally, we propose the concept of fault complexes, a homological framework for representing and analyzing faults in dynamic quantum error correction protocols. Extending the role of homology in static CSS codes, fault complexes provide a general language for the design and analysis of fault-tolerant schemes.

Keywords: Quantum Error Correction, Bosonic Codes, Superconducting Circuits, Quantum low-density parity-check codes, Squeezed cat qubit, Gottesman-Kitaev-Preskill code, Belief propagation, Ordered statistics decoding, Localized statistics decoding, CSS codes, Quantum Radial Codes, Schrieffer-Wolff transformation,

LIST OF PUBLICATIONS

This thesis is based on the work contained in the following papers:

Paper I	T. Hillmann, F. Quijandría, G. Johansson, A. Ferraro, S. Gas- parinetti, and G. Ferrini, Universal Gate Set for Continuous-Variable Quantum Computation with Microwave Circuits, Phys. Rev. Lett. 125, 160501 (2020), arXiv:2002.01402 [quant-ph]
Paper II	T. Hillmann and F. Quijandría, Designing Kerr Interactions for Quantum Information Processing via Counterrotating Terms of Asymmetric Josephson-Junction Loops, Phys. Rev. Applied 17 , 064018 (2022), arXiv:2107.06757 [quant-ph]
Paper III	A. M. Eriksson, T. Sépulcre, M. Kervinen, T. Hillmann , M. Kudra, S. Dupouy, Y. Lu, M. Khanahmadi, J. Yang, C. Castillo-Moreno, P. Delsing, and S. Gasparinetti, Universal control of a bosonic mode via drive-activated native cubic interactions, Nat Commun 15 , 2512 (2024), arXiv:2308.15320 [quant-ph]
Paper IV	T. Hillmann , F. Quijandría, A. L. Grimsmo, and G. Ferrini, Performance of Teleportation-Based Error-Correction Circuits for Bosonic Codes with Noisy Measurements, PRX Quantum 3 , 020334 (2022), arXiv:2108.01009 [quant-ph]
Paper V	T. Hillmann and F. Quijandría, Quantum error correction with dissipatively stabilized squeezed-cat qubits, Phys. Rev. A 107, 032423 (2023), arXiv:2210.13359 [quant-ph]
Paper VI	L. Berent, T. Hillmann , J. Eisert, R. Wille, and J. Roffe, Analog Information Decoding of Bosonic Quantum Low-Density Parity- Check Codes, PRX Quantum 5, 020349 (2024), arXiv:2311.01328 [quant-ph]
Paper VII	T. Hillmann , L. Berent, A. O. Quintavalle, J. Eisert, R. Wille, and J. Roffe, <i>Localized statistics decoding: A parallel decoding algorithm for quantum low-density parity-check codes</i> , June 2024, arXiv:2406.18655 [quant-ph]
Paper VIII	T. R. Scruby, T. Hillmann , and J. Roffe, <i>High-threshold</i> , <i>low-overhead and single-shot decodable fault-tolerant quantum memory</i> , June 2024, arXiv:2406.14445 [quant-ph]
Paper IX	T. Hillmann , G. Dauphinais, I. Tzitrin, and M. Vasmer, <i>Singleshot and measurement-based quantum error correction via fault complexes</i> , Oct. 2024, arXiv:2410.12963 [quant-ph]

Other work outside the scope of this thesis:

Paper A	Y. Lu, M. Kudra, T. Hillmann , J. Yang, HX. Li, F. Quijandría, and P. Delsing, Resolving Fock states near the Kerr-free point of a superconducting resonator, npj Quantum Inf 9 , 1–11 (2023), arXiv:2210.09718 [quant-ph]
Paper B	R. N. Alexander, J. E. Bourassa, J. Hastrup, and T. Hillmann , Generation of Gottesman-Kitaev-Preskill (GKP) States Based on Multi-Peak Quantum States of Light, US20240346357A1 (Oct. 2024)
Paper C	B. W. Walshe, B. Q. Baragiola, H. Ferretti, J. Gefaell, M. Vasmer, R. Weil, T. Matsuura, T. Jaeken, G. Pantaleoni, Z. Han, T. Hillmann , N. C. Menicucci, I. Tzitrin, and R. N. Alexander, Linear-Optical Quantum Computation with Arbitrary Error-Correcting Codes, Phys. Rev. Lett. 134 , 100602 (2025), arXiv:2408.04126 [quant-ph]
Paper D	 H. Aghaee Rad, T. Ainsworth, R. N. Alexander, B. Altieri, M. F. Askarani, R. Baby, L. Banchi, B. Q. Baragiola, J. E. Bourassa, R. S. Chadwick, I. Charania, H. Chen, M. J. Collins, P. Contu, N. D'Arcy, G. Dauphinais, R. De Prins, D. Deschenes, I. Di Luch, S. Duque, P. Edke, S. E. Fayer, S. Ferracin, H. Ferretti, J. Gefaell, S. Glancy, C. González-Arciniegas, T. Grainge, Z. Han, J. Hastrup, L. G. Helt, T. Hillmann, J. Hundal, S. Izumi, T. Jaeken, M. Jonas, S. Kocsis, I. Krasnokutska, M. V. Larsen, P. Laskowski, F. Laudenbach, J. Lavoie, M. Li, E. Lomonte, C. E. Lopetegui, B. Luey, A. P. Lund, C. Ma, L. S. Madsen, D. H. Mahler, L. Mantilla Calderón, M. Menotti, F. M. Miatto, B. Morrison, P. J. Nadkarni, T. Nakamura, L. Neuhaus, Z. Niu, R. Noro, K. Papirov, A. Pesah, D. S. Phillips, W. N. Plick, T. Rogalsky, F. Rortais, J. Sabines-Chesterking, S. Safavi-Bayat, E. Sazhaev, M. Seymour, K. Rezaei Shad, M. Silverman, S. A. Srinivasan, M. Stephan, Q. Y. Tang, J. F. Tasker, Y. S. Teo, R. B. Then, J. E. Tremblay, I. Tzitrin, V. D. Vaidya, M. Vasmer, Z. Vernon, L. F. S. S. M. Villalobos, B. W. Walshe, R. Weil, X. Xin, X. Yan, Y. Yao, M. Zamani Abnili, and Y. Zhang, Scaling and networking a modular photonic quantum computer, Nature 638, 912–919 (2025)

Contents

Al	bstract	i	
\mathbf{Li}	st of publications	iii	
Co	ontents	\mathbf{v}	
A	cknowledgements	ix	
1	Introduction1.1Towards Fault-Tolerant Quantum Computing1.2Outline	$ \begin{array}{c} 1 \\ 2 \\ 3 \end{array} $	
2	Fault-tolerant Quantum Computing 2.1 Fundamentals of Quantum Computing 2.2 Measurement-based Quantum Computing 2.3 Classical Error Correction 2.4 Quantum Error Correction 2.5 Chain Complexes 2.6 Decoding of Quantum Error-Correcting Codes 2.7 Conclusion and Outlook	5 7 10 14 18 24 28 39	
3	Quantum Continuous Variables3.1Fundamentals of Quantum Continuous Variables3.2The Wigner function3.3Quantum Harmonic Oscillators3.4Noise Channels3.5Universal Gates3.6Bosonic Quantum Error Correction3.7Conclusion and Outlook	41 42 44 46 49 51 58	
4	Quantum Computing Architectures 4.1 Superconducting Circuit Architectures 4.2 Engineering Effective Interactions 4.3 Extracting Effective Models 4.4 Conclusion and Outlook	61 68 71 72	
5	Conclusion and Perspective	73	
Re	References		
A	ppended papers	91	

Being a physicist is a great privilege. Be worthy of it. Most of humanity spends its life doing boring repetitive tasks.

David P. Stern, All I Really Need to Know..., Physics Today 46 (5), 63 (1993)

ACKNOWLEDGEMENTS

First and foremost, I would like to thank Fernando Quijandría for engaging with my ideas — however vague they may have been — throughout the past five years. Though you moved to Japan before formally becoming my thesis supervisor, your encouragement and guidance at key moments gave me the confidence to pursue this work independently. Thank you for your support, even from afar. I would also like to thank Giulia Ferrini for taking on the role of supervisor and for providing the framework that allowed this thesis to come together.

The time throughout my PhD has been shaped by various opportunities and people. At the beginning of it stands the fellowship from the Excellence Initiative Nano that funded this research, granting me the freedom and independence that I am now so used to. Furthermore, an important aspect of my PhD experience has been working with the team at Xanadu Quantum Computing. I would like to especially thank Rafael Alexander, Ben Baragiola, Eli Bourassa, Ilan Tzitrin, and Mike Vasmer for acting as great collaborators, teachers, friends, and mentors throughout this time. Possibly more than anything else, my participation in the IBM Quantum Error Correction Summer School 2022 had a profound influence on the shape this thesis ultimately took. I am grateful to the organizers and lecturers for creating such a formative experience.

I am also indebted to the organizers of the many conferences, group visits, and workshops I had the privilege to attend. In particular, QIP Ghent 2023, YITP QEC 2024, FTQT Benasque 2024, and Coogee 2025 stand out as events that not only contributed to my research but also led to lasting memories, friendships, and future collaborations. What made attending these events even more enriching was knowing that, no matter where I went, I could always find someone I knew — whether to discuss new research ideas or to enjoy a coffee together. Thank you, Asmae Benhenou, Lucas Berent, Matt McEwen, Julio Magdalena de Fuentes, Teague, Tom Scruby, and many more.

I was also glad to have made many friends at AQP, QTL, and around Gothenburg generally. Thank you, Alberto Del Ángel Medina, Oliver Hahn, Théo Sépulcre, David Fitzek, Therese Karmstrand, Henning Kirchberg, Claudia Castillo-Moreno, Amr Osman, and Chris Warren, Werner Dobrautz and others for the enjoyable coffee and fika breaks, lunch conversations, and climbing sessions. Thanks to Vitaly Shumeiko for being someone with whom I could always have meaningful physics discussions. To my first Master's student, Thomas Jaeken, thank you for meeting the challenge with trust, energy, and curiosity.

Many thanks to A43 Coffee and Beanaut Coffee Roasters, and the specialty coffee community of Sweden, for introducing me to the world of specialty coffee. I've made sure to carry that passion into the quantum error correction community.

To all my other friends, thank you for the visits, messages, and distractions — for keeping me connected to a life beyond this thesis. Finally, I want to thank my family for their continuous support throughout my 10 years of university, despite the distance that kept us apart for much of that time.

Timo Hillmann, Göteborg, May 2025

CHAPTER 1

Introduction

Classical computers have become indispensable tools for solving a wide range of problems. However, some problems are out of reach for these devices. Quantum computing offers an alternative paradigm of computing that utilizes the features of quantum mechanics, which promises to solve some of those problems more efficiently than classical computers. Most prominently, quantum computers promise an exponential speed up for integer factorization [1, 2] and the simulation of quantum systems [3]. The latter is particularly challenging for conventional computers because they require incorporating the laws of quantum mechanics into the simulation — a task that another quantum system, a quantum computer, is potentially much better suited for [4, 5]. Being able to more efficiently and accurately simulate quantum systems could have a significant impact on material engineering [6], aiding the design of improved batteries and photovoltaics, quantum chemistry [7–11], and the development of pharmaceuticals [12]. However, realizing these advantages requires overcoming profound physical and theoretical challenges, foremost among them the fragility of quantum information. This fragility necessitates the development of fault-tolerant quantum computers, where encoded quantum information can be protected and manipulated reliably over time.

It has become evident that quantum error correction will be necessary for almost any practical application of quantum computers, with one of the few exceptions possibly being small-scale physics simulations. In general, the need for quantum error correction introduces a substantial overhead [13, 14] that reduces the advantage of quantum algorithms over conventional classical algorithms. Quantum algorithms are further challenged by the classical algorithm design based on heuristics, which are efficient methods guided by intuition and insight that lack formal guarantees yet typically yield accurate results. Indeed, in classical computing, many algorithms began as heuristics, with their strengths understood only after broad application, see e.g., Ref. [15]. Approaching the development of quantum algorithms in a similar spirit is possibly hindered in two ways. First, quantum mechanics often defies direct intuition, and as a result, important discoveries, such as gate teleportation [16], were made numerically by accident [17]. Second, the absence of large-scale fault-tolerant quantum hardware limits the discovery of new protocols to those tractable by classical simulation.

The choice of the quantum error-correcting code carries many practical trade-offs. The current gold standard, the surface code [18–20], requires only nearest-neighbor connectivity in a planar architecture and has a high resilience against realistic noise, but will require thousands of physical qubits for each logical qubit it protects against noise. On the other hand, quantum low-density parity-check (qLDPC) codes [21] can reduce the number of physical qubits needed but have increased connectivity requirements and are typically less resilient against realistic noise processes compared to the surface code. Theoretical progress in this domain is rapid, not only on code construction and design [22–27], but also decoding [28–33], and logical operations [34–40].

Interestingly, we are approaching a tipping point in the design of quantum errorcorrecting codes where theoretical progress is beginning to translate into small-scale experimental implementations, for example, in superconducting circuits [41–47], neutral atoms [48–50], or trapped ions [51–56]. Importantly, this allows us not only to refine our understanding of fault tolerance in practical settings but also to design more efficient quantum error-correcting codes tailored to the physical architecture.

One of the subfields within quantum computing that has made exceptional experimental and theoretical progress in the last decade is the field of bosonic quantum computing and bosonic quantum error correction [57–59]. Bosonic quantum computing possibly resembles the quantum analog of the almost forgotten field of classical analog computing more closely than the notion of discrete-variable (quantum) information processing discussed so far. It is therefore also often referred to as continuous-variable quantum information processing or quantum computing. On the other hand, bosonic quantum error correction, or simply bosonic codes, represent discrete quantum information encoded into a system of quantum continuous variables. Some of the most prominent bosonic codes, such as the Gottesman-Kitaev-Preskill code [60] or the cat qubit [61, 62], can be viewed as utilizing ideas present in classical transistors to protect quantum information. For instance, in flash memory, a logical bit is defined by the charge on a floating gate — for practical purposes, a continuous quantity — and a threshold is used to distinguish between zero and one. The physical encoding is analog, but the (logical) computation is digital, similar to bosonic codes. The premise of bosonic codes is to exploit hardware-level protection to achieve fault tolerance with fewer physical resources than their (unprotected) discrete-variable counterparts. However, their improved protection comes with other trade-offs such as increased physical complexity, susceptibility to other noise channels, and more challenging gate implementations. Nonetheless, much progress has been made theoretically [63–79] as well as experimentally [42, 43, 80-89].

1.1 Towards Fault-Tolerant Quantum Computing

The pursuit of practical quantum computing is inextricably linked to the development of robust fault-tolerant architectures. While small quantum processors have demonstrated increasingly sophisticated capabilities [90–92], the realization of large-scale, useful quantum computations continues to be limited by the fragility of quantum information and the

substantial resource overheads required for its protection.

This thesis explores bosonic codes and qLDPC codes as candidates for building a faulttolerant quantum computer. In particular, at the core, this thesis investigates how bosonic codes and qLDPC codes can contribute to reducing the hardware and time-overhead of quantum error correction. Instead of focusing on these approaches individually, it aims to address these questions in a unified approach. On a broader scale, it contributes to bridging the gap between theoretical abstraction and practical implementations in fault-tolerant quantum computing. To this end, contributions range from physical-level encoding strategies and decoding protocols to the introduction of a framework for the representation of quantum error-correcting codes in space-time.

The following section provides a detailed, chapter-by-chapter outline of the thesis structure and the specific contributions of each paper.

1.2 Outline

Chapter 2 introduces foundational ideas in fault-tolerant quantum computing that frame the contributions of this thesis within the broader context of quantum error correction. It begins by outlining models of quantum computation, including both gate-based and measurement-based paradigms, and proceeds to cover stabilizer codes as a key class of quantum error-correcting codes. The chapter further introduces the language of chain complexes to describe code structure and delves into various approaches to the decoding problem. These topics form the theoretical foundation for the contributions presented in **Paper VII**, **Paper VIII**, and **Paper IX**, and quantum computing more generally.

Chapter 3 focuses on the formalism of quantum continuous-variable systems and their role in quantum information processing. It introduces relevant physical and mathematical concepts for describing infinite-dimensional quantum systems, followed by an overview of bosonic quantum error-correcting codes designed to protect information encoded in such systems. Particular emphasis is placed on cat codes and the Gottesman-Kitaev-Preskill encoding, which play a central role in the work presented in **Paper IV**, **Paper V**, and **Paper VI**. The chapter serves as a conceptual bridge between theoretical developments and their application in hardware-efficient fault-tolerant schemes.

Chapter 4 explores how the continuous-variable systems introduced in the previous chapter can be implemented using superconducting quantum circuits. It focuses on circuit quantum electrodynamics as a versatile platform for building quantum hardware, emphasizing techniques for engineering nonlinear interactions that are essential for quantum control. These capabilities enable the realization of bosonic codes and operations, connecting directly to the contributions of **Paper I**, **Paper II**, and **Paper III**.

CHAPTER 2

Fault-tolerant Quantum Computing

Quantum computers process information using quantum bits, *qubits*, instead of conventional (classical) bits. A classical bit stores a single unit of binary information, typically represented as 0 or 1. Similarly, a qubit can be represented by two distinguishable quantum states, $|0\rangle$ and $|1\rangle$. However, unlike classical bits, a qubit can exist in a superposition of these states. The general quantum state $|\psi\rangle$ of a single qubit is given by

$$|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle, \qquad (2.1)$$

where c_0 and c_1 are complex coefficients satisfying the normalization condition $|c_0|^2 + |c_1|^2 = 1$. We can also use a qubit to represent a classical bit by requiring that at most one of the coefficients c_i is nonzero. However, even though the coefficients c_i can take on infinitely many different values and thus allow us to store, in principle, an infinite amount of information, we can at most retrieve a single bit of information. When measured, a qubit collapses to the classical state 0 with probability $|c_0|^2$ and 1 with probability $|c_1|^2$. This measurement process erases any phase information and prevents direct access to the values of c_0 and c_1 from a single measurement. Instead, only their magnitudes can be inferred statistically by performing multiple measurements on identical copies of the same quantum state.

The difference in computational power of a qubit over a bit is hence much more subtle and is partially due to the possibility that the complex-valued coefficients can interfere, something that is not possible for classical bits. An operation that exemplifies quantum interference is the Hadamard gate H. The Hadamard operation transforms qubit states as follows

$$H|0\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle = |+\rangle, \qquad (2.2)$$

$$H|1\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle = |-\rangle.$$
 (2.3)

If we begin initially in the equal superposition state $|\psi\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$ and then apply a Hadamard operation to it, we obtain

$$H \left|\psi\right\rangle = \left(\frac{1}{2} + \frac{1}{2}\right)\left|0\right\rangle + \left(\frac{1}{2} - \frac{1}{2}\right)\left|1\right\rangle = \left|0\right\rangle, \qquad (2.4)$$

due to the cancellation of the $|1\rangle$ term, an example of destructive interference.

Another key ingredient that distinguishes quantum computation from classical computation is *entanglement*, a form of quantum correlation that arises in multi-qubit superposition states without a classical counterpart. To illustrate this, let us consider the so-called Bell state $|\Phi^+\rangle$ defined as

$$\left|\Phi^{+}\right\rangle = \frac{1}{\sqrt{2}}(\left|0\right\rangle \left|0\right\rangle + \left|1\right\rangle \left|1\right\rangle). \tag{2.5}$$

Measuring the first qubit, the outcome is 0 or 1 with equal probability. However, the result of the measurement determines the state of the second qubit, and the measurement outcome of it will be perfectly correlated with the measurement result of the first qubit. On the level of the measurement statistics, one obtains the classical bit string 00 or 11 with equal probability. Yet, these correlations alone do not fully capture the essence of entanglement. A classical system could mimic the same statistics by randomly selecting 0 or 1 and preparing either $|0\rangle |0\rangle$ or $|1\rangle |1\rangle$ accordingly. What distinguishes entanglement from mere correlation is its robustness against local operations such as the Hadamard operation. To see this, consider applying the Hadamard operation to both qubits before measuring them. A quick calculation shows that the Bell state $|\Phi^+\rangle$ remains unchanged under the application of the Hadamard operations as the anti-correlation terms $|0\rangle |1\rangle$ and $|1\rangle |0\rangle$ are canceled by interference effects. This cancellation, however, is not possible if the Hadarmard operation is applied to one of the non-superposition states $|0\rangle |0\rangle$ or $|1\rangle |1\rangle$ such that the measurement outcome of the first and second qubit become perfectly uncorrelated [93].

The quantum mechanical concepts of superposition and entanglement thus can be viewed as the origin of the separation of computational power of qubits and classical bits, as they are impossible to reproduce efficiently with classical bits alone. Unfortunately, harnessing this separation in computational power into useful quantum algorithms appears to be a highly nontrivial task. This might come as a surprise, especially considering the common misconception that superposition enables quantum computers to perform an exponentially large number of operations in parallel. In fact, while superposition does allow a quantum system to represent all possible solutions to a problem simultaneously, valid and invalid ones, this does not mean that a quantum computer can simply evaluate all solutions at once to output the desired answer. This oversimplification completely ignores the role of quantum interference in quantum information processing. Instead, designing quantum algorithms should be viewed as the intricate task of steering constructive and destructive interference by quantum operations in such a way that the measurement of the quantum state at the end of the algorithm deterministically yields the correct solution. A weaker statement of the above could be that the role of the algorithm is to amplify the probability of obtaining the correct solution while suppressing the likelihood of wrong solutions.

In the following, we provide a brief overview of quantum operations.

2.1 Fundamentals of Quantum Computing

Running a quantum algorithm, and thus performing quantum information processing, requires a set of operations executed in a particular order that will determine the output of the algorithm. This set of operations must be general enough to be able to reach any state with arbitrary precision and is called *universal* if that is the case. Surprisingly, even though the coefficients c_i of a quantum state can take arbitrary complex values, a finite set of operations suffices. This celebrated result is known as the Solovay-Kitaev theorem [94, 95] that states that given a universal set of operations \mathcal{G} , there is a finite sequence of operations S of length $O(\log^c(1/\epsilon))$ that can be found efficiently such that Sapproximates an arbitrary unitary $U \in SU(d)$ with accuracy $\epsilon > 0$ in the operator norm with constant $c \approx 3.97$. There are several universal gate sets, and we will focus on a particular one commonly referred to as "Clifford + T". This set consists of the controlled-NOT (CNOT or CX) gate, the phase gate S, the previously discussed Hadamard gate H, and the $\pi/8$ -phase gate T. We will define the notion of Clifford gates more precisely below.

2.1.1 Pauli group and Heisenberg representation

Consider a quantum computer, that is, a quantum system, in the state $N |\psi\rangle$, to which a unitary operation U is applied, where N is another operation we have previously applied. Then the state of the system is described by

$$UN |\psi\rangle = UNU^{\dagger}U |\psi\rangle, \qquad (2.6)$$

that is, the operator UNU^{\dagger} now acts on the (transformed) system as N did before applying U. As a result, instead of considering how the state vector $|\psi\rangle$ evolves through the applications of arbitrary operations U, one might equivalently track the evolution

$$N \to U N U^{\dagger},$$

for a sufficiently large set of operators $\{N\}$. Since the evolution of quantum systems is linear, it is sufficient to follow a set of operators that spans the space of linear operators for the Hilbert space \mathcal{H} in which $|\psi\rangle$ would evolve, i.e., $|\psi\rangle \in \mathcal{H}$.

To this end, let us introduce the set of Hermitian and unitary operations that can be represented as 2×2 matrices:

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad (2.7)$$

where X, Y, and Z are collectively known as the Pauli matrices. Importantly, the matrices in Eq. (2.7) form a basis for the space $\mathbb{C}^{2\times 2}$, and thus can be used to describe any operation on a single qubit. Let us also define an *n*-qubit Pauli operator $P = \alpha P_n$, where P_n is the *n*-fold tensor product $P_n \in \{I, X, Y, Z\}^{\otimes n}$, and $\alpha \in \{\pm 1, \pm i\}$ is a coefficient. We will often refer to the *n*-qubit Pauli operator P_n as a Pauli string and write it by omitting the tensor product, e.g., $X \otimes Z \otimes Z \otimes X$ is represented as XZZX. The set of all *n*-qubit Pauli operators forms the *n*-qubit Pauli group \mathcal{P}_n , and, as in the single-qubit case, the elements of \mathcal{P}_n form a basis for $\mathbb{C}^{2^n \times 2^n}$. A convenient basis is $\{X_1, \ldots, X_n, Z_1, \ldots, Z_n\}$ where X_i (Z_i) denote the Pauli X (Z) acting on the *i*th qubit and trivially elsewhere. Therefore, to completely specify a general operator on $\mathbb{C}^{2^n \times 2^n}$ we only need to track the evolution of 2n single-qubit operators. Since the single-qubit Pauli matrices anti-commute with each other, two *n*-qubit Pauli operators anti-commute only if an odd number of their tensor factors anti-commute. We will refer to the set of operators $\{X_1, \ldots, X_n, Z_1, \ldots, Z_n\}$ as *logical* operators, a term that will be justified in a later section. We denote by \overline{X}_i and \overline{Z}_i the operators obtained from transformations of X_i and Z_i , respectively.

Within this framework, unitary operators U that map Pauli operators to Pauli operators are elements of the Clifford group, that is, U is Clifford if $UPU^{\dagger} \in \mathcal{P}_n$, $\forall P \in \mathcal{P}_n$. The CX conjugates the Pauli X and Z operators as follows

$$CX: XI \to XX, IX \to IX, ZI \to ZI, IZ \to ZZ.$$
 (2.8)

Furthermore, the phase gate S maps $X \to Y$ and $Z \to Z$ under conjugation, while the H gate maps $X \to Z$ and $Z \to X$ under conjugation. As expected, the T gate is not a Clifford gate. It is commonly represented in the computational basis as

$$T = \sqrt{S} = \begin{bmatrix} 1 & 0\\ 0 & e^{i\pi/4} \end{bmatrix},\tag{2.9}$$

and, for example, conjugates the Pauli X operator as $TXT^{\dagger} = \cos(\pi/4)X + \sin(\pi/4)Y \notin \mathcal{P}_1$. In the following, we will also make use of the controlled-Z (CZ) gate that acts similarly to the CX gate up to a local Hadamard gate, that is, the CZ gate maps

$$XI \to XZ, \ IX \to ZX, \ ZI \to ZI, \ IZ \to IZ,$$
 (2.10)

under conjugation.

2.1.2 Stabilizer circuits and measurements

For some quantum circuits, we do not need to follow the evolution of all 2n Pauli operators. This is the case if some of the circuit inputs are fixed to be +1 eigenstates of elements of the Pauli group. The set of such operators constrains the state of some qubits and is closed under multiplication. This set of operators defines the stabilizer group S as

$$\mathcal{S} = \{ S \in \mathcal{P}_n \mid S \mid \psi \rangle = \mid \psi \rangle \}, \qquad (2.11)$$

which must hold for all valid input states $|\psi\rangle$. We will return in more detail to the stabilizer group when discussing quantum error-correcting codes in Section 2.4. For now, we will use it to outline an approach that incorporates measurements into the Heisenberg representation of quantum computing.

To this end, let us recall how the measurement of an observable A with eigenvalues ± 1 changes the quantum state $|\psi\rangle$ being measured. The measurement is described by two orthogonal projectors

$$P_{\pm} = \frac{1}{2} \left(I \pm A \right), \tag{2.12}$$



Figure 2.1: **a** Quantum circuit for the measurement of an observable A with eigenvalues ± 1 by an ancilla. The circuit performs a controlled-A gate between the n qubit system and an ancilla initially prepared in the state $|+\rangle$ and measures the ancilla in the X basis. **b** Circuit for measuring A = ZZ using CZ gates.

where I is the identity operator on the space of the n-qubit state $|\psi\rangle$. One can verify that P_{\pm} are projectors, satisfying $P_{\pm}^2 = P_{\pm}$, and that they are orthogonal, meaning $P_{\pm}P_{\mp} = 0$. The post-measurement state $|\psi^{\pm}\rangle = \frac{1}{2}(I \pm A) |\psi\rangle$ is projected into an eigenstate of A with eigenvalue ± 1 , depending on the measurement outcome. Such an observable A can be measured using an ancilla system. To achieve this, one applies a controlled-A gate between the n-qubit system and a single ancilla qubit initially prepared in the $|+\rangle$ state. Then, the ancilla is measured in the X basis; see also Figure 2.1a. One can verify that this circuit effectively measures the observable A through the measurement of the ancilla. To see this, note that the ancilla state is initialized in $|+\rangle$, such that the stabilizer group is generated by a single element: $S = \langle S_1 = X_3 \rangle$, where $\langle \cdot \rangle$ denotes the set of operators that generate S. After applying the controlled-A operation, this stabilizer is updated according to the Heisenberg picture: $X_3 \mapsto X_3A$. Since S_1 commutes with the X-basis measurement of the ancilla, the measurement outcome corresponds to the eigenvalue of A with respect to $|\psi\rangle$.

Before continuing, let us consider a concrete example: the measurement of A = ZZ; see also Figure 2.1b. For a general two-qubit state $|\psi\rangle = \sum_{i,j \in \{0,1\}} c_{ij} |ij\rangle$, the post-measurement state is

$$\left|\psi^{\pm}\right\rangle = \frac{1}{2}(I \pm ZZ) \left|\psi\right\rangle = \begin{cases} \frac{1}{2} \left[c_{00} \left|00\right\rangle + c_{11} \left|11\right\rangle\right], & m = +1, \\ \frac{1}{2} \left[c_{01} \left|01\right\rangle + c_{10} \left|10\right\rangle\right], & m = -1, \end{cases}$$
(2.13)

where $m = \pm 1$ denotes the measurement outcome of the ancilla qubit. In this example, the measurement projects the state into one of two subspaces: the even-parity subspace $(\langle ZZ \rangle = 1)$ or the odd-parity subspace $(\langle ZZ \rangle = -1)$. This observation is noteworthy, as it is a crucial component in quantum error correction with stabilizer codes, which will be described in Section 2.4.2 — it underpins the discretization of errors.

The example above does not capture the full picture — more generally, a measurement may not commute with all elements of the stabilizer group. We will see an example of this below. To update operators in the Heisenberg picture after measuring an observable $M \in \mathcal{P}_n$, apply the following rules [96]. First, identify an element $N \in \mathcal{S}$ that anticommutes with M. Remove N from the stabilizer group and replace it with M in the updated stabilizer group \mathcal{S}' . Then, for each remaining $S \in \mathcal{S} \setminus \{N\}$, add NS to \mathcal{S}' if S anti-commutes with M, and add S unchanged otherwise. Apply the same procedure to the logical operators \overline{X}_j and \overline{Z}_j being evolved in the Heisenberg representation. This update ensures that the post-measurement state lies in the +1 (or -1) eigenspace of M, consistent with the observed measurement outcome.

Importantly, as long as a quantum circuit consists only of gates from the Clifford group, together with initialization and measurement in the Pauli basis, the circuit can be efficiently simulated on a classical computer. This result is known as the Gottesman-Knill theorem [96]. It is the foundational reason why we can simulate the behavior of large quantum error-correcting codes on classical hardware — enabling the efficient benchmarking of codes such as those studied in **Paper VI**, **Paper VII**, **Paper VIII**, and **Paper IX**.

2.2 Measurement-based Quantum Computing

As an alternative to the gate-based or circuit-based model of quantum computation [97] described above, the measurement-based approach of quantum computing was proposed in the early 2000s [98–101]. While different variants of this model of quantum computation exist, we will restrict the discussion here to the graph state model and note that all variants can be unified [102]. In the graph state or cluster state model, one begins with a sufficiently large resource state and performs adaptive single-qubit measurements. The adaptivity in the measurement bases is based upon the fact that earlier measurement outcomes will influence the measurement bases at later stages in the computation. In comparison to gate-based computation, measurement-based quantum computing (MBQC) is often thought of as trading space for time, as it only requires shallow entangling circuits but resource states consisting of a large number of qubits. However, this is only partially correct, as there is no requirement for the full cluster state to exist before the computation begins, and it is sufficient for the cluster state to be extended while the computation proceeds. Furthermore, we note that one might argue that if one is concerned with fault-tolerant quantum computing, modern computation schemes employ the measurement-based approach of quantum computing in practice [103].

2.2.1 Cluster States and Measurements

We will begin with the general definition of a graph state. To this end, recall that a graph G = (V, E) consists of a set of vertices V and edges E between pairs of vertices. If one refers to a graph state, one typically considers that each vertex of G represents a qubit initialized in the state $|+\rangle$ and each edge $(v_1, v_2) \in E$ corresponds to a $CZ(v_1, v_2)$ gate applied between qubits v_1 and v_2 . The graph state is then described by the state vector $|G\rangle$ given by

$$|G\rangle = \prod_{(v_1, v_2) \in E} \operatorname{CZ}(v_1, v_2) |+\rangle^{\otimes |V|},$$
 (2.14)

where |V| is the number of vertices in the graph. Equivalently, we can describe the graph state in terms of its stabilizers, that is, for every qubit $v \in V$, there is a stabilizer

$$S_v = X_v \prod_{u \in N(v)} Z_u, \tag{2.15}$$

where N(v) is the set of neighboring qubits of v. It is easy to show that all stabilizers commute and that they are indeed stabilizers of the state $|G\rangle$.

The Heisenberg representation is convenient for describing how measurements in the Pauli basis affect the graph state. Let us illustrate this based on a small example, the five-qubit star graph given by the graph and stabilizers

$$G = \underbrace{\begin{array}{c}1\\5\\2\end{array}}^{(1)}, \qquad \mathcal{S} = \langle X_1 Z_5, X_2 Z_5, X_3 Z_5, X_4 Z_5, Z_1 Z_2 Z_3 Z_4 X_5 \rangle. \tag{2.16}$$

Measuring qubit 5 with outcome $m_P^{(5)} \in \{\pm 1\}$ with $P \in \{X, Y, Z\}$ produces the stabilizer groups

$$m_X^{(5)} : \mathcal{S} = \langle m_X^{(5)} X_5, X_1 X_2, X_1 X_3, X_1 X_4, m_X^{(5)} Z_1 Z_2 Z_3 Z_4 X_5 \rangle, \tag{2.17}$$

$$m_Y^{(5)}: \mathcal{S} = \langle m_Y^{(5)} Y_5, X_1 X_2, X_1 X_3, X_1 X_4, m_Y^{(5)} Z_1 Z_2 Z_3 Z_4 Y_5 \rangle,$$
(2.18)

$$m_Z^{(5)}: \mathcal{S} = \langle m_Z^{(5)} Z_5, m_Z^{(5)} X_1 Z_5, m_Z^{(5)} X_2 Z_5, m_Z^{(5)} X_3 Z_5, m_Z^{(5)} X_4 Z_5, m_Z^{(5)} Z_1 Z_2 Z_3 Z_4 Z_5 \rangle.$$
(2.19)

Thus, ignoring the measurement outcome, a Z measurement effectively "deletes" the measured qubit. This is not surprising, as after all, the measurement commutes with the entangling circuit of the graph state. An X measurement creates a stabilizer group that can be separated into generators consisting solely of Pauli X and Z operators, respectively. This will be an important detail when we discuss fault-tolerant graph states in the later sections of this thesis.

2.2.2 The teleportation primitive

In the following, we discuss briefly an important underlying idea of MBQC, that is, the one-bit teleportation circuit, see also Figure 2.2a. This circuit consists of two input qubit states, the first one prepared in an arbitrary state $|\psi\rangle$ and the second one prepared in $|+\rangle$. The states are entangled through the application of the CZ gate. Finally, upon measurement of the first qubit in the Pauli X basis, the state of the second qubit conditioned on the measurement outcome m_1 is $X^{m_1}H |\psi\rangle$. Thus, up to local Clifford operations, the state of the first qubit has been teleported onto the second qubit. Note that a measurement in the Z basis would disentangle the qubits, and the second qubit would always be in the state $|+\rangle$ independent of the measurement outcome. The circuit can be slightly generalized by considering the following modification.

Instead of preparing the second qubit in the state $|+\rangle$, prepare the state $P(\varphi) |+\rangle = (|0\rangle + e^{i\varphi} |1\rangle)/\sqrt{2}$, with $P(\varphi)$ the arbitrary phase gate. Then, the output of the one-bit



Figure 2.2: The one-bit teleportation primitive. The left-hand side of the equality shows how to teleport an arbitrary phase gate $P(\varphi)$ onto the input state $|\psi\rangle$ by preparing the second qubit in the state $P(\varphi) |+\rangle$. The right-hand side shows an equivalent circuit that only requires preparation of the state $|+\rangle$ and instead requires measurement of the first qubit in the rotated basis $P(\varphi)XP^{\dagger}(\varphi)$.

teleportation circuit is modified, and one obtains $P(\varphi)X^{m_1}H|\psi\rangle$ conditioned on the measurement outcome. Importantly, this means that even if we are unable to perform the phase gate $P(\varphi)$, as long as we are able to prepare the state $(|0\rangle + e^{i\varphi}|1\rangle)/\sqrt{2}$, we can apply the gate to state $|\psi\rangle$ by performing the teleportation circuit and teleporting the gate $P(\varphi)$ onto it. This is also known as gate-teleportation, see Refs [16, 100] for generalizations. Note that for $\varphi = \pi/4$ one realizes the non-Clifford T gate, see Eq. (2.9).

Lastly, let us mention that if one cannot prepare the initial state $P(\varphi) |+\rangle$, it is possible to derive an equivalent circuit that prepares the second qubit in the state $|+\rangle$ and measures the first qubit in the rotated basis $X_{\varphi} = P(\varphi)XP^{\dagger}(\varphi)$, see also Figure 2.2b. Thus, one can defer the choice of which gate should be applied until a basis for the destructive measurement is chosen and the measurement is performed. By concatenating multiple one-bit teleportation circuits, one can achieve arbitrary single-qubit unitary operations even if one is restricted to a finite set of measurement angles φ , see also Figure 2.3 for a simplified example.



Figure 2.3: Illustration of measurement-based quantum computing. A five-qubit cluster state is shown, with qubits (nodes) entangled as indicated by edges. Measurement proceeds left to right, with measured qubits shown in blue. **a** The left-most qubit is initialized in the state $|\psi\rangle$; others are in $|+\rangle$. **b** Measuring the first qubit in the X basis teleports $|\psi\rangle$ one node right, up to a Hadamard gate and a Pauli correction X^{m_1} . **c** Measuring the next qubit in the X basis further teleports the state, resulting in $X^{m_2}Z^{m_1} |\psi\rangle$ on the third qubit.

2.2.3 Measurement-induced entanglement

A key building block for understanding MBQC is what we refer to as the *measurement-induced entanglement primitive*. This protocol enables the entanglement of two informationcarrying qubits via intermediate measurements, without requiring direct interaction between them. A prominent instance of this primitive implements the CZ gate, as illustrated in Figure 2.4a.

Using the rules of Heisenberg operator evolution, one can verify that the equivalent gate-based circuit shown in Figure 2.4b realizes the same conjugation of logical operators as a controlled-phase gate acting between $|\psi_1\rangle$ and $|\psi_4\rangle$. This equivalence is depicted in Figure 2.4c.

Combined with the teleportation primitive, measurement-induced entanglement suffices to perform arbitrary quantum computation on a two-dimensional graph state. We will return later to the requirements for universal and fault-tolerant quantum computation within this model. However, it is worth emphasizing here that this entangling operation plays a foundational role in fault-tolerant quantum computing more broadly, even outside the MBQC formalism. In particular, this primitive underlies the realization of the logical CX through lattice surgery in the surface code [104], and it generalizes naturally to the context of quantum low-density parity-check (qLDPC) codes through recent developments in generalized lattice surgery [34, 36, 37].



Figure 2.4: **a** Cluster state representation of the remote entanglement primitive. **b** The remote entanglement primitive from the gate-based view consists of a low-depth circuit and measurement of the central qubits in the X basis. **c** Evolution of the unmeasured qubits in the Heisenberg picture.

2.3 Classical Error Correction

Classical error correction addresses the problem of transmitting messages through a noisy channel. Typically, a message consists of a sequence of bits, known as a *bit string*, which can also be referred to as a *word* in the context of information processing. Each element of a bit string belongs to the finite field GF(2), commonly denoted as \mathbb{F}_2 , containing the elements 0 and 1. Consequently, a bit string of length n can be viewed as a vector in the vector space \mathbb{F}_2^n .

When Alice wants to communicate with Bob, she sends a message u, a bit string, through the channel. Bob then receives the message, which is called the *received vector* and denoted by v. In the ideal case of a noiseless channel, Bob receives exactly the message Alice sent, that is, u = v. However, in the general case where noise is present, the message is corrupted during transmission. The error e that occurs during this process is defined as the difference between the sent and received messages:

$$\boldsymbol{e} = \boldsymbol{u} - \boldsymbol{v}.\tag{2.20}$$

It is crucial to note that the error vector e is unknown to both Alice and Bob. The goal of classical error correction is to infer the error e based only on the received vector v, and to recover the original message u.

2.3.1 Classical Noise Channels

There is a plethora of different ways a noisy channel may act upon the bit string \boldsymbol{u} . We will focus on the three most relevant ones for this thesis, which share a common feature, namely, that they act on each bit in the bit string individually. We also show a pictorial representation of them in Figure 2.5.

The erasure channel. The erasure channel can be considered the simplest form of a noisy channel. Each bit in a bit string is *erased* with a certain probability p_e . An erasure is typically represented by replacing the binary value with a "?", indicating complete uncertainty about the value of that bit. Maybe counter-intuitively, an error e due to an erasure channel is, in many cases, easier to correct than errors stemming from other channels. Intuitively, the reason is that while for erased bits there is complete uncertainty about their value, we have precise information about which bits are erased and which ones are not, and thus, which ones are not to be trusted and which ones can be trusted.

The binary symmetric channel. The binary symmetric channel can be considered one of the most important models for noisy communication, and we will mostly focus on this channel. The action of this channel is to flip the value of each transmitted bit with probability p < 0.5, changing it from 0 to 1 or from 1 to 0, depending on the transmitted word \boldsymbol{u} . As the name implies, the channel is symmetric with respect to 0 and 1 such that we can assign a probability to an error \boldsymbol{e} . To this end, it is instructive to define the *weight* wt(\boldsymbol{e}) of an error $\boldsymbol{e} \in \mathbb{F}_2^n$ as the number of nonzero components of \boldsymbol{e} , also known as the Hamming weight of \boldsymbol{e} . In the simplest case, if the channel acts identically on all bits, the probability for a given error e to occur is proportional to $p^{\text{wt}(e)}$, making errors of small weight more probable. This is important as one cannot correct arbitrary errors, and our goal is to be able to correct errors that only affect a small fraction of components of v.

The additive white Gaussian noise channel. The additive white Gaussian noise channel, also AWGN channel, describes a basic model that aims to model various random processes in nature and is particularly useful for satellite-based communication, for example. It is distinct from the erasure and the binary symmetric channel, as this channel considers a continuous type of noise in comparison to the previously considered discrete channels. The name implies specific characteristics, that is,

• additive — the noise term acts additively to the transmitted signal. For this noise model, it is common to assume the transmitted bit x to have values ± 1 instead of 0/1 such that the noisy "bit" y is obtained as

$$y = x + \delta, \tag{2.21}$$

where $\delta \in \mathbb{R}$ is a random variable connected to the noise,

- *white* refers to the idea that the power spectral density is uniform across a frequency range, as this is a noise model relevant for wireless communication,
- Gaussian the noise, i.e., the random variable $\delta \sim \mathcal{N}(\mu, \sigma^2)$, follows a Gaussian distribution characterized by a mean μ and variance σ^2 .

In practice, the received vector \boldsymbol{y} is usually quantized, and the magnitude of the elements of \boldsymbol{y} is used as *soft-information* to infer the send message \boldsymbol{x} . This quantization allows one to treat this channel as a binary symmetric channel, but with a varying prior from shot to shot and for each received bit. Thus, for the following discussion, we restrict our attention to discrete noise channels. However, the definitions and techniques introduced here can be naturally extended to the continuous setting.



Figure 2.5: Pictorial representation of three different classical noise channels, from left to right, the erasure channel, the binary symmetric channel, and the additive white Gaussian noise channel.

2.3.2 Fundamentals of Error-Correcting Codes

All these channels effectively specify a set of errors E. We say that with respect to E two words u_1 and u_2 are *distinguishable* if and only if

$$\boldsymbol{u}_1 + \boldsymbol{e}_1 \neq \boldsymbol{u}_2 + \boldsymbol{e}_2, \quad \forall \boldsymbol{e}_1, \boldsymbol{e}_2 \in E.$$
 (2.22)

Clearly, for bit strings of length n not all words, that is, all elements of \mathbb{F}_2^n , can be distinguished if E is nontrivial. Instead, we should restrict to a subset C of words from \mathbb{F}_2^n . The subset C then is an *error-correcting code* of length n and the elements of C are called *code words*. If the size of C is $|C| = 2^k$ then we say that C encodes k (logical) bits. A set of errors E is correctable by C if all pairwise combinations of code words are distinguishable with respect to E. Suppose E(t) contains errors of bounded weight, that is, wt $(e) \leq t, \forall e \in E$. If C corrects E(t) but does not correct E(t+1) then we say that C is a *t-error-correcting code*.

It is more common, however, to characterize C by its distance d(C). The distance of two code words c_1 and c_2 is defined as $d(c_1, c_2) = \text{wt}(c_1 - c_2)$. The code distance d(C) is the minimum distance between any pair of code words, that is,

$$d(C) = \min_{c_1, c_2 \in C, c_1 \neq c_2} d(c_1, c_2) = \min_{c_1, c_2 \in C, c_1 \neq c_2} \operatorname{wt}(c_1 - c_2).$$
(2.23)

A code C is t-error-correcting if and only if its minimum distance d satisfies $d \ge 2t + 1$. This follows because if two distinct code words were indistinguishable by errors of weight at most t, then the error difference would have weight at most 2t, contradicting the definition of the distance. For a code C of length n that encodes k (logical) bits with distance d, C is typically denoted as an [n, k, d] code.

2.3.3 Linear Block Codes

There are many classes of codes, but we will focus on *linear (block) codes*. These codes have certain properties that make them convenient and extendable to quantum errorcorrecting codes. A linear [n, k, d] code is a k-dimensional subspace C of \mathbb{F}_2^n with distance $d = \min_{\mathbf{c} \in C/\{\mathbf{0}\}} \operatorname{wt}(\mathbf{c})$. The code C can be efficiently specified by a full-rank¹ $(n-k) \times n$ *parity-check matrix* H over \mathbb{F}_2 , that is,

$$C = \left\{ \boldsymbol{c} \in \mathbb{F}_2^n \mid H\boldsymbol{c} = 0 \right\},\tag{2.24}$$

where we assume that any vector is a column vector. The parity-check matrix, therefore, identifies correctable errors independent of the transmitted code word, i.e., $H(c_1 + e) = H(c_2 + e) = He$. The product He is referred to as the syndrome s and

$$\boldsymbol{s} = H\boldsymbol{e},\tag{2.25}$$

is the syndrome equation.

 $^{{}^{1}}H$ doesn't need to be full rank; redundant parity checks are allowed and define the same code. However, there always exists a full-rank matrix \tilde{H} that specifies the same code. Nevertheless, including redundant checks can be relevant in certain settings, for example, in the classical theory of locally testable codes [105], or in quantum error correction schemes where redundancy enables single-shot decoding [106].

Through the parity-check matrix, linear codes admit a natural interpretation. To this end, consider an arbitrary element $\boldsymbol{x} \in \mathbb{F}_2^n$ that we write as $x_1x_2...x_n$. For \boldsymbol{x} to be a code word as specified by Eq. (2.24), \boldsymbol{x} needs to fulfill a set of constraints specified by the rows of the parity-check matrix H. These constraints are also sometimes referred to as *parity constraints* as all algebra is done over \mathbb{F}_2 such that a constraint is either fulfilled (0) or not (1). As there are only n - k constraints for n variables, the linear system of equations $H\boldsymbol{x} = \boldsymbol{0}$ is under-constrained, leaving k degrees of freedom that are connected to the k encoded bits. This redundancy is a necessary ingredient for error-correcting codes, and one can derive criteria for an [n, k, d] code to exist, see, e.g., Refs. [107, 108].

The relation between redundancy and encoded bits becomes even more apparent if one considers an alternate definition of the code space C. To this end, note that the rows of H do not form a basis of the code space C but instead its orthogonal complement C^{\perp} , that is, the subspace of vectors that are orthogonal to all vectors in C. Instead, the matrix whose rows specify a basis for C is called the generator matrix $G \in \mathbb{F}_2^{k \times n}$ and it satisfies $GH^T = 0$ such that it is determined by H. Given the generator matrix G and the check matrix H, we can define the $n \times n$ invertible encoding matrix V over \mathbb{F}_2 as [109]

$$V = \begin{pmatrix} G\\ (H^{-1})^T \end{pmatrix},\tag{2.26}$$

where, by slight misuse of notation, we denote by H^{-1} the right inverse of H over \mathbb{F}_2 such that $HH^{-1} = \mathbb{1}_{n-k}$, with $\mathbb{1}_{n-k}$ the identity operation on \mathbb{F}_2^{n-k} . Then, the code space can be equivalently defined in terms of the encoding matrix as

$$C = \left\{ \boldsymbol{c} = (\boldsymbol{b} : \boldsymbol{0}_{n-k})^T V \mid \boldsymbol{b} \in \mathbb{F}_2^k \right\}, \qquad (2.27)$$

where the notation $(\boldsymbol{a}:\boldsymbol{b})$ denotes the concatenation of \boldsymbol{a} followed by \boldsymbol{b} as column vectors. The representation of the code space through the encoding matrix is rather uncommon, however, it is beneficial to understand the connection between error-correcting codes and quantum stabilizer codes. For this note that for a noisy code word $\boldsymbol{v} = \boldsymbol{c} + \boldsymbol{e}$ resulting from the binary symmetric channel, we can use the inverse encoding matrix V^{-1} to decompose \boldsymbol{v} into the syndrome $\boldsymbol{s} \in \mathbb{F}_2^{(n-k)}$ and a logical error $\boldsymbol{\ell} \in \mathbb{F}_2^k$ as $\boldsymbol{e}^T V^{-1} = (\boldsymbol{\ell}:\boldsymbol{s})$. The reason that both the logical error $\boldsymbol{\ell}$ and the syndrome \boldsymbol{s} appear here is that an uncorrectable error from the set E(t+1) typically will have the same syndrome as a correctable error from E(t). As the encoding matrix is invertible, opposed to the parity-check matrix H, the logical error $\boldsymbol{\ell}$ encodes the additional information that is not contained in \boldsymbol{s} that is necessary to reconstruct \boldsymbol{e} .

We will delay a detailed discussion on decoding algorithms that can be used to obtain an estimate of the error e until Section 2.6. Here, we simply note that a key advantage of linear codes is that many of them can be constructed such that their structure allows one to efficiently infer the error from the syndrome. An instructive example for this is the [7,4,3] code, also known as the Hamming code, that does not require a dedicated decoding algorithm in the conventional sense. The parity-check matrix of the Hamming code is given by

$$H = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$
 (2.28)

As the columns are the binary representation of the numbers from one to seven, the syndrome $s = s_1 s_2 s_3$ directly gives the error location in binary representation (for a single error).

2.3.4 Tanner graphs

The Tanner graph is a useful tool for visualizing the structure of the parity-check matrix H. To this end, one can associate H with the biadjacency matrix of a bipartite graph $G = (V_v \cup V_c, E)$ consisting of variable nodes V_v , check nodes V_c , and edges E between V_v and V_c . In particular, there is an edge $(v_i, c_j) \in E$ between a variable node $v_i \in V_v$ and a check node $c_j \in V_c$ if and only if H[j, i] = 1, where H[j, i] denotes the j^{th} row and i^{th} column of the matrix H. As the variable nodes correspond to the bits of the linear codes, we call them bit nodes synonymously. In Figure 2.6 we show the Tanner graph of the Hamming code defined in Eq. (2.28). In this representation, we typically represent bit nodes as circle nodes and check nodes as squares (or boxes). As Tanner graphs contain the structure of the parity-check matrix, they are incredibly useful for the formulation of decoding algorithms. They can be extended to the case of quantum codes, and they play an important role in fault-tolerant measurement-based quantum computing as well.



Figure 2.6: Tanner graph of the [7, 4, 3] Hamming code defined in Eq. (2.28). Variable nodes v_i are represented as circles and check nodes c_i as squares.

2.4 Quantum Error Correction

The role of quantum error correction is to protect quantum information from noise. However, this task is significantly more intricate than its classical counterpart. For some time, it was believed that quantum error correction might be impossible due to the continuous nature of quantum errors, reminiscent of error accumulation in classical analog computing. Fortunately, this pessimism was misplaced. As is often the case in science, the word "impossible" was eventually replaced by the less daunting "difficult."

In 1995, Peter Shor [110], and independently, Andrew Steane [111, 112] demonstrated that it is possible to measure a carefully chosen set of observables that detect the presence of errors without disturbing the encoded quantum information. These measurements not only reveal whether an error has occurred, but also discretize the noise: they project the state into one of several orthogonal subspaces, effectively collapsing a continuous error into a discrete one. Importantly, these observables do not distinguish between different logical states, but only between error-free and erroneous subspaces. One can then use measurement outcomes to infer which kind of error has occurred. It turns out that many tools of classical coding theory can be applied from here.

2.4.1 Noise channels

As in the classical case, designing quantum error-correcting codes requires choosing a noise model that reflects the types of errors the code is meant to protect against. A physically motivated example is the single-qubit depolarizing channel, defined as

$$\mathcal{E}_1(\rho) = (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z), \qquad (2.29)$$

where ρ is the density operator of the quantum system. This channel leaves the state unchanged with probability 1 - p, and applies a random single-qubit Pauli error X, Y, or Z with probability p/3. It captures decoherence due to independent noise on each qubit.

However, this model is typically insufficient for describing realistic noise in quantum circuits. Two-qubit gates, such as the controlled-NOT (CX) gate, often introduce correlated noise. Such effects can be modeled using the two-qubit depolarizing channel,

$$\mathcal{E}_{2}(\rho) = (1-p)\rho + \frac{p}{15} \sum_{E \in \tilde{\mathcal{P}}_{2}/\{II\}} E\rho E^{\dagger}, \qquad (2.30)$$

where $\widetilde{\mathcal{P}}_2$ denotes the effective two-qubit Pauli group $\mathcal{P}_2/\{\pm 1, \pm i\}$.

In circuit-level noise models, noisy single-qubit gates are typically modeled as the ideal gate followed by the single-qubit depolarizing channel (2.29), and noisy two-qubit gates as the ideal gate followed by the two-qubit depolarizing channel (2.30).

Although such discrete noise models may appear overly simplistic — given that quantum systems evolve continuously — the act of measurement discretizes this evolution in a physically meaningful way. To understand this, consider again the Bell state (2.5) $|\Phi^+\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$. Suppose the first qubit undergoes a small rotation generated by the Pauli X operator, leading to the evolved state

$$\exp(i\delta X_1 t/2) \left| \Phi^+ \right\rangle = \cos(\delta t) I_1 \left| \Phi^+ \right\rangle + i \sin(\delta t) X_1 \left| \Phi^+ \right\rangle.$$
(2.31)

Now, imagine measuring the ZZ parity as described in Section 2.1.2. If $\delta t \ll 1$, with probability $\cos^2(\delta t)$ close to one, we obtain the outcome +1, and the state collapses back to the original Bell state. With small probability $\sin^2(\delta t)$, we instead obtain the outcome -1, projecting the state onto $X_1 |\Phi^+\rangle$, which is equivalent to a discrete Pauli error on the first qubit.

This example illustrates a key point: even continuous errors become discrete upon measurement of appropriate observables that anti-commute with the (generator of the) error. More generally, because any quantum operation can be decomposed into a superposition of Pauli operators, correcting all Pauli errors of weight less than t ensures the correction of arbitrary errors affecting at most t qubits.

2.4.2 Stabilizer codes

Quantum stabilizer codes [113] can be viewed as the quantum analog of classical linear block codes. Abstractly, a stabilizer code is defined by an Abelian subgroup S of the *n*-qubit Pauli group \mathcal{P}_n , with the condition that $-I \notin S$. The group S is called the *stabilizer group*, and its elements are referred to as *stabilizers*. The code space is defined as the joint +1-eigenspace of all elements in S,

$$C = \{ |\psi\rangle \mid S \mid \psi\rangle = |\psi\rangle, \ \forall S \in \mathcal{S} \}.$$

$$(2.32)$$

The stabilizer group is typically specified by a set of independent generators $S_1, S_2, \ldots, S_r \in \mathcal{P}_n$, such that $\mathcal{S} = \langle S_1, S_2, \ldots, S_r \rangle$. Each generator defines a constraint that any valid code state must satisfy. If the stabilizer group has r independent generators, the code encodes k = n - r logical qubits into n physical qubits. This mirrors the role of parity-check constraints in classical linear codes, a connection that will be made explicit later.

Having defined the stabilizer group and its generators, we now turn to the logical operators of the code. Logical operators correspond to symmetries of the code space that preserve the code space but act non-trivially on the logical qubits. The logical operators \mathcal{L} are the set of operators that commute with \mathcal{S} but are themselves not elements of \mathcal{S} , or more formally, the logical operators are given by

$$\mathcal{L} \cong \mathcal{N}(\mathcal{S})/\mathcal{S} = \{ L \in \mathcal{P}_n \mid SL = LS \; \forall S \in \mathcal{S} \}/\mathcal{S}, \tag{2.33}$$

where we write $\mathcal{N}(\mathcal{S})$ for the normalizer of \mathcal{S} in \mathcal{P}_n .

The distance d of the code is defined as the minimum weight of a non-trivial logical operator, i.e., the smallest number of qubits on which such an operator acts non-trivially. We use the notation [n, k, d] to refer to a stabilizer code with n qubits, k encoded logical qubits, and distance d. For a code encoding k logical qubits, there will be 2k logical operators, an encoded Pauli \overline{Z}_j and \overline{X}_j for each logical qubit. As the minimum weight of \overline{Z}_j and \overline{X}_j are not necessarily equal, one also encounters the notation $[n, k, d_X, d_Z]$ for stabilizer codes.

In the absence of errors, measuring the generators of the stabilizer group S will deterministically have outcome +1. This is because all code states $|\psi\rangle \in C$ are stabilized by every $S_i \in S$, i.e., $S_i |\psi\rangle = |\psi\rangle$. However, in the presence of errors, any generator S_i that anti-commutes with an error $E \in \mathcal{P}_n$ results in a measurement outcome -1, as $S_i E |\psi\rangle = -E g_i |\psi_i\rangle = -E |\psi\rangle$. By measuring all generators S_i of the stabilizer group, we obtain a list of measurement outcomes $\sigma(E)$ known as the syndrome. We typically view $\sigma(E)$ as a bit string $s \in \mathbb{F}_2^{n-k}$ that encodes the commutation relations of the errors with the generators, that is, $ES_i + (-1)^{s_i}S_iE = 0$ and make the dependence of s on the error E implicit.

We note that the elements of the stabilizer group, together with the logical operators, do not generate the full Pauli group \mathcal{P}_n . As a result, we cannot represent any error E as a product of a logical operator $L \in \mathcal{L}$ and a stabilizer $S \in \mathcal{S}$, as we could in the classical case (see the discussion around Eq. (2.27)). This discrepancy arises because \mathcal{P}_n contains 4^n elements (up to phases), whereas the stabilizer group \mathcal{S} contains 2^{n-k} elements and the logical operators \mathcal{L} contain 2^{2k} independent elements. Together, \mathcal{S} and \mathcal{L} contribute 2^{n+k} elements, leaving the remaining $4^n/2^{n+k} = 2^{n-k}$ elements to be called *pure errors* or *destabilizers*, denoted \mathcal{T} . These pure errors are formally defined as

$$\mathcal{T} \cong \mathcal{N}(\mathcal{L})/\mathcal{S}. \tag{2.34}$$

Together with the stabilizers S and logical operators \mathcal{L} , the destabilizers \mathcal{T} complete the set of operators that span the full *n*-qubit Pauli group \mathcal{P}_n . Any operator $E \in \mathcal{P}_n$ can therefore be represented as

$$E = TLS, (2.35)$$

where $T \in \mathcal{T}$, $L \in \mathcal{L}$, and $S \in \mathcal{S}$.

While the notion of pure errors is useful for the optimal decoding of stabilizer codes, as we will discuss later, their introduction also provides insight into an alternative characterization of a stabilizer code. Alternatively to Eq. (2.32) a stabilizer code can be specified through a Clifford transformation V on n qubits such that

$$C = \{ |\psi\rangle = V(|\varphi\rangle \otimes |0_{n-k}\rangle) \mid |\varphi\rangle \in \mathbb{C}^{2^k} \},$$
(2.36)

where V is known as the encoding unitary, analogous to the encoding matrix in the classical linear code definition in Eq. (2.27). Since the encoding matrix is an element of the Clifford group, it can be implemented using Clifford operations, such as CX, S, and H, only. This encoding unitary implicitly determines the logical operators, stabilizer generators, and the generators of the pure errors. These generators are obtained by conjugating the single-qubit Pauli operators, as follows:

$$V: \begin{cases} Z_j \to \overline{Z}_j = VZ_j V^{\dagger}, & \text{for } j = 1 \dots k, \\ X_j \to \overline{X}_j = VX_j V^{\dagger}, & \text{for } j = 1 \dots k, \\ Z_{j+k} \to g_j = VZ_{j+k} V^{\dagger}, & \text{for } j = 1 \dots n-k, \\ X_{j+k} \to T_j = VX_{j+k} V^{\dagger}, & \text{for } j = 1 \dots n-k. \end{cases}$$

$$(2.37)$$

Binary symplectic representation. To make the connection between classical linear codes and stabilizer codes more explicit, we now introduce the *binary symplectic representation* due to Ref. [114]. This representation defines a map $\operatorname{Mat}_{\mathcal{B}} : \mathcal{P}_n \to \mathbb{F}_2^{2n}$ such that a *n*-qubit Pauli operator can be written as a vector with elements in \mathbb{F}_2 , that is,

$$\alpha \bigotimes_{i=1}^{n} X^{x_i} Z^{z_i} \to (x_1, x_2, \dots, x_n \mid z_1, z_2, \dots, z_n).$$
(2.38)

We will typically ignore the phases $\alpha \in \{\pm 1, \pm i\}$ of elements in the Pauli group. The reason is that in the context of quantum error correction, we typically identify Pauli operators that differ only by a phase α , as such phases do not affect the syndrome. Additionally, whenever the stabilizer group S that specifies the stabilizer code C (2.32) contains an element S_i with phase -1, we can replace S_i with $-S_i$ without changing properties of the code. Hence, the phase α is irrelevant for error correction, but it is relevant for the simulation of Clifford circuits with measurements [96]. Through the binary symplectic representation, we can specify the stabilizer group $S = \langle S_1, \ldots, S_r \rangle$ that defines a stabilizer code C through Eq. (2.32) by an $r \times 2n$ binary parity-check matrix H where the i^{th} row is $\text{Mat}_{\mathcal{B}}(S_i)$. Multiplying two Pauli strings $P_1, P_2 \in \mathcal{P}_n$ corresponds to a simple addition within the binary symplectic representation, i.e., $\text{Mat}_{\mathcal{B}}(P_1) + \text{Mat}_{\mathcal{B}}(P_2) = \text{Mat}_{\mathcal{B}}(P_1P_2)$. P_1 and P_2 commute if and only if the symplectic inner product vanishes, i.e., $\text{Mat}_{\mathcal{B}}(P_1)\Lambda \text{Mat}_{\mathcal{B}}(P_2)^T = 0$, where Λ is the symplectic matrix and given by

$$\Lambda = \begin{bmatrix} 0 & \mathbb{1}_n \\ \mathbb{1}_n & 0 \end{bmatrix},\tag{2.39}$$

with $\mathbb{1}_n$ the $n \times n$ identity matrix. Hence, for S to be Abelian we require that $H\Lambda H^T = H_X H_Z^T + H_Z H_X^T = 0$ where we used that we can write $H = [H_X \mid H_Z]$. Similar to the classical case, the rank of the parity-check matrix is related to the number of encoded logical qubits, that is, it determines the number of independent generators of S, such that $k = n - \operatorname{rank}(H)$. It is instructive to consider the five-qubit code as an example. The stabilizer group of the five-qubit code can be written as $S = \langle XZZXI, IXZZ, XIXZZ, ZXIXZ \rangle$, which has the following binary symplectic representation

$$H = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}.$$
 (2.40)

It can be verified that the five-qubit code has parameters [5, 1, 3].

Calderbank-Shor-Steane codes. A particularly important class of stabilizer codes are Calderbank-Shor-Steane (CSS) codes [111, 112, 115] whose defining property is that their stabilizer group admits a set of generators S_1, S_2, \ldots, S_r such that they are either X-type or Z-type, that is, their check matrix H admits a block diagonal structure,

$$H = \begin{bmatrix} H_X & 0\\ 0 & H_Z \end{bmatrix},\tag{2.41}$$

where $H_X \in \mathbb{F}_2^{m_x \times n}$ and $H_Z \in \mathbb{F}_2^{m_x \times n}$. This particular structure simplifies the commutativity condition to

$$H_X H_Z^T = 0 \tag{2.42}$$

and the number of encoded logical qubits becomes $k = n - \operatorname{rank}(H_X) - \operatorname{rank}(H_Z)$. The additional structure inherent to CSS codes significantly simplifies their construction, making them a powerful tool in quantum error correction. As a result, many breakthrough results in the field have been achieved through CSS codes. These results have relied upon what is today often referred to as the "CSS-to-homology" correspondence, a mathematical framework that employs *chain complexes*, objects that are studied in homological algebra, to represent not only CSS codes but also classical linear codes. This formulation has become the modern perspective on (quantum) error correction and is expected to influence the field further. We will give a brief introduction to this formalism after reviewing how fault-tolerance properties of CSS stabilizer codes can be lifted to the measurement-based setting.

2.4.3 Fault-tolerant graph states

We have seen previously that universal quantum computation can be achieved in the measurement-based model by using a two-dimensional cluster state as a resource. This model mimics the circuit model in that logical gates are implemented through sequences of single-qubit measurements. However, to protect such computations against noise, additional redundancy is required. Unlike in the circuit model, where stabilizer measurements can be used to identify and correct errors during computation, the measurement-based approach requires that fault tolerance can be built directly into the resource state. This raises the question: how can one design such a resource state so that it supports fault-tolerant quantum computation? One answer is given by the method of *foliation*, which produces a fault-tolerant cluster state from any CSS code. This method, due to Bolt *et al.* [116], generalizes the construction of Raussendorf *et al.* [117] for the surface code.

Recall that a CSS code is characterized by two parity-check matrices H_Z and H_X . As described in Section 2.3.4, with each of these matrices one can associate a bipartite graph $G = (V_b \cup V_c, E)$. We can also associate with such a graph G a graph state or cluster state by identifying the vertex sets V_b and V_c with code qubits and check qubits, respectively. The edges E in the graph state indicate between which qubits a CZ gate is applied, see Section 2.2.1.

Consider the graph state G_Z associated with the Z parity-check matrix H_Z as described above. It is straightforward to see that the stabilizers centered on the ancilla qubits of the graph state can be written as

$$S_{c_j} = X_{c_j} \bigotimes_{i=1}^{n} Z_{b_i}^{H[j,i]}, \qquad (2.43)$$

for any $c_j \in V_c$ and the notation H[j, i] denotes the i^{th} entry in the j^{th} row associated to the check c_j . Thus, upon measurement of the ancilla qubits in the Pauli X basis, the resulting graph state is an eigenstate of the stabilizer group described by H_Z . Importantly, the graph state G_Z is also an eigenstate of the stabilizer group generated by H_X . The reason is that H_X and H_Z must commute and thus X- and Z-stabilizers must overlap on an even number of code qubits. As a result, when constructing X-type stabilizers associated with elements S_X from graph state stabilizers of code qubits $S_b = X_b \otimes_{a \in N(b)} Z_a$, any ancilla qubit *a* appears an even number of times. Since logical X operators can be seen as a special form of X-type stabilizer, the same argument applies to them. Hence, measurement of the ancilla qubits in the graph state projects the remaining qubits into the logical dual basis code state of the CSS code described by H_X and H_Z .

By layering the graph states G_Z and G_X in an alternating pattern and entangling code qubits between adjacent layers by CZ gates, one obtains the *foliated code*, see Figure 2.7 for an example. The foliated code is a fault-tolerant graph state that inherits properties of the underlying CSS code and can protect against Z-type errors on code and ancilla qubits. Within this fault-tolerant graph state, one distinguishes between two types of qubits: primal and dual qubits. Code qubits in G_Z and ancilla qubits in G_X are considered primal, while code qubits in G_X and ancilla qubits in G_Z are considered dual. This assignment aligns with the alternating structure of the foliated code and reflects the duality between X- and Z-type checks. Parity-check operators, also called detectors, can be identified within the graph state and inferred from sets of single-qubit X measurement outcomes. The parity of individual detectors yields a syndrome s which can be used together with a decoder to infer the presence and location of Z errors. We will not reproduce here the expression for the detectors of the foliated code and instead refer to Ref. [116] for the original derivation or to **Paper IX** for a derivation based on chain complexes and the hypergraph product, which we introduce in the next section.

Lastly, we mention that foliated codes do not belong to the class of stabilizer codes described in Section 2.4.2, but instead more closely resemble subsystem CSS stabilizer codes [118]. Furthermore, while we described the foliation of CSS codes, the approach can be extended to ordinary stabilizer codes, see Ref. [119].



Figure 2.7: Foliation of the $[\![4, 1, 2]\!]$ CSS code described by check matrices $H_Z = [1 \ 1 \ 1 \ 1]$ and $H_X = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$. The fault-tolerant graph state shown contains L = 2 foliation layers. Gray and red nodes indicate primal and dual qubits, respectively. Highlighted in blue is a set of X measurement outcomes that yield a detector of the foliated code, that is, a product of measurement outcomes that is deterministic in the absence of errors.

2.5 Chain Complexes

For a more thorough introduction to chain complexes, we refer the interested reader to Refs. [21, 120]. For the purpose of this thesis, a *chain complex* C of length n is defined as a sequence of n + 1 \mathbb{F}_2 -vector spaces C_i and n linear maps ∂_i^C called boundary operators,

$$\mathcal{C} = \{0\} \xrightarrow{\partial_{n+1}^C} C_n \xrightarrow{\partial_n^C} \dots \xrightarrow{\partial_1^C} C_0 \xrightarrow{\partial_0^C} \{0\}, \qquad (2.44)$$

with the composition of boundary operators fulfilling

$$\partial_i^C \partial_{i+1}^C = 0. \tag{2.45}$$
Here, the spaces C_i always have finite dimension n_i , and a basis of C_i can be identified with $\mathbb{F}_2^{n_i}$. We suppress the superscripts if no distinction is necessary, as well as make implicit the trivial boundary operators $\partial_{n+1} : \{0\} \to C_n$ and $\partial_0 : C_0 \to \{0\}$ that are treated formally as zero $n_n \times 0$ and $0 \times n_0$ matrices, respectively.

It is common to refer to elements of the kernel $Z_i(\mathcal{C}) \coloneqq \ker \partial_i$ as *i*-cycles, and elements of the image $B_i(\mathcal{C}) \coloneqq \operatorname{im} \partial_{i+1}$ as *i*-boundaries, and one has the embedding relation

$$B_i \subseteq Z_i \subseteq C_i. \tag{2.46}$$

The quotient vector space

$$H_i(\mathcal{C}) = Z_i/B_i = \ker \partial_i / \operatorname{im} \partial_{i+1}, \qquad (2.47)$$

is called the i^{th} homology group of C and is a central object of interest for the study of stabilizer codes through the presented formalism.

Associated with C is also a cochain complex, with coboundary operators $\delta^i : C_i \to C_{i+1}$ defined as $\delta^i = \partial_{i+1}^T$. This cochain complex is given by

$$\mathcal{C}^{T} = \{0\} \stackrel{\delta^{-1}}{\to} C_{0} \stackrel{\delta^{0}}{\to} \dots \stackrel{\delta^{n-1}}{\to} C_{n} \stackrel{\delta^{n}}{\to} \{0\},$$
(2.48)

with cohomology groups defined as

$$H^{i} \coloneqq \ker \delta^{i} / \operatorname{im} \delta^{i-1}.$$
(2.49)

Elements of $Z^i := \ker \delta^i$ and $B^i := \operatorname{im} \delta^{i-1}$ are referred to as cocycles and coboundaries, respectively. Thus, the *i*th cohomology group of \mathcal{C} is defined as

$$H^{i}(\mathcal{C}) = Z^{i}/B^{i} = \ker \delta^{i}/\operatorname{im} \delta^{i-1} = \ker \partial^{T}_{i+1}/\operatorname{im} \partial^{T}_{i}, \qquad (2.50)$$

which is similarly important to the i^{th} homology group of C in capturing the essential features of the underlying structure.

2.5.1 The CSS-to-homomology correspondence

Since the boundary maps of a chain complex \mathcal{C} can be identified with matrices over \mathbb{F}_2 , any length-1 chain complex can be viewed as a classical linear [n, k, d] code by identifying $\partial_1 = H$, where H maps an error vector from the space $C_1 \cong \mathbb{F}_2^n$ to the syndrome space $C_0 \cong \mathbb{F}_2^r$ with $r \ge n - k$. The codespace, denoted by C, coincides with the first homology group of \mathcal{C} . In other words, for a length-1 chain complex with boundary map $\partial_1 = H$, the first homology group is given by

$$H_1(\mathcal{C}) = \ker H / \operatorname{im} \partial_2 = \ker H / \{\mathbf{0}\}, \qquad (2.51)$$

and the smallest non-trivial weight element of $H_1(\mathcal{C})$ determines the distance d.

Notice that the defining property of a chain complex, namely the composition property of boundary operators in Eq. (2.45), is equivalent to the commutation condition (2.42) for the check matrices of a CSS code.

Thus, any CSS code, C, can be represented by a length-2 chain (sub)complex

$$\dots \to C_{i+1} \xrightarrow{\partial_{i+1}} C_i \xrightarrow{\partial_i} C_{i-1} \to \dots,$$
(2.52)

where, by convention, $\partial_{i+1} = H_Z^T$ and $\partial_i = H_X$. Identifying qubits with the space C_i , the number of qubits is $n = \dim C_i$.

The sets of X and Z logical operators are elements of the groups $H_i(\mathcal{C})$ and $H^i(\mathcal{C})$, respectively, with the smallest non-trivial weight element defining the distances d_X and d_Z , respectively. Intuitively, for a CSS code, a logical Z operator from the group

$$H_i(\mathcal{C}) = \ker H_X / \operatorname{im} H_Z^T, \tag{2.53}$$

is an element ℓ of C_i that does not produce an X-syndrome (i.e., $\ell \in \ker H_X$) and is not generated by the Z-stabilizers (i.e., $\ell \notin \operatorname{im} H_Z^T$).

2.5.2 Quantum codes from classical codes

Why should we explore this abstract formalism for representing a CSS code? To address this question, we first need to highlight two key points. First, while we can represent any CSS code as a length-2 chain complex, the reverse is also true: any length-2 chain complex corresponds to a valid CSS code. This equivalence between chain complexes and CSS codes forms the foundation for understanding how classical codes can be used to design quantum codes. Second, constructing good qLDPC codes — stabilizer codes with sparse check matrices and asymptotically linear parameters — has proven to be significantly more challenging than in the classical case. For classical linear codes, Gallager demonstrated in 1960 [121, 122] that taking a random sparse parity-check matrix is sufficient to obtain a good low-density parity-check (LDPC) code with parameters $[n, \Theta(n), \Theta(n)]$. However, the commutation condition for quantum codes prevents a similar straightforward construction.

It was only in 2021, following a series of breakthrough results [22–24, 123], that the existence of good qLDPC codes was established. These advancements largely rely on the formalism described here and the development of product constructions for pairs of chain complexes, while also applying techniques due to Sipser and Spielman [124] for the explicit construction of good LDPC codes in the classical setting. Interested readers can refer to Refs. [21, 125], and we now consider a simple example — the hypergraph product (or homological product) [126, 127] which illustrates how to construct quantum codes from classical ones. For a generalization to higher-dimensional cases, see Ref. [128].

Hypergraph product codes. Consider two length-1 chain complexes, $\mathcal{A} : A_1 \xrightarrow{H_A} A_0$ and $\mathcal{B} : B_1 \xrightarrow{H_B} B_0$, corresponding to two classical linear codes with parameters $[n_i, k_i, d_i]$ and $m_i \times n_i$ check matrices H_i for $i \in \{A, B\}$. The hypergraph product code $\mathcal{C} = \mathcal{A} \times \mathcal{B}$ is a length-2 chain complex with boundary operators given by:

$$\partial_1 = H_X = \left(H_A \otimes \mathbb{1}_{n_B} \mid \mathbb{1}_{m_A} \otimes H_B \right), \quad \partial_2^T = H_Z = \left(\mathbb{1}_{n_A} \otimes H_B \mid H_A \otimes \mathbb{1}_{m_B} \right), \quad (2.54)$$

and spaces given by

$$C_2 = A_1 \otimes B_1, \quad C_1 = A_1 \otimes B_0 \oplus A_0 \otimes B_1, \quad C_0 = A_0 \otimes B_0.$$
 (2.55)

The resulting chain complex corresponds to a CSS quantum code with parameters $[n_A m_B + m_A n_B, k_A k_B^T + k_A^T k_B, \min(d_A, d_B, d_A^T, d_B^T)]]$, where k_i^T and d_i^T are the parameters of the transpose codes. We use the convention that $d = \infty$ if k = 0.

Thus, through this product construction, we can generate quantum codes by combining classical codes in a structured way that preserves their error-correcting properties while satisfying the constraints of quantum error correction.

To exemplify this procedure and illustrate the origin of the name, let us consider a simple example, the product of two classical repetition codes. The length-n repetition code is a classical code with parameters [n, 1, n], and its codewords are the all-zero and all-one vectors of length n, that is $\mathbf{0}_n$ and $\mathbf{1}_n$, respectively. The $(n-1) \times n$ parity-check matrix of the code is given by

where we left the zero entries blank. Then, we define the repetition code length-1 chain complex

$$\mathcal{R}: R_1 \xrightarrow{H_R} R_0, \tag{2.57}$$

where $R_1 \cong \mathbb{F}_2^n$ represent the *n* bits and $R_0 \cong \mathbb{F}_2^{n-1}$ represents the n-1 checks. Recall that with H_R we can associate a bipartite Tanner graph. In this case, this is the line graph, starting and ending in a variable node. From a graphical perspective, the hypergraph product can be viewed as the Cartesian product of the Tanner graphs of H_A and H_B . There will be four types of nodes: the product of a bit node with another bit node, the product of a bit and a check, the product of a check and a bit, and the product of two checks. Visually, we represent the nodes of the different products as

$$v_A \times v_B : \mathbf{O} \times \mathbf{O} \to \blacksquare \qquad v_A \times c_B : \mathbf{O} \times \Box \to \textcircled{O}$$
$$c_A \times c_B : \Box \times \Box \to \blacksquare \qquad c_A \times v_B : \Box \times \mathbf{O} \to \textcircled{O}$$

where we have chosen the coloring and shapes according to the classification $V_A \times V_B \cong C_2$, $V_A \times C_B \otimes C_A \times V_B \cong C_1$, and $C_A \times C_B \cong C_0$. Recall that by Eq. (2.52), C_2 is the space of Z-syndromes, C_1 is the space of qubits, and C_0 is the space of X-syndromes. There is an edge between two nodes (a_1, a_2) and (b_1, b_2) in $V_A \cup C_A \times V_B \cup C_B$ if either $(a_1, b_1) \in E_A$ and $a_2 = b_2$ or if $a_1 = b_1$ and $(a_2, b_2) \in E_B$. The product of two n = 3repetition codes, with one of them transposed, is shown in Figure 2.8, showing the Tanner graph of the surface code [129].



Figure 2.8: Hypergraph product of the n = 3 repetition code with its transpose yielding the surface code with rough edges on the sides and smooth edges on the top and bottom. The code has parameters [13, 1, 3]. Red and blue grid lines correspond to the edges in Z and X Tanner graphs of the check matrices of H_Z and H_X , respectively.

2.6 Decoding of Quantum Error-Correcting Codes

The redundancy in (quantum) error-correcting codes yields protection against noise. However, this protection is usually not autonomous in the sense that some additional work is required to restore the corrupted code words to the code space. This task is carried out by a decoding algorithm, commonly referred to as the decoder. We will give a formal definition of the decoder later on, and for now, describe a decoder, informally, as an algorithm that takes into account the measured syndrome information, as well as information about the noise model, to determine a correction that returns the noisy code word to the code space. The problem that the decoder solves is known as the decoding problem. Unfortunately, solving the decoding problem exactly turns out to be an exceptionally hard problem in general — solving the decoding problem for stabilizer codes, which we will define more concretely later, is #P-complete [130], while the decoding problem is considered (under standard hardness assumptions) intractable in general, and one can typically only find an approximate solution. As a result, different decoding algorithms are typically characterized by their accuracy and speed trade-offs.

Access to highly accurate decoders is beneficial as it can lead to reduced hardware requirements, for example, lower overhead of additional qubits used for redundancy or less stringent requirements on the physical error rates present in the architecture. However, fault-tolerant quantum computing will require solving the decoding problem in *real time* and thus the decoder must operate at least as fast as the syndrome information is generated through measurements to prevent the backlog problem [132]. To achieve this, decoding algorithms with linear or almost-linear complexity in the size of the parity-check matrix are desirable. In practice, decoders will most likely be required to be implemented on specialized hardware such as FPGAs or ASICs to achieve the necessary speed, especially for qubit architectures that generate measurements with a high rate such as quantum computers based on superconducting circuits [133] or integrated photonics [134, 135], likely yielding one terabit of syndrome information per second [136]. However, fast decoders, especially all-purpose decoders, are also highly relevant for researchers to study the performance of quantum error correction protocols numerically. Estimating the performance of a stabilizer code under a realistic noise model typically requires millions of Monte Carlo samples to accurately estimate the probability of extremely rare events. Each sample requires solving the decoding problem for 10^4 to 10^5 or more distinct fault locations, that is, a parity-check matrix with equally many columns.

2.6.1 The decoding problem

We will now define the decoding problem and the condition of success. Quite generally, a decoder is a map $D : \{0,1\}^r \to \mathcal{P}_n$ that, given a list of measurement outcomes, returns a correction C and is successful in correcting a Pauli error E if $CE \in S$. Typically, in quantum error correction, the list of measurement outcomes is related to the syndrome $\sigma(E)$ of E such that one can alternatively write $D(\sigma(E)) = E/S$. In the following, we will phrase the problem through the binary symplectic representation, which allows one to reason about the implementation of decoding algorithms much less abstractly. In particular, we express the Pauli error E as $e = \Lambda \operatorname{Mat}_{\mathcal{B}}(E)$ where we include the symplectic matrix into the definition of e such that all products are ordinary products and not symplectic products. Thus, $e = (e_Z, e_X)$ where $e_X, e_Z \in \mathbb{F}_2^n$ are the Z and X components of the error, respectively, and are differently ordered in comparison to the ordinary representation of a Pauli string. The syndrome equation

$$\boldsymbol{s} = H\boldsymbol{e},\tag{2.58}$$

then defines the syndrome $\mathbf{s} \in \mathbb{F}_2^r$ as in the classical case of linear codes, see also Eq. (2.25), through the binary representation of the $r \times 2n$ parity-check matrix. As $H\mathbf{e}$ effectively computes the symplectic inner product of H and \mathbf{e} , the i^{th} entry of \mathbf{s} , s_i , determines if E commutes with the stabilizer generator S_i of the code.

In addition to the syndrome, the decoder receives information about the noise model, typically in the form of error probabilities $\boldsymbol{p} \in \mathbb{R}^{2n}$ where p_i is the probability that the i^{th} bit of \boldsymbol{e} is flipped. For an independent error model, the probability of an error \boldsymbol{e} for the prior \boldsymbol{p} is given by a product distribution

$$\Pr(\boldsymbol{e}) = \prod_{i=1}^{2n} (1-p_i)^{1-e_i} p_i^{e_i} = \prod_{i=1}^{2n} (1-p_i) \prod_{i=1}^{2n} \left(\frac{p_i}{1-p_i}\right)^{e_i}.$$
 (2.59)

The correction c returned by the decoder is valid if H(c + e) = 0, and is successful if the logical action of c and e is identical, that is, if Lc = Le. Here, L is the binary representation of the logical operators, usually written as

$$L = \begin{pmatrix} \ell_X \\ \ell_Z \end{pmatrix}, \tag{2.60}$$

where ℓ_X and ℓ_Z are given by Eq. (2.33) via the binary symplectic representation of $\{\overline{X}_j\}$ and $\{\overline{Z}_j\}$.

Circuit-level noise. In the related literature, the above definition of the decoding problem is commonly referred to as the case of *perfect* syndrome measurements, with a distinction made for the case of *noisy* syndrome measurements. Here, we avoid this distinction, as we believe it is unnecessary and potentially misleading. Rather than distinguishing at the level of the decoding problem, we argue that the distinction should be made at the level of the quantum error-correcting code and noise model. That is, the parity-check matrix H and the logical correlation matrix L should be derived from the quantum error-correcting (QEC) code in space-time [137–139], rather than from the instantaneous QEC code defined purely in space. In such a setting, the columns of $H \in \mathbb{F}_2^{|D| \times |F|}$ correspond to independent fault mechanisms F, while the rows correspond to so-called *detectors* D, which are distinct from the stabilizers of a static QEC code.² The entry H[j, i] of the parity-check matrix — also known as the *detector check matrix* — is 1 if the *i*th fault mechanism flips the parity of the *j*th detector. Similarly, $L \in \mathbb{F}_2^{|K| \times |F|}$ encodes the logical correlations K in space-time, with L[j, i] = 1 indicating that the *i*th fault mechanism flips the jth logical correlation.

Importantly, the space-time formulation typically leads to a significantly larger QEC code than its static counterpart. This increase in size stems primarily from the need to repeat syndrome measurements over time to maintain fault tolerance against measurement errors and the additional fault locations in the circuit. Each additional round of syndrome extraction introduces new fault mechanisms and new detectors into the space-time code, effectively adding an extra (discrete) temporal dimension to the structure of the code, see also Section 2.4.3 and Figure 2.7. Thus, a code in space-time can be understood as an ordinary stabilizer code at a single moment in time, but one that operates in a higher-dimensional space with a correspondingly increased number of checks and errors.

Therefore, in the following, we will not distinguish between the decoding problem arising from a code capacity noise model with perfect syndrome measurements and one due to the more realistic circuit-level noise model, see also Section 2.4. We will instead emphasize, if necessary, specific features that originate from the space-time formulation of the decoding problem. However, for most purposes of decoding, a code in space-time can be regarded as an ordinary, albeit larger, stabilizer code at a single moment in time. Accordingly, we will use the terms *error* and *fault*, as well as *check* and *detector*, interchangeably throughout the text.

We have now seen that the decoding problem in quantum error correction closely resembles that of classical linear binary codes. This analogy becomes even more evident when comparing the definitions of the code space via the encoding matrix, see Eq. (2.26) for classical linear codes and Eq. (2.36) for stabilizer codes. For classical linear codes, decoding succeeds only if the correction matches the error exactly, that is, $\boldsymbol{e} = \boldsymbol{c}$. This ensures that the logical content remains unchanged, as can be seen explicitly by noting that $\boldsymbol{e}V^{-1} = (\ell : \mathbf{0}_{n-k}) = \boldsymbol{c}V^{-1}$ where V is the encoding matrix as defined in Eq. (2.26).

²A detector is typically understood as a parity constraint on a set of measurement outcomes.

In contrast, stabilizer codes allow for a more relaxed decoding criterion. As discussed above, decoding is successful as long as Le = Lc, meaning that the logical prediction of the error and correction coincide. The vectors e and c may still differ by an element of the stabilizer group without affecting the encoded information. This becomes particularly transparent when decomposing the Pauli operators E and C into their logical, destabilizer, and stabilizer parts, see Eq. (2.35). Specifically, we write $E = T_E L_E S_E$ and $C = T_C L_C S_C$. Successful decoding requires $L_C = L_E$, while $T_C = T_E$ holds by construction for a valid correction. Any residual difference between S_C and S_E corresponds to stabilizer elements, which act trivially on the code space. This logic is also reflected in the action of the inverse encoding unitary. After applying V^{\dagger} to the noisy encoded state $E |\psi\rangle$, we find:

$$V^{\dagger}E |\psi\rangle = V^{\dagger}T_E L_E S_E V(|\varphi\rangle \otimes |0_{n-k}\rangle)$$
(2.61)

$$= \ell_E \left| \varphi \right\rangle \otimes \bigotimes_{j=k+1}^n X_j^{s_{j-k}} Z_j \left| 0_{n-k} \right\rangle \tag{2.62}$$

$$= \ell_E |\varphi\rangle \otimes |s_1, \dots, s_{n-k}\rangle, \qquad (2.63)$$

where $\ell_E \in \mathcal{P}_k$ is the logical operator obtained from $V^{\dagger}L_E V$, compare also Eq. (2.37). This expression shows that the logical operator ℓ_E acts on the logical state $|\varphi\rangle \in \mathbb{C}_2^{2^k}$, while the stabilizer component determines the syndrome information that is encoded into the state of n - k additional qubits used for redundancy.

2.6.2 Maximum-likelihood decoding

As the decoding problem for stabilizer codes is not identical to that of classical linear codes, we must solve a more general task that corresponds to determining the most likely logical correction, that is, the maximization problem

$$\boldsymbol{\ell}_{C} = \operatorname*{arg\,max}_{\boldsymbol{\ell} \in \operatorname{im}(L)} \Big[\sum_{\boldsymbol{e} \in \mathbb{F}_{2}^{|F|} : L\boldsymbol{e} = \boldsymbol{\ell}, H\boldsymbol{e} = \boldsymbol{s}} \operatorname{Pr}(\boldsymbol{e}) \Big] = \operatorname*{arg\,max}_{\boldsymbol{\ell} \in \operatorname{im}(L)} \Big[\sum_{S \in \mathcal{S}} \operatorname{Pr}(T_{s}LS) \Big].$$
(2.64)

The obtained logical coset ℓ_C is, by construction, the most likely logical error Le, such that a decoder that evaluates Eq. (2.64) is called a maximum-likelihood decoder. Unfortunately, evaluating Eq. (2.64) is, in general, an extremely hard problem, belonging to the complexity class of #P-complete problems [130]. Intuitively, the hardness of (naively) evaluating Eq. (2.64) becomes apparent by recognizing that $\operatorname{im}(L)$ contains $2^{|K|}$ elements and that additionally exponentially many elements $e \in \mathbb{F}_2^{|F|}$ exist. Note that by decomposing the error as $E = T_s LS$, it is possible to reduce the number of elements in the sum to $2^{|S|}$, reducing the prefactor but retaining the exponential scaling in the size of the code. Some quantum error correction codes exhibit sufficient structure such that Eq. (2.64) can be efficiently evaluated [140], or allow approximate evaluations with high accuracy for modest system sizes [141–143]. Unfortunately, at the time of writing, it is unknown whether there exist classes of codes for which the maximum-likelihood decoding problem can be solved (approximately) efficiently under a realistic noise model. In other words, it remains unclear whether quantum codes in space-time exist that are efficiently (approximately) optimally decodable. As a result, the maximum-likelihood decoding problem is typically not solved directly, even approximately. Instead, the focus is often on solving the *minimum-weight* decoding problem.

2.6.3 Minimum-weight decoding

Instead of predicting the most likely logical error, as the maximum-likelihood decoder does, a *minimum-weight* decoder predicts the most probable *physical error*. The correction c returned by the minimum-weight decoder is obtained through

$$\boldsymbol{c} = \operatorname*{arg\,max}_{\boldsymbol{\tilde{c}} \in \mathbb{F}_{2}^{|F|}: H\boldsymbol{\tilde{c}} = \boldsymbol{s}} \Big[\Pr(\boldsymbol{\tilde{c}}) \Big], \tag{2.65}$$

which still is an NP-hard problem [131, 144, 145]. We note that for classical linear codes, this is the optimal solution.

The difference in accuracy between a maximum-likelihood decoder and a minimumweight decoder cannot be easily quantified and depends on various properties of the decoding problem. However, a common observation is that the difference increases if the stabilizer group contains many low-weight elements [146, 147], a common feature of qLDPC codes and quantum codes in space-time. As in the case of the maximumlikelihood decoding problem, there exist codes with sufficient structure such that the minimum-weight decoding problem is efficiently solvable.

In many cases, for Eq. (2.65) to be efficiently solvable, it is required that the parity-check matrix H is sparse, or more specifically, has low column and row weights. If additionally, H contains no short cycles when viewed as the biadjacency matrix of a bipartite graph, also known as the *decoding graph*, then iterative message-passing algorithms are efficient in approximately solving Eq. (2.65). We will describe this algorithm and the decoding graph in more detail below.

Before concluding this section, we note a particular relevant structure in H that allows the problem to be solvable efficiently. Given that the column weight of H is bounded by two, the minimum-weight perfect matching (MWPM) decoder [148] solves the minimum-weight decoding problem exactly in polynomial time [149]. Codes, for which the column weight condition is satisfied, are usually referred to as *matchable*. An important example of a matchable code is the two-dimensional toric code [94, 148], also known as the surface code when embedded into the plane. Surface codes are currently the gold standard for building a fault-tolerant quantum computer due to a variety of desirable properties, such as the possibility to realize them in a 2D architecture with only nearest-neighbor connectivity, a high error-correction threshold, and the existence of efficient minimum-weight decoders even under realistic circuit-level noise models [150]. Unfortunately, while scalable to achieve arbitrary distance d, surface codes are limited to encoding a single logical qubit k = 1. That is, when laid out as a two-dimensional grid of linear size L, the surface code is a quantum error-correcting code with parameters $[L^2, 1, L]^3$. The drawback that the surface code can only encode a single logical qubit is significant, as one requires roughly 1000 physical qubits for each additional surface code logical qubit [129] for a realistic physical error rate of $p = 10^{-3}$.

³To be precise, this is the *rotated* version of the surface code [151].

Lastly, in light of the discussion on circuit-level noise in Section 2.6.1, it is important to clarify that matchability is not a property of a code alone, but rather of the combination of a code and an error model. In particular, under the general depolarizing noise model introduced in Section 2.4.1, optimal decoding of the surface code is as hard as decoding an arbitrary stabilizer code [152]. This hardness arises because, although each qubit in the surface code is involved in only two X- and Z-type stabilizers, a Pauli Y error flips four syndrome bits, two X and two Z, making it incompatible with a matching-based decoder.

To faithfully represent all single-qubit Pauli errors as independent events, one constructs a parity-check matrix of the form

$$H = \begin{bmatrix} H_X & 0 & H_X \\ 0 & H_Z & H_Z \end{bmatrix},$$
(2.66)

where the columns correspond to X, Z, and Y errors, respectively. However, a common simplification is to represent Y errors as the combination of an X and a Z error, thereby eliminating the third column in Eq. (2.66). This renders the decoding problem matchable, but introduces a critical inaccuracy: it treats Y errors as two independent faults, misrepresenting their true physical probability O(p) as $O(p^2)$. Representing Y errors as combinations of X and Z errors is common practice, not solely to enable matchability, but also because it decouples the decoding into smaller, independent subproblems. Beyond that, it also alleviates issues in other decoding approaches, as we will see shortly.

2.6.4 Message-passing decoding

Unfortunately, implementing a minimum-weight decoder by exhaustively comparing the syndrome with all valid codewords is infeasible, especially for codes with large n. Instead, we describe here a decoding approach that dates back to Gallager [121] in the context of classical error-correcting codes. This algorithm is known under various names in different communities (see NOTES of chapter 2 in [108]), and we will usually refer to it as belief propagation (BP) or message-passing. More generally, the algorithm utilizes the generalized distributive law, see Ref. [153]. In the following, we often (implicitly) assume that the $r \times c$ parity-check matrix H is sparse and, more importantly, that both its row and column weights are small and constant. In other words, the Tanner graph associated with H has low-degree variable and check nodes. Tanner graphs are a crucial tool for understanding the algorithm outlined below.

The parity-check matrix H imposes rank(H) constraints on potential codewords $\mathbf{c} \in \mathbb{F}_2^c$. Each of these constraints can also be interpreted as defining a local subcode involving a single check and a subset of the codeword bits, specifically, those bits connected to that given check node. While such a subcode has distance d = 2, each bit typically participates in w_c such subcodes, where w_c is the column weight of H, and the global code can therefore have a significantly larger distance. This local structure is illustrated in Figure 2.9 from the perspective of a variable node v_b .

Within the subcode, what is the probability that the bit associated with variable node v_i is in error? More formally, we aim to evaluate the conditional probability that an error

 $e_i = 1$ occurred given the observed syndrome s, that is,

$$\Pr[e_i = 1 \mid \boldsymbol{s}]. \tag{2.67}$$

Note that only a subset of the syndrome bits in s are relevant to this probability. It turns out that the above conditional probability has a relatively simple expression when expressed as a likelihood ratio. Let $s_c \in \{0, 1\}$ denote the observed syndrome bit for check c. Then, following [121], we find

$$\frac{\Pr[e_i = 0 \mid \boldsymbol{s}]}{\Pr[e_i = 1 \mid \boldsymbol{s}]} = \frac{1 - p_i}{p_i} \prod_{c_j \in N(v_i)} \left[\frac{1 + (-1)^{s_{c_j}} \prod_{v_{i'} \in N(c) \setminus \{v_i\}} (1 - 2p_{i'})}{1 - (-1)^{s_{c_j}} \prod_{v_{i'} \in N(c) \setminus \{v_i\}} (1 - 2p_{i'})} \right].$$
(2.68)

Here, we have excluded the variable v_i from the product over $v \in N(c)$ since we are computing the conditional likelihood ratio with respect to e_i . Each term in the product corresponds to the contribution of a parity check c_j connected to v_i . The expression inside the product reflects the likelihood of an even or odd number of errors occurring among the neighboring variable nodes of c_j , excluding v_i . The factor $(-1)^{s_c}$ ensures that the check contributes with the correct sign depending on whether the observed syndrome bit is 0 or 1. To obtain this expression, we assume that the error on each bit is independent, and denote the corresponding vector of bitwise error probabilities by $\mathbf{p} = (p_1, \ldots, p_c)$. Each $p_i \in [0, 1]$ is the probability that the *i*th bit is in error, i.e., $e_i = 1$.

Equation (2.68) forms the basis of the message-passing decoding algorithm. To see this, first note that the expression can be rephrased in terms of likelihood ratios $\lambda = (1 - p)/p$, such that $1 - 2p = (\lambda - 1)/(\lambda + 1)$. Moreover, the right-hand side of Eq. (2.68) can be evaluated in parallel for all bits. However, since the equation only considers a local neighborhood in the Tanner graph, the resulting estimate is not the true marginal probability. Fortunately, it is possible to derive an iterative algorithm based on Eq. (2.68) that explores the whole graph.

This algorithm works by sending *messages* between check and variable nodes according to specific update rules. That is, for each iteration step, first, each check node c_j sends the message

$$\mu_{c_j \to v_i} = \prod_{v_{i'} \in N(c_j) \setminus \{v_i\}} \left[\mu_{v_{i'} \to c_j} \right]^{(1-2s_{c_j})},$$
(2.69)

to all its connected variable nodes $v_i \in N(c_i)$ and, in turn, each variable node v_i sends updated probabilities to its connected check nodes $c_j \in N(v_i)$ as the message

$$\mu_{v_i \to c_j} = \lambda_i \prod_{c_{j'} \in N(v_i) \setminus \{c_j\}} \mu_{c_{j'} \to v_i}, \qquad (2.70)$$

where $\lambda_i = (1 - p_{v_i})/p_{v_i}$ is the original likelihood of v_i being in error obtained from the channel probabilities p. The messages $\mu_{v_i \to c_j}$ can be understood as *a posteriori* probabilities for the *i*th bit to be in error given the constraints of the *j*th check, that is, the *j*th subcode. The algorithm is initialized by setting the messages $\mu_{v_i \to c_j}$ as the likelihoods λ_i . By inserting Eq. (2.69) into Eq. (2.70), one almost recovers Eq. (2.68) with the only



Figure 2.9: **a** Local subcodes or parity checks overlapping on the variable node v_i . **b** Message-passing during a single iteration from the perspective of a variable node v_i sending a message to check node c_j . **c** Message-passing during a single iteration from the perspective of a check node c_j sending a message to variable node v_i .

modification that the product is not taken over all checks. This assures that there is no immediate self-interference in the calculation of the i^{th} a posteriori probability. That is, information sent from a variable node to a check node will never flow back through the same edge from the check node to the variable node, and the independence assumption made to derive Eq. (2.68) remains valid, see also Figure 2.9. Thus, if this iteration process were continued until the local information has explored the whole graph, it would calculate the exact marginal probability of the i^{th} bit to be in error. After the iteration steps, the marginal probability can be obtained by multiplying all messages from neighboring check nodes, that is,

$$\mu_{v_i} = \lambda_i \prod_{c_{j'} \in N(v_i)} \mu_{c_{j'} \to v_i}.$$
(2.71)

However, this is only true if the independence assumptions remain valid, which requires the Tanner graph not to have cycles, that is, it must be a tree. For a graph with cycles of length g, also called the girth of the graph, the independence assumption holds for $\lfloor (g-2)/4 \rfloor$ iterations [154]. As already pointed out by Gallager, we can ignore the lack of independence and continue the iteration process which "is ultimately justified, of course, only by the fact that it works" [121]. It is also common to express the algorithm in the logarithmic domain by introducing the log-likelihood ratio $L(v_i) = \log((1-p_i)/p_i)$. In this formulation, the product in Eq. (2.70) becomes a sum, which is why the algorithm is also known as the sum-product algorithm. In this variant of the algorithm, one can make an approximation to reduce the cost of evaluating Eq. (2.69), which replaces the product with the min function, which leads to the min-sum algorithm, see, e.g., Ref. [155, 156].

A significant feature of the algorithm is its computational complexity. Without parallelization, the algorithmic complexity is linear in the number of nodes in the Tanner graph, the degree of the nodes, and the number of iterations. As noted above, the calculation of messages is completely local and can be straightforwardly parallelized.

Additionally, for a family of (quantum) LDPC codes, typically, the check and bit degrees are constant. Thus, a parallel implementation has at most a time complexity that is linear in the number of iterations. As the algorithm does not have any formal convergence guarantees in the presence of cycles in the graph, a stopping criterion is required. To this end, after each iteration step, one typically performs a hard-decoding $e_i = 0$ if $\mu_{v_i} < 0.5$ and $e_i = 1$ otherwise. If the current estimate of the error e satisfies the syndrome equation s = He, the algorithm is terminated. However, for quantum error-correcting codes subject to realistic noise, it is unclear whether the average number of iterations required is bounded or scales with the size of the code, see, e.g., Ref. [157, App. E].

Unfortunately, applying the BP algorithm to quantum error-correcting codes yields additional issues. One of those is related to the requirement of commutating stabilizers and is observed for many noise models. For stabilizers to commute in a CSS code, a X stabilizer S_X and a Z stabilizer S_Z must overlap on an even number of qubits, but at least two, say q_i and q_j . Now, a Pauli Y error on qubit q_i will anti-commute with both stabilizers S_X and S_Z and if we represent it as an independent faults mechanism in the parity-check matrix, this leads to a cycle of length-4 ($S_Z, Y_{q_i}, S_X, Y_{q_j}, S_Z$). A simple solution is to not represent Y errors as independent faults and represent them as a product of Pauli X and Z operators, and solve the decoding problems for X and Z faults separately. While doing this alleviates the problem of the existence of 4-cycles, it ignores correlations in the noise model that can aid the decoding performance in general, as it misrepresents the probability of a Y error as $O(p^2)$ instead of O(p). For practical considerations, one might accept this performance degradation if it is not too large for a decoder that has a linear complexity in the number of fault locations.

The second issue, however, is not as easily averted and is due to degeneracy. To reiterate, degeneracy refers to the fact that logical observables are only well defined up to elements of the stabilizer group S. Equivalently, there exist combinations of faults $\{t \mid t \in \ker H \cap \ker L\}$ that neither flip a detector nor a logical observable and therefore can be considered stabilizers. For a low-weight combination of faults t, it is possible for two corrections c and c+t to be both equally highly probable, such that BP does not converge to one of them. This situation is referred to as the *split belief* problem and occurs as the *a posteriori* probability distribution is not sufficiently peaked on a single configuration of faults. Forcing the algorithm to terminate after a finite number of iterations typically yields no correction that satisfies the syndrome, i.e., $H(c + c + t) = Ht = 0 \neq s$. It is no coincidence that we refer to such errors by the symbol t as an analogy of the issue that exists in the classical literature, where it is known as a trapping set, see, e.g., Refs. [158–162]. We will now describe a way to reduce the impact of this problem, unfortunately, at the cost of a significant increase in complexity.

2.6.5 Inversion decoding

The syndrome decoding problem, introduced around Eq. (2.25), s = He, is concerned with finding an estimate of e given the observed syndrome s. While so far we have been mostly focused on describing the maximum-likelihood decoding problem (Section 2.6.3) and the easier minimum-weight decoding problem (Section 2.6.3), we have seen that due to the existence of short cycles in the decoding graph, the standard approach of decoding (quantum) LDPC codes through BP may fail due to non-convergence, that is, the correction c does not reproduce the syndrome s. Naively, one might expect that it is simple and efficient to always obtain a correction that agrees with the syndrome by computing

$$\boldsymbol{c} = H^{-1}\boldsymbol{s},\tag{2.72}$$

which has cubic complexity in the size of the check matrix H due to the necessity to calculate the inverse H^{-1} , which can be achieved by Gaussian elimination. However, this approach does not work: a (left) inverse of H does not exist, since H lacks full column rank — a necessary condition for H to be the parity-check matrix of a nontrivial linear code. Nonetheless, the conceptual idea of Eq. (2.72) can be preserved by considering a more informed method.

Let *I* be a set of column indices constructed by selecting a linearly independent subset of the columns of the check matrix *H*. Accordingly, the size |I| of the index set is given by $|I| = \operatorname{rank}(H) \le r < c$ for the $r \times c$ matrix *H*. Then, $H_{[I]}$ is an invertible matrix formed by selecting the columns of *H* indexed by the set *I*. We can therefore compute a partial correction $\mathbf{c}' = H_{[I]}^{-1}\mathbf{s}$, which specifies the correction only on the coordinates indexed by *I*. To obtain a full correction vector $\mathbf{c} \in \mathbb{F}_2^c$, we define it component-wise as

$$c_j = \begin{cases} c'_j, \ \forall j \in I, \\ 0, \text{ otherwise.} \end{cases}$$
(2.73)

Since I indexes a linearly independent set of columns, each choice of I defines a unique partial solution c'. However, in general, randomly choosing the basis described by I is unlikely to result in a good correction c. In other words, it is unlikely that c corresponds to the minimum-weight correction as defined by Eq. (2.65). For instance, if I does not contain the index corresponding to a symbol in error, the correction cannot reproduce the error⁴, $c \neq e$.

After describing the conceptual ideas involved in inversion decoding, we now turn to the ordered statistics decoding (OSD) algorithm. Originally introduced by Fossorier and Lin [163], OSD serves as a post-processing step to improve upon invalid solutions produced by BP, helping to reduce error floors in classical LDPC codes. To this end, OSD leverages the marginal probabilities output by the BP decoder as soft information to guide the construction of the index set I, as we explain below. In 2019, Panteleev and Kalachev [164] were the first to apply OSD in the quantum setting, introducing BP+OSD as a surprisingly effective two-stage decoder of random qLDPC codes. Due to the generality of the (combined) decoding algorithm and its applicability even to codes in space-time, BP+OSD has become the gold standard for decoding general qLDPC codes.

We begin by describing the post-processing step, often referred to as OSD-0, which is invoked when the BP algorithm fails to converge within a maximum number of iterations. The OSD-0 algorithm is presented formally in Algorithm 1.

 $^{^{4}}$ We note that this is not necessarily an issue for stabilizer codes as the correction must only agree up to a stabilizer with the error.

\mathbf{A}	lgorithm	1	OSD-0	Post-Pro	cessing	Step
--------------	----------	----------	-------	----------	---------	------

- 1: **Input:** Soft information λ (from BP decoder)
- 2: Output: Correction c
- 3: Utilize the soft information vector λ to construct an ordered list of indices Π_{λ} (from most to least likely in error, MLE).
- 4: Rearrange the columns of the check matrix H according to the ordering Π_{λ} to obtain H'.
- 5: Construct the set of indices I as the first rank(H) linearly independent columns of H'
- 6: Calculate the OSD-0 solution based on the indices I by matrix inversion: $c_{[I]} = H_{[I]}^{-1}s$.
- 7: Obtain the correction across all bits as:

$$c' = (c_{[I]}, c_{[J]}) = (c_{[I]}, 0),$$

where $J = \overline{I}$ is the complement of I.

8: Reverse the column ordering induced by Π_{λ} to go from MLE ordering to the physical ordering, resulting in the final correction c.

There we see that the inversion decoding algorithm aims to solve the minimum-weight decoding problem (2.65) by selecting a basis that corresponds to the bits most likely in error, that is, the MLE basis. Indeed, if I corresponds to the rank(H) most likely bits according to the ordering Π_{λ} , the OSD-0 solution is the minimum-weight solution, that is, it is the optimal solution for the decoding problem of a linear code [165]. The attentive reader notices that the complexity of obtaining the optimal solution is polynomial, which hints that the OSD-0 solution constructed in this way will not always yield the optimal solution to the classical decoding problem. This issue arises when the |I| most likely error locations are not linearly independent, that is, they do not form a basis of the column space of H. In that case, the set I does not index the MLE basis and elements of the MLE basis are contained in I = J, the complement of I, such that J contains error locations that have larger error probabilities according to the soft information vector $\boldsymbol{\lambda}$ than some error locations contained in I. To find the optimal solution, one can systematically search through all error configurations in J that potentially provide a more likely estimate \tilde{c} . As this search space is exponentially large in the size of J, the computational cost of finding the optimal solution becomes prohibitively large for all but a few smaller codes. Thus, in practice, only configurations with a Hamming weight up to w are considered, known as order-w reprocessing. We refer the interested reader to Refs. [163–166] for a more complete discussion of higher-order reprocessing and briefly discuss the idea here.

In ordered statistics decoding, higher-order reprocessing refers to considering solutions for which $c_{[J]} \neq 0$. Given the OSD-0 solution $c_{[I]}$, the higher-order OSD solution for a particular choice of $c_{[J]}$ is given by

$$\boldsymbol{c}' = \left(\boldsymbol{c}_{[I]} + H_{[I]}^{-1} H_{[J]} \boldsymbol{c}_{[J]}, \boldsymbol{c}_{[J]}\right), \qquad (2.74)$$

which fulfills the syndrome equation for arbitrary $c_{[J]}$. Completing the reprocessing routine of order w then involves systematically searching through the set of admissible $c_{[J]}$

of Hamming weight w and retaining the solution c' of lowest total weight. It is possible to choose from various search strategies that explore different admissible subspaces for $c_{[J]}$ at finite w, see, e.g., Refs [163, 166].

2.7 Conclusion and Outlook

In this chapter, we have described most of the fundamentals required for fault-tolerant quantum computing. Rather than providing a comprehensive account of all aspects of quantum computing and quantum error correction, we have focused on the concepts most crucial for understanding the contributions of this thesis.

To this end, we have revisited classical error-correcting codes, in particular, linear block codes, to gain additional intuition about the structure of stabilizer codes. Stabilizer codes, especially Calderbank-Shor-Steane (CSS) codes, form the backbone of most practical quantum error correction strategies today. The correspondence between CSS codes and chain complexes, the primary elements of study in homology theory, is an additional reason for the importance of CSS codes. While homology initially provided a language that explained already known properties of codes, it has since proven to be tremendously productive, yielding various quantum low-density parity-check (qLDPC) codes with good properties. The appended work on *fault complexes*, see **Paper IX**, aims to extend the power of this framework from (CSS) codes at an instantaneous moment in time, to codes in space-time, or more generally, fault-tolerant protocols. It intends to serve as a unifying umbrella that incorporates multiple previously distinct viewpoints, that of gate-based and measurement-based computing, while making the full suite of mathematical tools from homology theory available to analyze these systems.

Regardless of the introduction of abstract structures such as chain complexes, this chapter aimed at introducing quantum error correction not merely from a theoretical viewpoint, but as a practical tool for scalable devices. This includes the questions of deriving low-complexity decoding algorithms that are capable of achieving the throughput and low-latency requirements that come with real-time quantum error correction. **Paper VII** introduces the *localized statistics decoding* algorithm, the first parallel decoding algorithm for general quantum error correction protocols that matches the performance of the current state-of-the-art belief propagation plus ordered statistics decoding algorithm whilst being substantially faster. Central to this is a novel linear algebra routine for parallel matrix factorization, which can efficiently solve sparse linear systems that we call *on-the-fly elimination*.

While low-complexity decoders will be necessary for fault-tolerant quantum computing, reducing the size of the decoding problem alleviates the issue. A promising path to achieve this is by considering quantum error-correcting codes that can be accurately decoded over a small (constant) window of time, or equivalently, a small window of syndrome measurement cycles. Such codes are also known as single-shot (decodable) quantum codes [106, 167]. Quantum radial codes, introduced in **Paper VIII**, are a family of qLDPC codes derived from the lifted product of classical quasi-cyclic codes. Numerical simulations suggest that these codes are single-shot decodable even under circuit-level noise, while additionally showing comparable error suppression to surface

codes of similar distance while using approximately five times fewer physical qubits. Overall, their error correction capabilities, tunable parameters, and small size make them promising candidates for implementation on near-term quantum processors.

It is worth emphasizing that this chapter has mostly omitted the discussion of a fundamental aspect of fault-tolerant quantum computing. That is, fault-tolerant quantum computing is not only about quantum memories but also about enabling reliable computation on encoded information. The question of fault-tolerant logical operations is largely beyond the scope of this thesis and is therefore omitted here. However, outside of this thesis, fault-tolerant operations for qLDPC codes are an extremely active topic of research. One illustrative approach to fault-tolerant logical operations is surface code lattice surgery [104], where logical operations are realized by merging and splitting encoded patches through local measurements. Lattice surgery can be naturally interpreted within the framework of measurement-based quantum computation and connects closely to the idea of Pauli-based computation [103], where computation proceeds via sequences of multi-qubit Pauli measurements. Generalizing lattice surgery to qLDPC codes poses unique challenges because these codes typically encode multiple logical qubits into a single block. As a result, addressing individual qubits becomes generally challenging. We point the interested reader to Ref. [36, Sec. 3] for an overview of the current developments around logical operations on qLDPC codes.

Finally, the methods discussed so far inevitably lead to large resource overheads. While stabilizer codes offer a powerful framework for protecting quantum information, their practical implementation demands substantial redundancy in physical qubits and operations. This motivates the exploration of alternative approaches. In the next chapter, we will shift focus to bosonic quantum error correction, where the structure of continuousvariable systems may offer a path toward reducing the overheads that plague traditional qubit-based schemes. By leveraging the rich physics of bosonic modes, new strategies for fault-tolerant quantum computing become accessible, expanding the toolkit available for future devices.

CHAPTER 3

Quantum Continuous Variables

Up to this point, we have focused on finite-dimensional quantum systems. By combining multiple two-level systems, we expanded the Hilbert space, either to enhance computational capabilities or to introduce redundancy for error correction. However, not all physical systems are inherently two-dimensional — or even finite-dimensional. Many naturally occurring quantum systems are described by infinite-dimensional Hilbert spaces. A prominent example are continuous-variable (CV) systems, where the expectation values of observables such as position and momentum take values over a continuum.

While a mathematically rigorous treatment of continuous-variable quantum mechanics lies beyond the scope of this chapter, we will introduce the essential concepts required to understand their use in quantum information. We will not concern ourselves with subtleties such as unphysical infinite-energy states, instead focusing on the tools and intuition needed for practical applications. This introduction aims to provide readers, particularly those familiar with discrete-variable systems, with the prerequisites for understanding the results presented in the appended papers.

We will consider the case of a single degree of freedom, also known as a single mode, for this presentation and note that the generalization to a finite number of modes is straightforward.

3.1 Fundamentals of Quantum Continuous Variables

In continuous-variable systems of a single degree of freedom, there exists a pair of selfadjoint operators, \hat{q}^1 and \hat{p} , that we refer to in the following as position and momentum, respectively, that satisfy

$$[\hat{q}, \hat{p}] = i\hbar \mathbb{1},\tag{3.1}$$

¹In the literature one will also often encounter the symbol x or X for this operator. Here we use q to distinguish it clearly from the Pauli X operator.

where in the following we will work in natural units $\hbar = 1$ and make the (infinitedimensional) identity operator $\mathbb{1}$ implicit. The above equation is known as the *canonical commutation relation* (CCR) and the pair of operators \hat{q} and \hat{p} as *canonical* operators, accordingly. The naming terminology originates from classical Hamiltonian dynamics, where the commutator in Eq. (3.1) is replaced by the Poisson bracket. Indeed, the above commutation relations typically arise when quantizing a simple phase space. Typically, one aims to find a representation for operators through matrices; however, in this case, no representation with *finite* dimensional matrices exists. Informally, this can be seen by taking the trace of Eq. (3.1). The cyclic property of the trace leads to $0 = i \operatorname{Tr} \mathbb{1}$, which is undoubtedly incorrect. Therefore, mathematically speaking, the position and momentum operators \hat{q} and \hat{p} are unbounded, and hence they cannot be trace-class, finite-rank, or represented on a finite-dimensional Hilbert space \mathcal{H} . Instead, one can define representations of these operators on the Hilbert space of square-integrable functions over the real line, $\mathcal{H} = L^2(\mathbb{R})$, such that

$$(\hat{q}\psi)(q) = q\psi(q), \quad (\hat{p}\psi)(q) = -i\frac{d}{dq}\psi(q), \quad \forall \psi \in L^2(\mathbb{R}).$$
 (3.2)

While the eigenstates of \hat{q} and \hat{p} do not belong to $L^2(\mathbb{R})$, we adopt the standard formalism in which we work with their quasi-eigenstates $|q\rangle$ and $|p\rangle$, defined by

$$\hat{q} |q\rangle = q |q\rangle, \quad \hat{p} |p\rangle = p |p\rangle, \quad q, p \in \mathbb{R},$$
(3.3)

which, although not normalizable, are conceptually useful for analytical computations. Note that their eigenvalues form a continuous set on the real line, constituting the basis of the terminology of *quantum continuous variables*. Position and momentum eigenstates also form a basis in the generalized sense, allowing any quantum state to be represented as a superposition of these basis elements

$$|\psi\rangle = \int_{-\infty}^{\infty} \psi(a) |a\rangle \, \mathrm{d}a,$$
 (3.4)

where $|a\rangle$ denotes the eigenstate of either \hat{q} or \hat{p} with eigenvalue a, and the complex-valued function $\psi(a) \in L^2(\mathbb{R})$ is called the wavefunction in the corresponding representation.

3.2 The Wigner function

While either the position or momentum representation completely describes the quantum state, it is still useful to have a representation that includes both. This is somewhat analogous to the description of a classical system within phase space. For example, the position and momentum of a single particle along one dimension can be represented by a point in a two-dimensional phase space, one axis representing position and the other momentum. Uncertainties in the knowledge of the position or momentum of that particle can be represented if we replace the point with a probability distribution that indicates the relative likelihood of encountering the particle with a particular combination of position and momentum.

In quantum mechanics, position and momentum are fundamentally linked by the uncertainty principle. As a result, constructing a phase-space representation of a quantum state is less straightforward than in classical mechanics. Nevertheless, several useful functions have been developed to describe quantum states in phase space [168–170]. Possibly one of the most prominent of those is known as the *Wigner function*, due to Wigner and Szilard [171], and is defined in terms of the density operator $\hat{\rho}$ by

$$W(q,p) = \frac{1}{\pi} \int_{-\infty}^{\infty} e^{i2py} \langle q+y|\hat{\rho}|q-y\rangle \,\mathrm{d}y\,.$$
(3.5)

This expression shows that the Wigner function is essentially the Fourier transform of the off-diagonal elements of $\hat{\rho}$ in the position basis. Just like the representation of a quantum state in the position or momentum basis, the Wigner function contains all the information about the quantum state². As Eq. (3.5) is invariant under complex conjugation by flipping the sign of y, the Wigner function is real-valued for all physical states. This makes the Wigner function representation a valuable tool to explain the behavior of continuous-variable quantum states intuitively. To this end, it is useful to note that the marginal distributions of the Wigner function recover the usual probability distributions, that is,

$$P(q) = \langle q|\hat{\rho}|q\rangle = \int_{-\infty}^{\infty} W(q,p) \,\mathrm{d}p, \quad P(p) = \langle p|\hat{\rho}|p\rangle = \int_{-\infty}^{\infty} W(q,p) \,\mathrm{d}q, \qquad (3.6)$$

and consequently, the Wigner function is normalized like an ordinary probability distribution,

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W(q, p) \,\mathrm{d}q \,\mathrm{d}p = 1.$$
(3.7)

However, the Wigner function is only a quasi-probability distribution, as it can take on negative values. These negativities are typically observed in Wigner functions of superpositions of distinct states, as we will see in more detail later.

Indeed, it is possible to use the Wigner function to visually distinguish different quantum states through their Wigner functions or to gain intuition about the expectation values of operators with respect to a quantum state. This is due to the fact that operator expectation values can be calculated as phase-space averages of their Wigner function representation weighted by the Wigner function of the corresponding quantum state, that is,

$$\operatorname{Tr}[\hat{A}\hat{\rho}] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W(q,p) W_A(q,p) \,\mathrm{d}q \,\mathrm{d}p, \qquad (3.8)$$

where \hat{A} is an arbitrary Hermitian operator and W_A denotes its phase space representation as defined by Eq. (3.5). For example, let \hat{A} represent another quantum state $\hat{\rho}_2 = |\psi_2\rangle\langle\psi_2|$.

²Indeed, all of quantum mechanics can be recast into quantum phase space.

For two states to be orthogonal, $|\langle \psi_1 | \psi_2 \rangle|^2 = 0$, this orthogonality condition translates to the requirement that the overlap integral in terms of their Wigner functions vanishes

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W_1(q, p) W_2(q, p) \,\mathrm{d}q \,\mathrm{d}p = 0.$$

This can occur in two ways. Either the Wigner functions have no overlapping support in phase space, or their product includes positive and negative contributions that exactly cancel out. In the latter case, regions where the Wigner function takes negative values play a crucial role, allowing for cancellation due to interference effects and symmetries in the states.

3.3 Quantum Harmonic Oscillators

A particularly relevant example where quantum continuous variables appear is during the quantization of the modes of an electromagnetic field. To this end, let us consider the electromagnetic field confined in a finite volume. The vector potential of this field obeys the wave equation, and its solution is a linear combination of frequency modes with a spatial profile determined by the boundary conditions. In terms of these modes, the Hamiltonian of the electromagnetic field reduces to that of a set of independent harmonic oscillators. Quantization of the electromagnetic field then reduces to the quantization of these individual modes.

The Hamiltonian that describes a single quantized mode of the electromagnetic field is given by

$$\hat{H} = \frac{1}{2} \left(\hat{p}^2 + \omega^2 \hat{q}^2 \right),$$
(3.9)

where ω is the frequency of the mode. The above Hamiltonian is also known as the *quantum harmonic oscillator* and it can be diagonalized by introducing the bosonic annihilation \hat{a} and creation \hat{a}^{\dagger} operators such that

$$\hat{a} = \frac{1}{\sqrt{2\omega}} \left(\omega \hat{q} + i \hat{p} \right), \quad \hat{a}^{\dagger} = \frac{1}{\sqrt{2\omega}} \left(\omega \hat{q} - i \hat{p} \right), \tag{3.10}$$

that fulfill the bosonic canonical commutation relations $[\hat{a}, \hat{a}^{\dagger}] = \mathbb{1}$. In the following we will always work with the dimensionless version of the operators \hat{q} and \hat{p} , that is, we redefine $\hat{q} \to \sqrt{\omega}\hat{q}$ and $\hat{p} \to \hat{p}/\sqrt{\omega}$ such that $\hat{q} = (\hat{a} + \hat{a}^{\dagger})/\sqrt{2}$ and $\hat{p} = i(\hat{a}^{\dagger} - \hat{a})/\sqrt{2}$. We will also sometimes refer to \hat{q} and \hat{p} as field quadratures, as they yield a Hamiltonian that is a sum of squares.

Expressed in terms of the annihilation and creation operators, the quantum harmonic oscillator takes the form

$$\hat{H} = \omega \left(\hat{a}^{\dagger} \hat{a} + \frac{1}{2} \right), \qquad (3.11)$$

but we will typically neglect the constant energy shift. Surprisingly, even though the field quadratures represent continuous variables, the energy spectrum of the quantum

harmonic oscillator is discrete, with possible energy levels $E_n = \omega(n + \frac{1}{2})$ with eigenstates $|n\rangle$ for all $n \in \mathbb{N}$. These states are called *Fock states* and form an orthonormal basis. Physically, we associate with $|n\rangle$ a state with n bosonic excitations, which in the case of the electromagnetic field we call photons. As the name implies, annihilation and creation operators subtract and add excitations, that is,

$$\hat{a}|n\rangle = \sqrt{n}|n-1\rangle, \quad \hat{a}^{\dagger}|n\rangle = \sqrt{n+1}|n+1\rangle, \quad \hat{a}^{\dagger}\hat{a}|n\rangle = \hat{n}|n\rangle = n|n\rangle, \quad (3.12)$$

where we introduced the number operator \hat{n} .

3.3.1 States of Quantum Harmonic Oscillators

To gain intuition about the state space of a single bosonic mode, we now turn to specific examples of quantum states and their behavior in phase space. We will use the Wigner function introduced in Section 3.2 to visualize some of those states.

The ground state of the quantum harmonic oscillator in Eq. (3.11) is called the vacuum state, corresponding to the state $|n = 0\rangle$ in the Fock basis. The Wigner function of the state corresponds to a rotation-symmetric Gaussian function in two dimensions that is centered at the origin of the phase space, see Figure 3.1a. The vacuum state saturates the Heisenberg uncertainty relations, that is, it minimizes the standard deviation in the position and momentum quadrature simultaneously. As expected from the quadratic potential of the harmonic oscillator, Fock states with non-zero excitation number n occupy a larger area in phase space and are not minimum uncertainty states. Their Wigner function remains rotational symmetric but now displays oscillations that are also apparent in the marginals, see Figure 3.1b and Figure 3.1c. These oscillations make those states highly non-classical.

A more classical-like family of states is formed by the *coherent states*, which are eigenstates of the annihilation operator

$$\hat{a} \left| \alpha \right\rangle = \alpha \left| \alpha \right\rangle.$$
 (3.13)

They are labeled by a complex amplitude α and can be generated from the vacuum via the displacement operator

$$|\alpha\rangle = \hat{D}(\alpha) |0\rangle, \quad \hat{D}(\alpha) = \exp\left(\alpha \hat{a}^{\dagger} - \alpha^* \hat{a}\right).$$
 (3.14)

Coherent states preserve the Gaussian shape of the vacuum in phase space but are displaced to the point $(q = \sqrt{2}\text{Re}(\alpha), p = \sqrt{2}\text{Im}(\alpha))$, see Figure 3.1d. They follow classical trajectories under free evolution and are often regarded as the "most classical" quantum states. Importantly, they also form an overcomplete basis for the Hilbert space, satisfying the resolution of the identity

$$\frac{1}{\pi} \int |\alpha\rangle\!\langle \alpha| \, \mathrm{d}^2 \alpha = \mathbb{1}. \tag{3.15}$$

A further generalization is provided by *squeezed states*, which also have Gaussian Wigner functions but redistribute uncertainty between the \hat{q} and \hat{p} quadratures. This



Figure 3.1: Wigner function W(q, p) and marginals of various states of the quantum harmonic oscillator. **a** The vacuum state $|0\rangle$, **b** Fock $|1\rangle$, **c** Fock $|5\rangle$, **d** Coherent ($\alpha = 1+1i$), **e** Squeezed (-10 dB), **f** Squeezed displaced (10 dB, $\alpha = -2$).

squeezing can reduce noise below the vacuum level in one quadrature, at the expense of increased fluctuations in the conjugate variable, while still respecting the uncertainty principle. The resulting Wigner function becomes elliptical rather than circular, see Figure 3.1e. Squeezed vacuum states are generated by applying a squeezing operator to the vacuum, as we will discuss in Section 3.5. More generally, *squeezed coherent states* combine displacement and squeezing to yield squeezed states centered away from the origin, see Figure 3.1f, representing finite-energy approximations of position or momentum states dependent upon the squeezing direction.

3.4 Noise Channels

If we want to use quantum continuous-variable systems for quantum information processing, we need to interact with the system in order to manipulate and control it. However, if we are able to interact with the quantum system in a controllable way, then there must also be an environment that is interacting in an uncontrolled way with the system. This uncontrolled interaction with an environment causes decoherence, as is does for qubit systems. However, for continuous-variable quantum systems, due to their infinite-dimensional Hilbert space, there is a plethora of relevant decoherence mechanisms. Under certain, typically fulfilled, assumptions, noise can be modeled through the Gorini-Kossakowski-Sudarshan-Lindblad master equation [172, 173] given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\rho} = \mathcal{L}\hat{\rho} = -i[\hat{H},\hat{\rho}] + \sum_{j}\kappa_{j}\mathcal{D}[\hat{L}_{j}]\hat{\rho},\qquad(3.16)$$

where \mathcal{L} is known as the Lindbladian, and $\mathcal{D}[\hat{L}_j]\rho$ is the Lindblad dissipator acting on $\hat{\rho}$, that is $\mathcal{D}[\hat{L}_j]$ is a superoperator, given by

$$\mathcal{D}[\hat{L}_{j}]\hat{\rho} = \hat{L}_{j}\hat{\rho}\hat{L}_{j}^{\dagger} - \frac{1}{2}\hat{L}_{j}^{\dagger}\hat{L}_{j}\hat{\rho} - \frac{1}{2}\hat{\rho}\hat{L}_{j}^{\dagger}\hat{L}_{j}.$$
(3.17)

The operators \hat{L}_j are called jump operators which depend on the type of noise and $\kappa_j \geq 0$ are dissipation rates that characterize the strength of the noise. Without loss of generality, we will typically assume that $\hat{H} = 0$ which can be achieved by an appropriate reference frame transformation. In this case, we can also represent the noise channel written in the Kraus representation [174–178],

$$\hat{\rho}(t) = \sum_{k} \hat{K}_{k}(t)\hat{\rho}(0)\hat{K}_{k}^{\dagger}(t), \quad \text{with } \sum_{k} \hat{K}_{k}^{\dagger}(t)\hat{K}_{k}(t) = \mathbb{1},$$
(3.18)

which is obtained by formally integrating Eq. (3.16) up to time t, with $\hat{K}_k(t)$ the so-called Kraus operators.

In the following, we summarize some of the most common continuous-variable noise channels and describe intuitively their effect on quantum states. We will usually not comment on their physical origin as this can depend on the physical system we are trying to model. See also Refs. [179–181] for an in-depth introduction.

Photon Loss. One of the dominant noise channels in optics as well as superconducting circuits, is the loss of excitations, often referred to as photon loss in those architectures. Losses are represented in the master equation by the Lindblad jump operator,

$$\hat{L}^{(\text{loss})} = \hat{a}, \tag{3.19}$$

or within the Kraus representation, by the Kraus operator

$$\hat{K}_{k}^{(\text{loss})} = \frac{(1 - e^{-\kappa t})^{k/2}}{\sqrt{k!}} e^{-\kappa \hat{n}t/2} \hat{a}^{k}.$$
(3.20)

For the loss channel, the Kraus operator $\hat{K}_k^{(\text{loss})}$ describe the time evolution generated by the exact loss of k photons, and $1 - \exp(-\kappa t)$ is the probability for losing a single photon. Additionally to the factor \hat{a}^k that maps Fock state $|n\rangle$ to Fock state $|n-k\rangle$, the other contribution is the Fock damping operator $e^{-\kappa \hat{n}t/2}$ that leaves Fock states invariant. Its effect is more readily observed by considering the Heisenberg evaluation it generates. Using the Baker-Campbell-Hausdorff (BCH) formula, one finds that $\hat{a} \rightarrow \hat{a}e^{-\kappa t/2}$, contracting phase space towards the center, representing a continuous loss of energy from the system to the environment.

Photon Number Dephasing. While photon number dephasing is typically considered small in superconducting architectures, it becomes important if the system is nonlinear or is coupled to a nonlinear system, such as a two-level system. The Lindblad jump operator is given by

$$\hat{L}^{(\text{dephasing})} = \hat{a}^{\dagger} \hat{a} = \hat{n} \tag{3.21}$$

and the channel can equivalently be represented through the Kraus operators

$$\hat{K}_{k}^{(\text{dephasing})} = \frac{(\kappa_{\phi}t)^{k/2}}{\sqrt{k!}} e^{-\kappa_{\phi}\hat{n}^{2}t/2} \hat{n}^{k}.$$
(3.22)

Notice that in this case the exponential is quadratic in \hat{n} such that the BCH formula does not yield a closed-form expression as above. While the discrete Kraus representation is sufficient to realize that the channel is energy-preserving, that is, it commutes with the harmonic oscillator Hamiltonian, additional intuition can be derived from the continuous Kraus representation of the channel given by

$$e^{\kappa_{\phi}\mathcal{D}[\hat{n}]t}\hat{\rho} = \int_{-\infty}^{\infty} p_{\kappa_{\phi}t}(\theta)e^{-i\theta\hat{n}}\hat{\rho}e^{i\theta\hat{n}}\,\mathrm{d}\theta\,,\quad\text{with }p_{\kappa_{\phi}t}(\theta) = \frac{1}{\sqrt{2\pi\kappa_{\phi}t}}\exp\left(-\frac{\theta^2}{2\kappa_{\phi}t}\right) (3.23)$$

from which dephasing can be interpreted as causing random rotations of the state with respect to an underlying Gaussian distribution of variance $\sigma^2 = \kappa_{\phi} t$.

Photon Gain. Photon gain can be considered the opposite of photon loss and is accordingly represented by the Lindblad jump operator

$$\hat{L}^{(\text{gain})} = \hat{a}^{\dagger}, \qquad (3.24)$$

and Kraus operators,

$$\hat{K}_{k}^{(\text{gain})} = \frac{(e^{\kappa t} - 1)^{k/2}}{\sqrt{k!}} e^{-\kappa \hat{n}t/2} \left(\hat{a}^{\dagger}\right)^{k}.$$
(3.25)

Using the BCH formula, one finds that $\hat{a} \to \hat{a}e^{\kappa t/2}$, expanding phase space. Typically, photon gain does not appear on its own and instead appears together with photon losses.

Thermal Noise. Thermal noise originates from the coupling of a system to a finite temperature environment and corresponds to a combination of loss and gain, requiring two Lindblad jump operators

$$\hat{L}_{-} = \sqrt{n_{\rm th} + 1}\hat{a}, \quad \hat{L}_{+} = \sqrt{n_{\rm th}}\hat{a}^{\dagger}, \quad \text{with } n_{\rm th} = \frac{1}{\exp(\hbar\omega/k_B T) - 1}, \qquad (3.26)$$

where we introduced the average thermal occupation number $n_{\rm th}$ and both jumps happen with rate κ . The loss channel is recovered in the limit of $T \to 0$. Since creation and annihilation operators, together with the identity, form a closed algebra, it is possible to express the Kraus operators of this channel as a product of the Kraus operators for loss and gain, see Ref. [174] for details.

Gaussian displacements. The Gaussian displacement channel has Lindblad jump operators

$$\hat{L}_1 = \hat{a}, \quad \hat{L}_2 = \hat{a}^{\dagger}, \tag{3.27}$$

and thus is recovered from the thermal noise channel in the infinite temperature limit $T \to \infty$ and zero coupling limit $\kappa \to 0$ with $n_{\rm th}\kappa t/2 = \sigma^2$. Here, we introduced the variance σ^2 that appears in the continuous Kraus representation of the Gaussian displacement channel given by

$$\hat{\rho} \to \frac{1}{2\pi\sigma^2} \int e^{-|\alpha|^2/2\sigma^2} \hat{D}(\alpha) \hat{\rho} \hat{D}^{\dagger}(\alpha) \,\mathrm{d}^2\alpha, \qquad (3.28)$$

giving the channel its name. A somewhat uncommon way to write this channel is in the following form [60]

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\rho} = -\frac{D}{2}\left[\hat{p}, [\hat{p}, \hat{\rho}]\right] - \frac{D}{2}\left[\hat{q}, [\hat{q}, \hat{\rho}]\right], \qquad (3.29)$$

where $D = \kappa/2$ plays the role of a diffusion constant. As a result, the channel be interpreted as broadening features of the Wigner function and thus washing out rapidly oscillating parts of the phase space distribution. We note that this channel is somewhat of a quantum analog of the additive white Gaussian noise channel discussed in Section 2.3.1.

3.5 Universal Gates

A universal quantum computation generally consists of three steps: preparing an initial computational state, applying a sequence of universal operations, and finally measuring the system's state. In this section, we define the notion of universality for CV quantum computing and introduce a set of operations that enable universal quantum computation in this framework.

We have seen earlier that the concept of universality in discrete-variable (DV) quantum computing is well established: a finite set of single- and two-qubit gates can approximate any unitary operation to arbitrary accuracy. In the CV setting, it was long believed that this notion does not directly carry over. The core difficulty lies in the fact that unitary transformations on an infinite-dimensional Hilbert space are generally characterized by infinitely many parameters and thus cannot be approximated using a finite set of elementary operations [182]. However, recent results [183] have shown that any *physical* single-mode unitary operation can, in fact, be approximated by a finite-degree polynomial $P(\hat{q}, \hat{p})$ in the quadrature operators, likely generalizable to the multimode case. This insight opens the door to defining a meaningful notion of universality in CV quantum computation based on polynomial Hamiltonians.

Therefore, one is interested in being able to synthesize arbitrary polynomials in the bosonic creation and annihilation operators. To achieve this, one does not need to be able to implement any monomial $\hat{a}^{\dagger m} \hat{a}^{n}$, but instead can synthesize them from a few generating Hamiltonians $\{\hat{H}_l\}$. For that note, the two identities following from the Baker-Campbell-Hausdorff formula [184]

$$e^{-i\hat{A}\delta t}e^{-i\hat{B}\delta t}e^{i\hat{A}\delta t}e^{i\hat{B}\delta t} = e^{[\hat{A},\hat{B}]\delta t^2} + O\left(\delta t^3\right),\tag{3.30}$$

$$e^{i\hat{A}\delta t/2}e^{i\hat{B}\delta t/2}e^{i\hat{B}\delta t/2}e^{i\hat{A}\delta t/2} = e^{i(\hat{A}+\hat{B})\delta t} + O\left(\delta t^{3}\right),$$
(3.31)

which are valid for two operators \hat{A} and \hat{B} . A necessary condition for the generating set $\{\hat{H}_l\}$ is that it does not form a closed algebra with respect to the commutator. One can easily convince oneself that this requires at least one of the generators $\{\hat{H}_l\}$ to be higher than quadratic order in the bosonic operators.

In the following, we describe a set of operations that is sufficient for the above notion of universality in CV quantum computing, see also Refs. [182, 184, 185].

Rotation. Arbitrary rotations in phase-space by an angle θ are obtained by the unitary

$$\hat{R}(\theta) = e^{-i\theta \hat{a}^{\dagger}\hat{a}} = e^{-i\theta(\hat{q}^2 + \hat{p}^2)/2}.$$
 (3.32)

The gate transforms the quadrature operators \hat{q} and \hat{p} as $\hat{q} \to \cos(\theta)\hat{q} + \sin(\theta)\hat{p}$ and $\hat{p} \to \cos(\theta)\hat{p} - \sin(\theta)\hat{q}$. For the choice $\theta = \pi/2$ the gate is termed the Fourier transform $\hat{F} = \hat{R}(\theta)$ since it takes the quadrature operators to its conjugate. We remark that the Fourier transform \hat{F} is the continuous-variable version of the Hadamard gate H.

Displacement. We have already encountered the displacement operator as the operator that creates coherent states. The displacement operator $\hat{D}(\alpha) = \exp(\alpha \hat{a}^{\dagger} - \alpha^* \hat{a})$ transforms the bosonic annihilation operator as $\hat{D}(\alpha)\hat{a}\hat{D}^{\dagger}(\alpha) = \hat{a} + \alpha$.

Squeezing. The squeezing operation with real squeezing factor s is given by

$$\hat{S}(s) = e^{-i\log(s)(\hat{q}\hat{p} + \hat{p}\hat{q})/2}.$$
(3.33)

This corresponds to an operation with $\frac{10}{\log(10)}\log(s^2)$ dB of squeezing. The action of $\hat{S}(s)$ on the quadrature operators is given by $\hat{q} \to s\hat{q}$ and $\hat{p} \to s^{-1}\hat{p}$, i.e., the operator squeezes one quadrature and stretches the conjugate one. The operator is also commonly expressed in terms of creation and annihilation operators as

$$\hat{S}(\xi) = e^{\frac{\xi^*}{2}\hat{a}^2 - \frac{\xi}{2}\hat{a}^{\dagger 2}}, \qquad (3.34)$$

with complex squeezing parameter $\xi = re^{i\theta}$. The squeezing operator implements a Bogoliubov transformation given by $\hat{a} \rightarrow \hat{a} \cosh r - e^{i\theta} \hat{a}^{\dagger} \sinh r$ from which the transformation Eq. (3.33) is obtained for $r = |\log s|$ and $\theta = \pi$ if s > 1 and $\theta = 0$ if s < 1. **Beam Splitter.** The beam splitter is usually given in terms of annihilation and creation operators of two modes a and b as

$$\hat{B}(\theta) = e^{\theta(\hat{a}\hat{b}^{\dagger} - \hat{a}^{\dagger}\hat{b})}, \qquad (3.35)$$

and we omit the representation in terms of quadrature operators. The action of the beam splitter on the annihilation operators is given by $\hat{a} \rightarrow \sqrt{\tau}\hat{a} + \sqrt{1-\tau}\hat{b}$ and $\hat{b} \rightarrow \sqrt{\tau}\hat{b} - \sqrt{1-\tau}\hat{a}$ where $\tau = \cos^2 \theta$ is known as the transmissivity of the beam splitter in quantum optics.

Cubic Phase Gate. The cubic phase gate is defined as [185–190],

$$\hat{\Gamma}(\gamma) = \exp(i\gamma\hat{q}^3), \qquad (3.36)$$

and is commonly used in quantum information protocols as the necessary nonlinear gate. The coefficient γ is known as the cubicity of the gate. Any non-zero value for the cubicity is in principle sufficient as it can be enhanced by a squeezing transformation, i.e,

$$\hat{\Gamma}(\gamma') = \hat{S}^{\dagger}(-r)\hat{\Gamma}(\gamma)\hat{S}(-r), \qquad (3.37)$$

with $\gamma' = \gamma e^{3r}$ and r the real squeezing parameter.

3.6 Bosonic Quantum Error Correction

Bosonic quantum error correction is a field in which one is interested in achieving robust quantum information processing by encoding quantum information into a subspace of the Hilbert space of one or multiple harmonic oscillators. The basis of this idea is that instead of using many qubits to provide the redundancy required to protect the encoded information, one can directly benefit from the vastness of the harmonic oscillator Hilbert space. While one can approach this field from an information-theoretic approach, here we are more interested in a practical approach that encodes information in a non-local way in phase space and in this way achieves protection against physical noise channels such as the ones previously described, e.g., photon loss, but also undesired nonlinearities that appear as natural consequences of the hardware platform used to realize the encoding.

While thinking about bosonic codes, it is important to consider that those are not simply generalizations of two-dimensional qubit systems to *d*-dimensional qudit systems in an appropriate limit. The reason that this is not the case is that while qudit Pauli operators are a natural object to study from the viewpoint of stabilizer theory, these are not naturally realized in physical realizations of bosonic systems. What is realized, though, are products of creation and annihilation operators, \hat{a}^{\dagger} and \hat{a} . Even though this seems to be a major complication, it necessitates thinking more closely about the physical constraints and leads to a tighter co-design of quantum systems and quantum error correction codes. Thinking outside the usual stabilizer theory framework leads to exploring things that have not been considered or are strictly forbidden by no-go theorems. Particular examples are the possibility of a bias-preserving CNOT gate [64] or even a continuous-parameter set of transversal gates [191], both of which are possible due to the continuous-variable nature of the system and cannot be achieved in finite-dimensional systems [64, 192].

In the following, we will focus on two prominent examples of bosonic quantum errorcorrecting codes encoded in a single harmonic oscillator. Even though such an encoding possibly accesses an infinite-dimensional Hilbert space, these bosonic codes do not have an error threshold [193] that allows for arbitrarily low logical error rates — in contrast to certain families of stabilizer codes. As a result, they are often used as inner codes concatenated with an outer stabilizer code to achieve fault tolerance. Furthermore, depending on the noise channel, implementing the recovery unitary to restore the noisy state to the code space can be highly complicated. Instead, one of the two alternative approaches is typically considered: engineering the dynamics of the system such that the noisy states get autonomously restored to the code space, or employing the measurementbased paradigm of quantum error correction such that physical errors cannot spread throughout the computation.

3.6.1 The cat code

The cat code, as we are thinking about it today, most likely originated from the work of Leghtas *et al.* [61, 62] that proposed the bosonic cat encoding designed to protect against single photon losses. That is, the computational basis states are defined as

$$|0\rangle_L = \frac{|\alpha\rangle + |-\alpha\rangle}{\sqrt{2(1+e^{-2|\alpha|^2})}} = \frac{1}{\cosh(\alpha^2)} \sum_{n=0}^{\infty} \frac{\alpha^{2n}}{\sqrt{(2n)!}} |2n\rangle, \qquad (3.38)$$

$$|1\rangle_{L} = \frac{|i\alpha\rangle + |-i\alpha\rangle}{\sqrt{2(1 + e^{-2|\alpha|^{2}})}} = \frac{1}{\cosh(\alpha^{2})} \sum_{n=0}^{\infty} \frac{(-1)^{n} \alpha^{2n}}{\sqrt{(2n)!}} |2n\rangle.$$
(3.39)

Notice that both computational basis states are supported on even Fock states only. Additionally, a single photon loss would take us from the subspace of even Fock states to the subspace of odd Fock states, the error space. While we cannot perform a measurement in the Fock basis without collapsing the state, it is possible to perform a syndrome measurement that only distinguishes even and odd Fock states. This measurement is known as a photon number parity measurement and allows one to infer whether a photon has been lost without collapsing the states. Intuitively, the photon number parity measurement is determined by the sign of the Wigner function at the origin of the phase space, see Figure 3.2.

It should be emphasized that it is not possible to correct the loss of two photons. Indeed, the operator \hat{a}^2 acts like a Pauli Z operator on the logical subspace, that is, $\hat{a}^2(|0\rangle_L \pm |1\rangle_L) \propto (|0\rangle_L \mp |1\rangle_L)$ as the coherent states are eigenstates to the annihilation operator. Using our intuition that the parity measurement is determined by the sign of the Wigner function at the origin, we can also convince ourselves that the cat code protects against dephasing errors. However, the dephasing rate should be small enough such that the computational states $|0\rangle_L$ and $|1\rangle_L$ are still distinguishable, as they are related by a $\pi/4$ rotation in phase space.

More recently, the above bosonic quantum error correction code is more commonly referred to as the four-component cat code, while the encoding of the "cat code" is more



Figure 3.2: Wigner functions W(q, p) and marginal distributions of the four-component cat code, see Eqs. (3.38) and (3.39), for the six different cardinal states given by, up to normalization, $\mathbf{a} |+Z\rangle = |0\rangle$, $\mathbf{b} |-Z\rangle = |1\rangle$, $\mathbf{c} |+X\rangle = |0\rangle + |1\rangle$, $\mathbf{d} |-X\rangle = |0\rangle - |1\rangle$, $\mathbf{e} |+Y\rangle = |0\rangle + i |1\rangle$, $\mathbf{f} |-Y\rangle = |0\rangle - i |1\rangle$. The amplitude of the coherent states is $\alpha = 2.5$.

likely to refer to a simpler encoding [194–196] defined via the dual basis states

$$\left|\pm\right\rangle_{\rm cat} = \frac{\left|\alpha\right\rangle \pm \left|-\alpha\right\rangle}{\sqrt{2(1\pm e^{-2\left|\alpha\right|^{2}})}},\tag{3.40}$$

$$|0\rangle_{\rm cat} = \frac{|+\rangle_{\rm cat} + |-\rangle_{\rm cat}}{\sqrt{2}} \approx |\alpha\rangle + O(e^{-2|\alpha|^2}), \tag{3.41}$$

$$|1\rangle_{\rm cat} = \frac{|+\rangle_{\rm cat} - |-\rangle_{\rm cat}}{\sqrt{2}} \approx |-\alpha\rangle + O(e^{-2|\alpha|^2}), \tag{3.42}$$

which is unable to correct a single photon loss, as in this case, the annihilation operator acts like a Pauli Z operator on the logical subspace. The Wigner functions of the six cardinal states of the encoding are shown in Figure 3.3. Instead, interest in this encoding is due to Mirrahimi *et al.* [197], Puri *et al.* [63], and related work which introduced a scheme that dynamically protects cat qubits against photon number dephasing errors, introducing

stabilized cat qubits. Here, we will sometimes refer to these ideas as the confinement of cat qubits to distinguish more clearly from stabilizer codes in the discrete-variable setting.

To describe the idea of confinement, note that the code space of a bosonic encoding is typically not stable under physical noise channels. Thus, any logical information will eventually leak outside of the code space. However, it is possible to stabilize the code space through appropriate confinement schemes. To this end, one aims to engineer effective interactions whose dynamics yield a degenerate ground state isomorphic to the code space. To achieve this, recall that the dynamics of a quantum system are either described by the Schrödinger equation, $\partial_t |\psi\rangle = -i\hat{H}(t) |\psi\rangle$, for the evolution of a closed system, or by the master equation, $\partial_t \hat{\rho} = \mathcal{L}(t)\hat{\rho}$, for an open system. If we can find \hat{H} or \mathcal{L} such that the code states are eigenstates to these (super)operators with vanishing eigenvalues, then the code space is a fixed point of the evolution. Also, the dissipative evolution generated by \mathcal{L} ensures that if leakage occurs, the state will relax back to the code space.

Let us make these ideas more explicit based on the example of the cat code. Notice that the coherent states $|\pm\alpha\rangle$ are eigenstates of the operator $\hat{F} = (\hat{a}^2 - \alpha^2)$ with eigenvalue zero such that, by linearity, $|\pm\rangle_{cat}$ are eigenstates of \hat{F} as well. Then, the dissipative evolution generated $\mathcal{D}[\hat{F}]$ and the Hamiltonian evolution generated by $\hat{H}_F = \frac{1}{2}\hat{F}^{\dagger}\hat{F}$ both continuously stabilize the code space [198]. From the Knill-Laflamme conditions [199], one finds that for single photon losses, stabilized cat codes allow for an arbitrary suppression of the logical bit-flip rate Γ_X , however, at the cost of an increasing phase-flip error rate Γ_Z , that is,

$$\Gamma_X \propto |\langle -\alpha | \hat{a} | \alpha \rangle|^2 = |\alpha|^2 e^{-2|\alpha|^2}, \qquad (3.43)$$

$$\Gamma_Z \propto |\langle +_{\text{cat}} |\hat{a}| -_{\text{cat}} \rangle|^2 = |\alpha|^2 \tanh(|\alpha|^2) \overset{|\alpha|^2 \to \infty}{\sim} |\alpha|^2.$$
(3.44)

We refer the interested reader to Refs. [67, 200, 201] for a more detailed discussion on error suppression properties of the cat qubit.

The equations above highlight a key advantage of the simpler cat qubit encoding: for sufficiently large coherent state amplitudes (α), the bit-flip rate of the encoded qubit can be suppressed to the point of becoming practically negligible. Consequently, the problem of correcting errors on the qubit is reduced to that of a classical bit, significantly simplifying the approach. This leaves only one type of error to correct, almost turning the quantum error correction challenge into a classical one and making the overall process of building a fault-tolerant quantum computer much more manageable.

We conclude this section by taking a step back and emphasizing that both the fourcomponent cat code introduced in Eqs. (3.38)-(3.39) and the ordinary cat code introduced in Eq. (3.40) correspond to coherent states arranged on a circle of radius $|\alpha|$ in phase space, with code states that are invariant under π and 2π rotations, respectively. This concept was further generalized by Grimsmo, Combes, and Baragiola [202], who consider rotation-symmetric bosonic (RSB) codes composed as superpositions of arbitrary states and not just coherent states. This framework also encompasses other well-known codes such as binomial codes [203]. Using this construction, it is possible to design a code that is protected against the loss of N - 1 photons by considering dual basis code states $(|\pm\rangle)$ that possess an N-fold discrete rotation symmetry in phase space, that is, they are invariant under discrete rotations by an angle $2\pi/N$.



Figure 3.3: Wigner functions W(q, p) and marginal distributions of the ordinary cat code, defined by the dual basis code states in Eq. (3.40), for the six different cardinal states given by, up to normalization, $\mathbf{a} |+Z\rangle = |0\rangle$, $\mathbf{b} |-Z\rangle = |1\rangle$, $\mathbf{c} |+X\rangle = |0\rangle + |1\rangle$, $\mathbf{d} |-X\rangle = |0\rangle - |1\rangle$, $\mathbf{e} |+Y\rangle = |0\rangle + i |1\rangle$, $\mathbf{f} |-Y\rangle = |0\rangle - i |1\rangle$. The amplitude of the coherent states is $\alpha = 2.5$

3.6.2 The Gottesman-Kitaev-Preskill code

While cat qubits cannot be viewed as stabilizer codes in the conventional sense, the bosonic encoding proposed by Gottesman, Kitaev, and Preskill (GKP) in 2001 [60] can be viewed as a stabilizer code. Indeed, GKP codes can be formally derived as the $d \to \infty$ limit of *d*-dimensional qudit stabilizer codes. Restricting to a single bosonic GKP qubit encoded in a single mode of the harmonic oscillator, the logical Pauli X and Z operators are given by

$$X = \exp(\alpha \hat{a}^{\dagger} - \alpha^* \hat{a}) = \hat{D}(\alpha), \qquad (3.45)$$

$$Z = \exp(\beta \hat{a}^{\dagger} - \beta^* \hat{a}) = \hat{D}(\beta), \qquad (3.46)$$

where the complex displacements α and β need to fulfill

$$\beta \alpha^* - \beta^* \alpha = i\pi, \tag{3.47}$$

such that X and Z commute up to a phase, that is, they reproduce the commutation relations of the Pauli operators, XZ = -ZX. Since X^2 should act like the identity on the code space, one can define the stabilizer generators of the GKP encoding as $S_X = X^2 = \hat{D}(2\alpha)$ and $S_Z = Z^2 = \hat{D}(2\beta)$. As the stabilizer generators are displacement operators with a particular amplitude, code words require a translation symmetry in phase space, which is determined by the choice of α and β that can be seen as the generators of a lattice on which the states have support. For brevity, we will restrict to the simplest and one of the most common choices, that is, the square lattice with $\alpha = \sqrt{\pi/2}$ and $\beta = i\alpha = i\sqrt{\pi/2}$. For this choice, logical operators and stabilizer generators reduce to

$$X = e^{-i\sqrt{\pi}\hat{p}}, \quad Z = e^{i\sqrt{\pi}\hat{q}}, \quad \hat{S}_X = e^{-i2\sqrt{\pi}\hat{p}}, \quad \hat{S}_Z = e^{i2\sqrt{\pi}\hat{q}}.$$
 (3.48)

Then, the ideal, infinite energy computational states are infinite trains of position eigenstates, that is,

$$\left|\overline{\mu}_{\rm GKP}\right\rangle = \sum_{n=-\infty}^{\infty} \left|q = (2n+\mu)\sqrt{\pi}\right\rangle,$$
(3.49)

with $\mu \in \{0, 1\}$. Measuring the S_X stabilizer is equivalent to measuring non-destructively the q quadrature modulo $\sqrt{\pi}$ and reveals any displacement error along the q quadrature. If the shift is less than $\sqrt{\pi}/2$, applying a displacement with the same magnitude and opposite sign restores the state into the code space without applying a logical operation. The same applies to shifts in the p quadrature.

The ideal code states of the GKP code are unphysical and non-normalizable. Nevertheless, it is possible to work with approximate code states for which high-energy contributions are exponentially suppressed. The finitely squeezed code states can be expressed in various equivalent ways [204], for example, in terms of a weighted sum of squeezed coherent states as [205]

$$|\mu_{\Delta}\rangle \propto \sum_{n=-\infty}^{\infty} e^{-\frac{\pi}{2}\Delta^2 (2n+\mu)^2} \hat{D}\left(\sqrt{\pi/2} \left(2n+\mu\right)\right) \hat{S}\left(-\ln\Delta\right) |0\rangle, \qquad (3.50)$$

where $\Delta \in [0, 1]$ and we have treated both quadratures symmetrically and point to Ref. [60] for the general case. We show the six cardinal states of this encoding in Figure 3.4. We mention that an alternative regularization can be achieved by applying the Fock damping operator to ideal code words, that is, $|\mu_{\delta}\rangle \propto \exp(-\delta^2 \hat{n}) |\mu_{\text{GKP}}\rangle$ and $|\mu_{\delta}\rangle \approx |\mu_{\Delta}\rangle$ for $\delta \approx \Delta$ and $\delta, \Delta \ll 1$ such that it is possible to use both views for intuitive explanations. Both of those representations yield a pure state. A representation of finite energy states that yields a mixed state is known as state twirling [206] and can be viewed as sending the state through the Gaussian displacement channel with a standard deviation that corresponds to the per-peak squeezing.

A relevant trait of the ideal GKP code is that all Clifford operations — and therefore also the syndrome extraction circuit — can be performed by Gaussian operations, that



Figure 3.4: Wigner functions W(q, p) and marginal distributions of the finite-energy Gottesman-Kitaev-Preskill code, see Eq. (3.50), for the six different cardinal states given by, up to normalization, $\mathbf{a} |+Z\rangle = |0\rangle_{\Delta}$, $\mathbf{b} |-Z\rangle = |1\rangle_{\Delta}$, $\mathbf{c} |+X\rangle = |0\rangle_{\Delta} + |1_{\Delta}\rangle$, $\mathbf{d} |-X\rangle = |0\rangle_{\Delta} - |1\rangle_{\Delta}$, $\mathbf{e} |+Y\rangle = |0\rangle_{\Delta} + i |1\rangle_{\Delta}$, $\mathbf{f} |-Y\rangle = |0\rangle_{\Delta} - i |1\rangle_{\Delta}$. The regularization parameter is chosen as $\Delta = 0.25$.

is, unitary operations that have a generating Hamiltonian which is at most quadratic in the quadrature operators. This makes this encoding particularly relevant for optical platforms where those operations are considered easily implementable. However, the Clifford operations of the finite-energy GKP code are not Gaussian anymore, indeed, they are non-unitary as they are obtained from the conjugation of the ideal operation with the non-unitary damping operator. Applying the ideal operations to the finite energy states thus yields additional errors. This makes the GKP encoding especially suited for measurement-based quantum computing, where the accumulation of errors is limited due to low-depth circuits followed by measurements [207–209].

When Gottesman, Kitaev, and Preskill proposed encoding a qubit into an oscillator 25 years ago, some considered their proposal a purely theoretical endeavor, due to the encoding being highly non-classical and therefore *beyond impossible* [210] to realize

experimentally. Fast forward 25 years, and preparing GKP states is yet another example of why we should think twice before calling something "impossible" — "difficult" is usually closer to the truth. Today, GKP states have been stabilized in motional states of trapped ions [211], superconducting circuits [43, 212, 213], and a universal gate set has been implemented in trapped ions as well [214]. Even on the optical side, significant progress has been made, and low-quality states have been prepared [215].

3.7 Conclusion and Outlook

In this chapter, we reviewed the framework for describing quantum information encoded into bosonic modes. Starting from the basic structure of states, observables, and transformations for a quantum harmonic oscillator, we established a formalism for quantum information processing with continuous variables.

Gaussian operations, for example, unitary evolution generated by a quadratic Hamiltonian, have an important role in continuous-variable quantum computing. They play an analogous role to Clifford gates in the discrete-variable setting in the sense that they are efficiently simulatable with classical computers, assuming Gaussian input states such as squeezed coherent states. Similar to their discrete-variable counterparts, these operations are not enough to prepare arbitrary quantum states, requiring, in this case, the addition of a non-Gaussian operation such as the cubic phase gate.

However, continuous-variable quantum systems are subject to noise, as any other quantum system. While analog stabilizer codes exist [216, 217], this approach has not been considered extensively due to several complications, such as unphysical code states and a no-go theorem for protecting Gaussian states against Gaussian errors in quantum communication protocols [218]. In practice, one therefore mostly considers discrete encodings of quantum information into quantum continuous variables, for example. **Paper IV** investigates the performance of two prominent classes of bosonic codes within the measurement-based paradigm of quantum computing. The work particularly focuses on the role of imperfect or noisy measurements when measuring the bosonic modes. To this end, numerically exact simulations of rotation-symmetric bosonic codes and Gottesman-Kitaev-Preskill codes under realistic measurement models are performed, revealing vulnerabilities and requirements for this approach to be viable with near-term devices.

Single-mode bosonic codes alone will not suffice to achieve fault-tolerant quantum computing, and they must therefore be concatenated with discrete-variable stabilizer codes described in the previous chapter. To bridge bosonic and discrete-variable codes, **Paper VI** proposes decoding strategies that explicitly exploit analog syndrome information available from bosonic qubit readout. These techniques apply to general concatenated architectures and reduce the need for repeated measurements, offering a promising route toward efficient fault-tolerant schemes based on those concatenated encodings.

Additionally, **Paper V** introduces the dissipatively stabilized squeezed cat qubit, a nonlocal encoding in phase space based on squeezed coherent states. This approach significantly enhances the error suppression properties of the dissipative cat qubit by deforming its basis states through a squeezing transformation. Importantly, for superconducting

circuits, the same physical device that dissipatively stabilizes cat qubits can, in principle, be used to stabilize squeezed cat qubits, only requiring two additional drive tones supplied through the same drive line. Indeed, in Ref. [80], the experimental stabilization of squeezed cat qubits and ordinary cat qubits on the same device was demonstrated, following the proposal of **Paper V**.

The following chapter of this thesis explores how one can utilize superconducting circuits to engineer almost arbitrary interactions and, in this way, realize quantum continuous variable systems in hardware.
CHAPTER 4

Quantum Computing Architectures

4.1 Superconducting Circuit Architectures

The field of circuit quantum electrodynamics (cQED) emerged after the discovery of the Josephson effect [219] as a way to explore quantum effects in macroscopic systems [220], including the observation of quantum tunneling [221] and the measurement of discrete energy levels [222]. Later, the observation of coherent oscillations in a superconducting qubit [223] showcased the potential of superconducting quantum circuits as a platform for quantum information processing [224]. Today, superconducting quantum circuits provide a highly flexible and controllable platform for quantum information processing. Their design leverages a toolbox of fundamental circuit elements, which, when combined strategically, allow for precise control over quantum states and interactions. These circuits enable the realization of artificial atoms [223, 225–228] with tunable energy levels, strong nonlinearities, and tailored couplings to their environment. Achieving high-fidelity quantum operations demands both a deep understanding of the approximations underlying effective circuit models and the development of analytical and numerical techniques to systematically obtain better approximations. Ultimately, the goal is for these refined methods to enable the engineering of quantum operations with significantly lower physical error rates.

In what follows, we begin by reviewing the Hamiltonian formulation for superconducting quantum circuits, setting the stage for a deeper understanding of these systems. Following this, we offer an intuitive approach to engineering effective interactions, both within the system and with its environment. To this end, we also propose a method to systematically account for higher-order effects accurately.

4.1.1 Lumped-element circuit diagrams

We briefly review the Hamiltonian formulation of superconducting circuits, see, for example, Refs. [133, 229, 230] for more details. In this approach, we describe an electrical

circuit in terms of a graph consisting of branches that represent two-terminal lumped circuit elements. Furthermore, the properties of the lowest frequency modes of distributedelement systems can also be described within this formalism if they are represented as lumped-element circuits. An example of a lumped-element circuit is shown in Figure 4.1a and we will discuss its different components in more detail in Section 4.1.2. To derive the equations of motion of the electrical circuit, we can construct its Lagrangian in terms of the energy associated with each circuit component.

In general, each branch element b of a circuit is characterized by a voltage $V_b(t)$ across it and a current $I_b(t)$ through it, see Figure 4.1b, which are defined in terms of the electromagnetic fields. The total energy stored in a branch element b is obtained by integrating the power $V_b(t)I_b(t)$ over time leading to

$$\mathcal{E}_b(t) = \int_{-\infty}^t V_b(t') I_b(t') dt'.$$
(4.1)

We also define the branch charge $Q_b(t)$ and the branch flux $\Phi_b(t)$ variables as

$$\Phi_b(t) = \int_{-\infty}^t V_b(t') \, \mathrm{d}t', \quad Q_b(t) = \int_{-\infty}^t I_b(t') \, \mathrm{d}t'$$
(4.2)

where we assumed that at $t' = -\infty$ the voltage and current are both equal to zero. The branch variables are not completely independent but related through Kirchhoff's laws,

$$\sum_{\text{all } b \text{ incident to } n} Q_b = q_n, \qquad \sum_{\text{all } b \text{ around } l} \Phi_b = \tilde{\Phi}_\ell, \tag{4.3}$$

such that the number of degrees of freedom will always be less than the number of branch elements because the directed sum of the voltages must be zero. Here, q_n is the charge at node n and $\tilde{\Phi}_{\ell}$ is the electromagnetic flux through the loop ℓ . The terminology of nodes and branches already hints at the fact that a natural language for analyzing circuits is graph theory.

Whether we want to choose a formulation in terms of branch fluxes $\Phi_b(t)$ or branch charges $Q_b(t)$ depends on the type of elements the circuit is composed of. In particular, this choice is determined by the element-dependent constitutive laws that relate $I_b(t)$ and $V_b(t)$. For a capacitive element, the constitutive relation has the form

$$V_b(t) = f_b(Q_b(t)),$$
 (4.4)

where in the case of an ideal capacitor $f_b(Q_b(t)) = Q_b(t)/C$ is a linear function. Similarly, an inductive element has a constitutive relation of the form

$$I_b(t) = g_b(\Phi_b(t)), \tag{4.5}$$

with $g_b(\Phi_b(t)) = \Phi_b(t)/L$ for an ideal inductor L. Typically, the relevant capacitive elements have a linear constitutive relation and we will choose to work with branch fluxes.



Figure 4.1: **a** Example of a lumped-element circuit consisting of a loop of Josephson junctions and a capacitor in parallel connected by a capacitor to another subcircuit formed by a linear inductor and a capacitor. **b** An arbitrary two-terminal branch element *b* with two nodes. The branch element is characterized by a voltage $V_b(t)$ across it and a current $I_b(t)$ through it, in the opposite direction. **c** A dc-SQUID, consisting of two Josephson junction with energies E_{J1} and E_{J2} in parallel, forming a loop. An external flux Φ_{ext} is threading the loop.

4.1.2 Circuit components

We will list here the three primary components of superconducting circuits and how they contribute to the Lagrangian of the circuit, see Ref. [231] for a derivation of these energy contributions. Additionally, voltage and current sources provide a means of (externally) manipulating the properties of the circuit in time. Lastly, we treat loop configurations of Josephson junctions, such as the dc-SQUID, as primary elements. We summarize all those elements in Table 4.1.

Capacitors. We consider capacitors as linear elements where the voltage is proportional to the charge stored on the capacitor $V(t) = q(t)/C = \dot{\Phi}(t)$, where C is the capacitance. The energy stored in a capacitor is given by

$$E = \frac{1}{2}C\dot{\Phi}^2(t) \tag{4.6}$$

resembling a kinetic energy term within the Lagrangian.

Inductors. For a linear inductor with inductance L, the current is related to the magnetic flux, $I(t) = \Phi(t)/L$. The energy stored in an inductor is given by

$$E = \frac{1}{2L} \Phi^2(t), \tag{4.7}$$

resembling a potential energy term within the Lagrangian.

Josephson junctions. A Josephson junction consists of two superconductors separated by a thin insulating layer. Because the layer is thin, Cooper pairs can tunnel through this barrier, leading to a current across the junction. The current depends on the phase difference φ between the wave functions describing the condensate of Cooper pairs on each superconductor, which is captured by the first Josephson relation [219]

$$I(t) = I_c \sin \varphi(t), \tag{4.8}$$

where I_c is the critical current that depends on the geometry of the junction. Additionally, the voltage between the superconductors is given by $V(t) = (\hbar/2e)\dot{\varphi}$. This allows us to view the phase difference as a generalized flux, that is, $\Phi = \hbar \varphi/2e$.

Up to a constant energy shift, the energy of the Josephson junction that behaves like a nonlinear inductance takes the form

$$E = I_c \int \left(\frac{\mathrm{d}t}{\mathrm{d}\Phi}\right) \sin\left(\frac{2\pi}{\Phi_0}\Phi\right) \mathrm{d}t = -E_J \cos\left[\frac{2\pi}{\Phi_0}\Phi(t)\right],\tag{4.9}$$

with $E_J = \Phi_0 I_c/2\pi$ the Josephson energy and $\Phi_0 = 2\pi\hbar/2e$ is the magnetic flux quantum. It is common to introduce the reduced flux $\varphi(t) = 2\pi\Phi(t)/\Phi_0$ to lighten the notation, as we will do in the following.

Voltage and current sources. The Lagrangian formulation introduced above allows us to incorporate ideal current and voltage sources into the circuit description in terms of node fluxes. A voltage source is modeled as a large capacitor $C_s \to \infty$ with large charge $Q_s \to \infty$ so that $Q_s/C_s = V_g$ is constant, where the constant V_g is the applied voltage. The voltage source is included in the Lagrangian by adding a kinetic energy term of the form

$$E = \frac{1}{2}C_g[\dot{\Phi}(t) - V_g(t)]^2.$$
(4.10)

Current sources are treated similarly, that is, we consider a large inductor $L_s \to \infty$ with large flux $\Phi_s \to \infty$ so that $\Phi_s/L_s = I_s$ is constant, where I_s is the applied current. Accordingly, we add a potential energy term of the form

$$E = I_s \Phi(t), \tag{4.11}$$

to the Lagrangian.

dc-SQUID. Above, we have introduced two methods, external current and voltage sources, to change the properties of the system (in time) by classical control. Another way to change the state of the system is by applying an external magnetic field through a loop formed by several Josephson junctions or a loop formed by a linear inductance and at least a single Josephson junction. The presence of an external magnetic flux $\Phi_{\text{ext}}(t)$ enters in the Lagrangian as a phase $\phi_{\text{ext}}(t) = 2\pi \Phi_{\text{ext}}(t)/\Phi_0$ due to the flux quantization (4.3). The potential energy of the SQUID, shown in Figure 4.1c, can be written as [232]

$$E = -E_{J1}\cos(\varphi + \alpha\phi_{\text{ext}}) - E_{J2}\cos[\varphi + (\alpha - 1)\phi_{\text{ext}}].$$
(4.12)

For time-independent $\phi_{\text{ext}}(t)$, the parameter $\alpha \in \mathbb{R}$ is a gauge degree of freedom and can be chosen arbitrarily: any choice will give rise to the same predictions. Recently, it has been pointed out [232, 233] that additional care should be taken when considering time-dependent magnetic fields¹. We will ignore such issues here, which is typically sufficient for qualitative results and gaining an intuition.

As an example, let us continue from Eq. (4.12), choosing $\alpha = 1/2$, then one can derive that the potential energy of the dc-SQUID corresponds to the potential of a single Josephson energy with a (flux) tunable Josephson energy $E_J(\phi_{\text{ext}})$, that is,

$$E = \underbrace{(E_{J1} + E_{J2})\sqrt{\cos^2(\phi_{\text{ext}}/2) + \delta^2 \sin^2(\phi_{\text{ext}}/2)}}_{E_J(\phi_{\text{ext}})} \cos(\varphi - \varphi_0), \qquad (4.13)$$

where $\varphi_0 = \arctan[\delta \tan(\phi_{\text{ext}}/2)]$ is an inconsequential flux offset and $\delta = (E_{J1} - E_{J2})/(E_{J1} + E_{J2})$ characterizes the asymmetry of the two junctions.

4.1.3 Method of nodes and the Lagrangian

Having introduced a toolbox of components for assembling superconducting circuits, we now briefly present the methods of nodes as a practical approach to modeling most circuits containing Josephson junctions. We will restrict this overview to static external fields, referring to [232, 233] for time-dependent fields. See Refs. [230, 231] for a more complete presentation.

Above, we already began utilizing the language of graph theory, and we will continue to do this here, especially to remove unnecessary degrees of freedom in a way that does not require Kirchhoff's law. As each branch flux Φ_b corresponds to an edge b in the network graph, we can equivalently represent it by two consecutive node fluxes ϕ_n and $\phi_{n'}$ that correspond to the vertices of b. Thus, all voltages will be determined as the difference of two node fluxes, and we can arbitrarily choose one reference node called ground to which we associate the value 0. Additionally, we want to choose a spanning tree \mathcal{T}_C of the circuit C, that is, informally, a connected subgraph of the circuit that contains all nodes but does not contain loops. The spanning tree naturally bipartitions the branches of the circuit into two sets, those contained within the spanning tree, and those contained in its closure (or complement) $\overline{\mathcal{T}}_C$. Whether a branch b is contained in the spanning tree or is a closure branch determines how we express it in terms of the node fluxes, that is,

$$\Phi_b = \begin{cases} \phi_n - \phi_{n'}, \, \forall b \in \mathcal{T}, \\ \phi_n - \phi_{n'} + \Phi_{\text{ext}}, \, \forall b \in \overline{\mathcal{T}}, \end{cases}$$
(4.14)

where Φ_{ext} is the external flux through the loop closed by the branch. We list the energies of the different components if they are contained in the spanning tree in Table 4.1, the energies for elements in the closure branch follow immediately from Eq. (4.14). From this, the Lagrangian of the circuit can be obtained as a signed sum of capacitive and inductive

¹This can be best understood by noting that the unitary transformation relating different gauges now requires additional terms in the Schrödinger equation, i.e., $H' = \hat{U}(t)H\hat{U}(t)^{\dagger} + i\frac{d}{dt}\hat{U}\hat{U}(t)^{\dagger}(t)$

elements with the latter carrying a minus sign. As customary, the classical Hamiltonian is obtained through the Legendre transformation of the Lagrangian, that is,

$$\mathcal{H} = \sum_{k} q_k \dot{\phi}_k - \mathcal{L}, \quad \text{with } q_k = \frac{\partial \mathcal{L}}{\partial \dot{\phi}_k}, \tag{4.15}$$

with the generalized momentum q_k also known as node charges.

4.1.4 Quantization and anharmonic oscillators

Given the classical Hamiltonian, we can obtain a quantum-mechanical description of the circuit through canonical quantization by promoting the variables to operators, that is, $\phi_k \rightarrow \hat{\phi}_k$ and $q_k \rightarrow \hat{q}_k$. With the node flux operator $\hat{\phi}_k$ corresponding to a generalized coordinate and the node charge operator \hat{q}_k corresponding to generalized momentum², they should obey the canonical commutation relations $[\hat{\phi}_n, \hat{q}_m] = i\delta_{nm}$ where δ_{nm} is the Kronecker delta.

In the case of an ordinary LC circuit, the quantum Hamiltonian is that of the quantum harmonic oscillator. We have seen previously, Section 3.3, that the harmonic oscillator is diagonalized by introducing the bosonic annihilation \hat{a} and creation \hat{a}^{\dagger} operators fulfilling the canonical canonical commutation relations $[\hat{a}, \hat{a}^{\dagger}] = \mathbb{1}$. In this case, the flux and charge operators are expressed as

$$\hat{\phi} = \phi_{\text{zpf}}(\hat{a} + \hat{a}^{\dagger}), \quad \hat{q} = -iq_{\text{zpf}}(\hat{a} - \hat{a}^{\dagger}), \quad (4.16)$$

with $\phi_{\text{zpf}} = \sqrt{\hbar Z_0/2}$ and $q_{\text{zpf}} = \sqrt{\hbar/2Z_0}$, the zero-point fluctuations of flux and charge, respectively, and $Z_0 = \sqrt{L/C}$ is known as the impedance. We introduce here the impedance explicitly as it plays an important role in the quantization of nonlinear circuits, as we will see in the following.

For superconducting circuits that include Josephson junctions, the Hamiltonian is inherently nonlinear, preventing direct diagonalization in terms of bosonic annihilation and creation operators. Nevertheless, in the weakly nonlinear regime, a common strategy is to first solve the linearized version of the circuit to obtain a basis in which the system is diagonal and then treat the junction's nonlinearity as a small perturbation. This approximation holds when the system remains in the low-energy regime, where flux fluctuations are confined near the minimum of the potential. Under this assumption, the flux across the junction can be expressed as a linear combination of the M mode system, that is,

$$\hat{\phi} = \sum_{m=1}^{M} \sqrt{\frac{\hbar \mathcal{Z}_m^{\text{eff}}}{2}} (\hat{a}_m + \hat{a}_m^{\dagger}), \qquad (4.17)$$

where $\mathcal{Z}_m^{\text{eff}}$ is the effective impedance of the m^{th} mode of the circuit, a classical quantity, that is, in principle measurable. This approach is known as blackbox quantization [234].

²We have maneuvered ourselves into a corner. The node charge operators \hat{q}_k that take the role of a generalized momentum variable should not be confused with the position quadrature \hat{q} introduced in Chapter 3.

Element	\mathbf{Symbol}	Spanning tree	Black-box quantization
Capacitor	•	$rac{C}{2}(\dot{\phi}_n-\dot{\phi}_{n'})^2$	$q_{ m zpf}^2(\hat{b}-\hat{b}^\dagger)^2$
Inductor		$\frac{1}{2L}(\phi_n - \phi_{n'})^2$	$\phi_{\rm zpf}^2(\hat{b}+\hat{b}^\dagger)^2$
Josephson junction	$- \square -$	$-E_J\cos(\phi_n-\phi_{n'})$	$-E_J \cos[\sum_m \phi_m (\hat{b}_m + \hat{b}_m^{\dagger})]$
Current source		$I_S \phi_n$	$I_S(t)(\hat{b}+\hat{b}^\dagger)$

Table 4.1: Circuit elements and their corresponding representations as symbols in diagrams, as Lagrangian terms if they are contained in the spanning tree, and as terms in the Hamiltonian within the framework of black-box quantization. For the Josephson junction, ϕ_m denotes the zero-point fluctuations of the m^{th} mode mixing in the junction.

Note that in the junction, typically all modes of the circuit mix, allowing for the design of higher-order interactions between the modes.

For a nonlinear superconducting circuit in a parameter regime for which the black box quantization approach is valid, the linearized circuit is obtained through the Taylor expansion of the potential energy in the Hamiltonian. For example, for a single Josephson junction, we have

$$E_J \cos \hat{\phi} = E_J - \frac{E_J}{2} \hat{\phi}^2 + \frac{E_J}{24} \hat{\phi}^4 + O(\hat{\phi}^6).$$
(4.18)

Ignoring the constant term and incorporating the quadratic contribution into an effective linear inductance, the remaining quartic term introduces an anharmonicity to the otherwise linear system. Combining the Josephson junction with a capacitor results in what is commonly referred to as an anharmonic quantum oscillator³. For a single mode, expressing the quantized Hamiltonian in terms of the bosonic operators \hat{a} and \hat{a}^{\dagger} ,

$$\hat{H}_{\rm AHO} = \omega \hat{a}^{\dagger} \hat{a} + \frac{E_J}{24} \phi_{\rm zpf}^4 (\hat{a} + \hat{a}^{\dagger})^4 \approx \omega_{\rm eff} \hat{a}^{\dagger} \hat{a} - K \hat{a}^{\dagger 2} \hat{a}^2, \qquad (4.19)$$

one often finds the anharmonic oscillator in the latter form. Here, $\omega_{\text{eff}} = \omega - 6K$ is the effective frequency of the oscillator and the parameter K is known as the Kerr nonlinearity. The approximation we have performed above is known as the rotating wave approximation and will be described in more detail in the following section.

³In a particular parameter regime, the resulting circuit can be viewed either as the charge qubit [223] or the transmon qubit [235–237], see also Ref. [238] for a review.

4.2 Engineering Effective Interactions

The dominating energy scale in a Hamiltonian that describes a superconducting quantum circuit is typically set by the resonance frequency of the circuit ω_r which is on the order of GHz. Any other coupling strengths, such as nonlinearities from Josephson junctions, are usually small compared to ω_r and typically on the order of tens to hundreds of kHz or MHz. They can be seen as perturbations to the evolution of the system governed by the oscillator Hamiltonian $\hat{H}_0 = \omega_r \hat{a}^{\dagger} \hat{a}$. Hence, any desired interaction can only be obtained effectively with respect to a specific frame, i.e., the interaction picture with respect to the free evolution \hat{H}_0 . Because the corresponding unitary transformation $\hat{U}(t) = \exp(i\hat{H}_0 t)$ results in a rotation of the phase space with frequency ω_r , it is common to refer to this frame as the rotating frame.

In the rotating frame, the effective Hamiltonian takes the form

$$\hat{H}_{\rm eff}(t) = \hat{U}(t)\hat{H}\hat{U}(t)^{\dagger} + i\frac{\mathrm{d}\hat{U}(t)}{\mathrm{d}t}\hat{U}^{\dagger}(t) = \hat{U}(t)\hat{H}\hat{U}(t)^{\dagger} - \omega_r \hat{a}^{\dagger}\hat{a}.$$
(4.20)

The transformation $\hat{U}(t)\hat{H}\hat{U}(t)^{\dagger}$ corresponds to the replacement $\hat{a} \rightarrow \hat{a}(t) = \hat{a}e^{-i\omega_r t}$ and the transformation for the creation operator \hat{a}^{\dagger} is obtained by Hermitian conjugation. Therefore, $\hat{H}_{\text{eff}}(t)$ will often become time-dependent even if \hat{H} is time-independent. Because ω_r is on the order of GHz, this leads to fast oscillations which approximately average out over relevant time-scales set by a given interaction with strength g on the order of MHz or less. They can therefore be neglected if their coupling g is small compared to their effective oscillatory frequency. This is known as the rotating wave approximation (RWA) [239, 240].

Energy preserving terms, for example, diagonal terms such as the Kerr interaction $K\hat{a}^{\dagger 2}\hat{a}^2$ are time-independent (and thus resonant) in any rotating frame. On the other hand, energy non-conserving terms, for example, off-diagonal terms containing an unequal number of creation and annihilation operators, become time-dependent and have negligible effects. To select these processes resonantly, it is therefore necessary to achieve an external time-dependent modulation of the relevant prefactors that counters the time-dependence of $\hat{a}^{(\dagger)}$ so that the desired interaction becomes effectively time-independent in the rotating frame. For example, the squeezing type interaction $g_{sq}(a^2 + a^{+2})$ requires a time modulation of the coupling strength at twice the frequency of the oscillator, that is, $g_{sq} \rightarrow g_{sq} \cos(2w_r t)$.

In cQED, there are two common approaches to achieve modulation of couplings in time, which we are going to discuss now qualitatively. The first method is based on modulating the flux through a superconducting loop, such a the SQUID, see Section 4.1.2. This approach is commonly used for parametric amplification [241] or to engineer an exchange interaction between two modes [242, 243]. Ignoring the issue of quantization in the presence of time-dependent fluxes, the potential term of the SQUID containing the

external magnetic flux $\varphi_{\text{ext}}(t)$ can be written as [242]

$$\cos(\varphi - \varphi_{\text{ext}}(t)) = \cos(\varphi) \cos(\varphi_{\text{ext}}(t)) - \sin(\varphi) \sin(\varphi_{\text{ext}}(t))$$

$$\approx \left(1 - \frac{1}{2!}\varphi^2 + \frac{1}{4!}\varphi^4\right) \cos(\varphi_{\text{ext}}(t)) - \left(\varphi - \frac{1}{3!}\varphi^3 + \frac{1}{5!}\varphi^5\right) \sin(\varphi_{\text{ext}}(t))$$

$$+ \mathcal{O}(\varphi^6).$$
(4.21)

This results in time-dependent couplings in general. In principle, any arbitrary interaction can be selected by an appropriate modulation of $\varphi_{\text{ext}}(t)$.

Alternatively, effective time-dependent couplings can be obtained by applying an external current source, which leads to a displacement $\alpha(t)$ of the annihilation operator $\hat{a} \rightarrow \hat{a} + \alpha(t)$. Here, the displacement parameter $\alpha(t)$ depends on the driving strength and the time-dependence is determined by the external modulation of the current source (see also Section 4.1.2). Thus, the presence of an *n*-th order nonlinearity allows for resonantly selecting nonlinear terms of order n-1 or lower, which can be seen from the binomial identity,

$$\left(\hat{a} + \hat{a}^{\dagger}\right)^{n} \to \left(\hat{a} + \hat{a}^{\dagger} + \alpha(t) + \alpha^{*}(t)\right)^{n}$$

$$(4.22)$$

$$=\sum_{k=0}^{n} {\binom{n}{k}} \left(\hat{a} + \hat{a}^{\dagger}\right)^{n-k} \left(\alpha(t) + \alpha^{*}(t)\right)^{k}.$$
(4.23)

Both approaches have advantages and drawbacks depending on the concrete microwave architecture that is considered and the specific interaction that is desired.

Engineering Effective Environments

It is not always sufficient to solely engineer the interaction of closed quantum systems, as in some cases we desire to engineer the dissipative dynamics of an open quantum system. In particular, we aim for dynamics that do not drive the state of the system to a "trivial" thermal state but rather to a desired quantum state or manifold of quantum states. For example, imagine that we wish to engineer the dissipation such that, ideally, the system of interest evolves according to the master equation,

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\rho} = \mathcal{D}[\hat{A}]\hat{\rho},\tag{4.24}$$

where $\mathcal{D}[\hat{A}]$ is the Lindblad dissipator with arbitrary jump operator \hat{A} , compare Section 3.6.1 where we required $\hat{A} = \hat{a}^2 - \alpha^2$ to confine the states of the harmonic oscillator to the code space of the cat qubit. It is possible to engineer such effective dissipation channels by starting from a larger system and adiabatically eliminating the dynamics of all but the system of interest. To exemplify this procedure, let us consider a two-mode system (\hat{a}, \hat{b}) evolving according to the master equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\rho}_{SB} = -i\left[\hat{H}_{SB}, \hat{\rho}_{SB}\right] + \kappa_S \mathcal{D}[\hat{a}]\hat{\rho}_{SB} + \kappa_B \mathcal{D}[\hat{b}]\hat{\rho}_{SB}, \qquad (4.25)$$

where $\hat{H}_{SB} = g(\hat{A}\hat{b}^{\dagger} + \hat{A}^{\dagger}\hat{b})$ is the system-bath interaction with rate g and \hat{b} is the bosonic annihilation operator of the auxiliary mode. We emphasize that the operator \hat{A} can be an arbitrary function of \hat{a} and \hat{a}^{\dagger} . For the system and the auxiliary mode, single-photon losses with rates κ_S and κ_B , respectively, occur. Consider a regime characterized by a small dimensionless parameter λ such that $g/\kappa_B \sim \lambda$ and $\kappa_S/\kappa_B \sim \lambda^2$. Phrased similarly, the rate of single-photon losses of the auxiliary system is the dominating energy scale of the system, and we have $\kappa_S \ll q \ll \kappa_B$. We can therefore assume that the state of the auxiliary mode is close to the vacuum state and thus has a photon population much smaller than one. To gain intuition, consider the case where the auxiliary mode B is in the vacuum state while the system S is in an arbitrary state, i.e., $\hat{\rho}_{SB} = \hat{\rho}_S \otimes |0\rangle \langle 0|_B$. In this case, the interaction $\hat{A}\hat{b}^{\dagger}$ will generally be allowed whereas the reverse process, $\hat{A}^{\dagger}\hat{b}$, is prohibited due to $\hat{b} |0\rangle = 0$. That is, the system S is transferring an effective excitation \hat{A} to the ancilla B, which creates a photon (\hat{b}^{\dagger}) of the same energy. As the reverse process is forbidden, the system is effectively dissipating \hat{A} . As a result, the system evolves towards a state $\hat{\rho}$ that satisfies $\hat{A}\hat{\rho} = 0$. This state must lie within the kernel of the operator \hat{A} . If there is only a single state within the kernel of \hat{A} , the state satisfying $\hat{A}\hat{\rho} = 0$ becomes the unique stationary state irrespective of the initial state. If the kernel of A contains multiple states, the stationary state is not unique and will depend upon the initial state.

To make the above arguments more rigorous, let us consider an initial state of the form

$$\hat{\rho}_{SB} = \hat{\rho}_{00} |0\rangle\langle 0| + \lambda \left[\hat{\rho}_{10} |1\rangle\langle 0| + \hat{\rho}_{01} |0\rangle\langle 1|\right] + \lambda^2 \hat{\rho}_{11} |1\rangle\langle 1|.$$
(4.26)

Here we used $\hat{\rho}_{nm}$ to denote the projection of $\hat{\rho}_{SB}$ onto the Fock states n, m of the ancillary mode. We aim to derive the effective dynamics of the system alone, given by $\hat{\rho}_S = \text{Tr}_B[\hat{\rho}_{SB}] = \hat{\rho}_{00} + \lambda^2 \hat{\rho}_{11}$ up to second-order in λ . To this end, we assume $||\hat{A}|| = O(1)$ in λ . Following the derivation that can be found in Refs. [62, 244], the time evolution of the different $\hat{\rho}_{mn}$ are given by

$$\frac{1}{\kappa_B} \frac{\mathrm{d}}{\mathrm{d}t} \hat{\rho}_{00} = -i \frac{g}{\kappa_B} (\hat{A}^{\dagger} \hat{\rho}_{10} - \hat{\rho}_{01} \hat{A}) + \frac{\kappa_S}{\kappa_B} \mathcal{D}[\hat{a}] \hat{\rho}_{00} + \hat{\rho}_{11} + \dots, \qquad (4.27)$$

$$\frac{1}{\kappa_B} \frac{\mathrm{d}}{\mathrm{d}t} \hat{\rho}_{10} = -i \frac{g}{\kappa_B} \hat{A} \hat{\rho}_{00} - \frac{1}{2} \hat{\rho}_{10} + \dots, \qquad (4.28)$$

$$\frac{1}{\kappa_B} \frac{\mathrm{d}}{\mathrm{d}t} \hat{\rho}_{11} = -i \frac{g}{\kappa_B} (\hat{A} \hat{\rho}_{01} - \hat{\rho}_{10} \hat{A}^{\dagger}) - \hat{\rho}_{11} + \dots$$
(4.29)

To solve this system of equations, we can make the adiabatic approximation, assuming that $\hat{\rho}_{10}$ is continuously in its steady state, that is, $\partial_t \hat{\rho}_{10} = 0$ and similarly $\partial_t \hat{\rho}_{11} = 0$. Within this approximation, $\hat{\rho}_{10}$ and $\hat{\rho}_{11}$ are determined solely by $\hat{\rho}_{00}$ (and \hat{A}), and we arrive at the desired result

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{\rho}_S = \kappa_A \mathcal{D}[\hat{A}]\hat{\rho}_S + \kappa_S \mathcal{D}[\hat{a}]\hat{\rho}_S, \qquad (4.30)$$

with $\kappa_A = 4g^2/\kappa_B$. We note that another possible approach, which is much more general than this example, is based upon the effective operator formalism for open quantum systems by Reiter and Sørensen [245].

4.3 Extracting Effective Models

As the prospect of fault-tolerant quantum computing relies critically on the quality of quantum-gate operations, access to coherent gates with low error rates is crucial. While the qualitative approach described above is typically a good starting point for engineering proof-of-concept effective interactions, engineering high-fidelity operations routinely requires going beyond the rotation wave approximation, and, for qubit systems, going beyond the two-level approximation.

4.3.1 The Schrieffer-Wolff transformation

The Schrieffer-Wolff transformation [246, 247] is an analytical tool that allows one to accurately and systematically capture effects that go beyond the rotation wave approximation and many other approximations.

The traditional perturbative approach due to Rayleigh and Schrödinger provides corrections to the energy levels and eigenstates of the Hamiltonian $\hat{H} = \hat{H}_0 + \hat{V}$ with respect to the unperturbed properties of eigenstates and eigenvalues of \hat{H}_0 due to a perturbation \hat{V} . However, one might argue that the modern view of quantum mechanics is less concerned with states and energies, but that one rather becomes more used to thinking in terms of Hermitian operators, that is, elementary Hamiltonians, and how combining them affects properties of the system. Fortunately, Schrieffer-Wolff perturbation theory is formulated on the level of operators and can therefore be viewed as a modern approach to perturbation theory in quantum mechanics.

In a nutshell, Schrieffer-Wolff perturbation theory is concerned with finding an anti-Hermitian generator $\hat{S} = -\hat{S}^{\dagger}$ of a unitary transformation $\hat{H}_{\text{eff}} = \exp(\hat{S})\hat{H}\exp(\hat{S}^{\dagger})$ which transforms $\hat{H} = \hat{H}_0 + \hat{V}$ to a basis in which it is *more* diagonal. Here, *more* diagonal refers to a transformation that suppresses the high-energy contributions of the perturbation \hat{V} up to a desired order, while retaining its influence within the low-energy subspace. To be explicit, by choosing the generator \hat{S} to satisfy the operator equation $[\hat{H}_0, \hat{S}] = V$, one obtains an effective Hamiltonian

$$\hat{H}_{\text{eff}} = \hat{H}_0 + \frac{1}{2} \Big[\hat{S}, \hat{V} \Big] + O(\hat{V}^3), \qquad (4.31)$$

for which the perturbation \hat{V} is removed to first order. In principle, one can remove the off-diagonal part of the perturbation up to arbitrary order by an appropriate choice of \hat{S} .

However, historically, the difficulty of the Schrieffer-Wolff transformation was to apply it to higher orders, as no constructive methods to solve the respective operator equations existed beyond the first order [248].

4.4 Conclusion and Outlook

In this chapter, the fundamental building blocks of superconducting circuits have been outlined as a means to engineer complex quantum systems. The focus has been placed on how these systems can realize controllable nonlinear interactions between quantum degrees of freedom, enabling the realization of protected quantum memories, quantum gates, and tailored dissipative dynamics.

At the heart of circuit quantum electrodynamics (cQED) lies the principle that, by combining Josephson junctions with linear circuit elements, effective Hamiltonians of remarkable complexity can be "printed" directly onto a substrate. However, deriving the desired dynamics from the set of available circuit components requires accurate perturbative techniques for extracting effective low-energy models that faithfully capture the essential physics. To this end, **Paper II** introduces an iterative procedure for constructing the generators of the Schrieffer-Wolff transformation to arbitrary perturbative order⁴. This provides a systematic method to eliminate high-energy contributions while computing effective models directly at the operator level. Crucially, it allows one to obtain an approximately linear effective Hamiltonian from an inherently nonlinear quantum system by judiciously tuning the coefficients of low-degree nonlinearities such that they cancel in the effective model.

This idea lies at the heart of **Paper I**, which proposes an explicit construction of a universal gate set for continuous-variable quantum computation with superconducting circuits based on the superconducting nonlinear inductive element (SNAIL) [250]. The proposed architecture enables the implementation of both Gaussian and non-Gaussian gates, such as the cubic phase gate, by selectively activating interactions through parametric modulation. At the same time, the static, i.e., undriven, effective dynamics remain (approximately) linear. This separation of static and driven dynamics allows for high-fidelity gate operations while suppressing spurious nonlinear effects.

The theoretical proposal of **Paper I** is successfully demonstrated experimentally in **Paper III**, showcasing the transition from theoretical design to practical realization. The techniques developed and employed in this chapter provide a foundation for bridging theoretical constructions with experimental implementations, a central theme that runs throughout this thesis.

⁴Around the same time, Ref. [249] introduced a similar method.

CHAPTER 5

Conclusion and Perspective

This thesis has explored quantum low-density parity-check (qLDPC) codes and bosonic codes based on quantum continuous variables as foundational elements for scalable fault-tolerant quantum computing. Throughout, a central aim has been to bridge the gap between abstract theoretical advances and the realities of physical hardware, developing abstractions that shift experimental challenges toward more achievable targets. By examining the interplay between code design, decoding complexity, and physical implementation constraints, this work offers insights into advancing quantum error correction strategies for near-term devices while developing conceptual foundations for scalable architectures beyond the limitations of current technology.

To this end, on the near-term level, we provide both a theoretical proposal and an experimental demonstration of a universal gate set for continuous-variable quantum computing, leveraging the hardware-native nonlinearities of the superconducting nonlinear asymmetric inductive element (SNAIL). This showcases the theoretically proposed versatility of the SNAIL, enabling access to both charge-driven and flux-driven interactions while remaining approximately Kerr-free.

Additionally, the dissipatively stabilized squeezed cat qubit, a noise-biased bosonic encoding, is introduced. Compared to the ordinary cat qubit, squeezed cat qubits offer significantly improved error protection against bit-flips with very limited effect on the protection against phase-flips. These advantages are retained under a concrete noise model derived from the proposed dissipative stabilization mechanism. Recently, Rousseau *et al.* [80] experimentally demonstrated the enhanced protection, with exceptional agreement to the theoretical proposal.

Going beyond near-term devices, this thesis proposes quantum radial codes as a high-threshold, low-overhead, and single-shot quantum memory. While these codes cannot achieve asymptotic optimality in encoding rate and distance, the emphasis lies on realistic scenarios involving circuit-level noise and finite, practically relevant code sizes. In this regime, their performance is competitive with bivariate bicycle codes [25, 251]. However, they offer an alternative geometric perspective that may prove advantageous for constructing fault-tolerant operations or tailored decoders.

Furthermore, as a step towards solving the problem of accurate real-time decoding of qLDPC codes, the localized statistics decoding algorithm is introduced as the first parallel decoding algorithm that matches the performance of the current state-of-the-art ordered statistics decoding algorithm for general quantum error correction protocols. Most abstractly, the thesis introduces the concept of fault complexes, which broadens the homological framework of quantum error correction from static CSS codes to dynamic, time-evolving fault-tolerant protocols. Fault complexes open the door to applying the full machinery of homology theory to the analysis and design of quantum error correction protocols in space-time.

However, building a fault-tolerant quantum computer remains a daunting challenge, where theoretical ideas can shape experiments by pushing the boundaries of what is necessary and what is possible, and experimental realities reshape theoretical imagination. A timely and prominent example is a class of bivariate bicycle codes investigated by Bravyi *et al.* [25], which, with their high encoding rate and threshold, compete with the surface code, which has remained unchallenged in terms of its high error threshold for almost 20 years. While the investigated codes require some long-range connectivity in addition to the otherwise nearest-neighbor connectivity in the plane, the significant reduction in overhead motivates the additional effort of engineering long-range couplers within a superconducting architecture.

Studies such as the one by Bravyi *et al.* [25] and the work presented on quantum radial codes in this thesis show that to build a high-threshold, low-overhead quantum computer, it is crucial to simulate potential constructions within the sheer endless design space. The reason is that currently, the only method to accurately estimate the error threshold of a quantum error-correcting code is by direct simulation with a realistic circuit-level noise model. However, an additional complication with these realistic noise models is that the ordering of operations during the syndrome extraction cycle becomes relevant and can significantly impact the fault distance of the code. However, no general, efficient tools exist to determine the circuit distance of a code in the presence of noise.

A promising avenue to approach this problem is rooted within the formalism of fault complexes extended to circuit-level noise. Intuitively, there should exist a mapping from the phenomenological noise case to the circuit-level noise case, potentially described by a chain map, that is, a homomorphism of chain complexes. Given such a mapping, one might uncover (simple) conditions under which the fault distance is preserved, enabling more precise predictions of code performance.

Furthermore, it might be worthwhile to consider an approach to quantum error correction inspired by classical error correction. In classical coding theory, decoder performance often dictates the direction of code development. One might argue that such an approach has been partially followed by rephrasing the decoding problem of various topological codes such that it becomes matchable, see e.g., [252–254]. However, this approach has been limited and only considered minimum-weight error decoders.

The performance gap between (classical) minimum-weight error and (quantum) maximum-likelihood decoders for circuit-level noise is largely unexplored with a few exceptions, e.g., [255], due to the complexity and associated overhead of the decoder. While certain code families [130, 140] have efficient maximum-likelihood decoders in

the static, i.e., code capacity, case, to the best of our knowledge, no efficient maximumlikelihood decoding algorithms exist for codes in space-time. Developing such decoders is a crucial open problem that will require further theoretical insights on the structure of the decoding problem in space-time. This represents an exciting area of future research, with the potential to significantly enhance the performance of quantum error correction on near-term devices and beyond.

Even though gate fidelities and coherence times for discrete-variable qubits are steadily increasing, bosonic encodings potentially offer a shortcut to lower effective physical error rates. A major challenge ahead is the development of scalable architectures involving multiple bosonic modes, along with the validation of corresponding theoretical error models. For stabilized cat qubits, the implementation of high-fidelity two-qubit gates remains a key obstacle. While a fully bosonic two-qubit gate between two cat qubits has not vet been demonstrated, experimental progress suggests that such a realization is within reach. In the meantime, two-qubit operations between hybrid cat-transmon architectures have been successfully demonstrated, enabling operation of the repetition code near the memory threshold [42]. Although the current memory lifetime is not limited by the two-qubit gate fidelity, improving these operations will be critical for future scaling. One possible avenue is to replace stabilized cat qubits with stabilized squeezed cat qubits, which may reduce control errors related to the narrower peak structure of the squeezed cat qubit wavefunction compared to the ordinary cat qubit. Overall, the sheer vastness of the superconducting qubit design space offers enormous potential to further enhance bosonic qubit performance — potentially beyond what currently seems imaginable.

Fault-tolerant quantum computing stands at an extraordinary moment. Theoretical ideas, once thought too abstract or futuristic, can now find their way into laboratory demonstrations within years, if not months. This rapid interplay between theory and experiment creates a momentum that is both exhilarating and unprecedented. In many ways, the current pace and scope of progress evoke the intellectual thrill that must have accompanied the discovery of quantum mechanics a century ago — when a new formalism redefined our understanding of nature. Likewise, the rapid development today demands that we reconsider the mathematical formalism to describe quantum error-correcting codes. While the language of chain complexes has been tremendously fruitful for the discovery of error-correcting codes, we must ask ourselves whether thinking about codes limits our imagination. Recent discoveries [27, 138, 256–259] point toward a broader perspective, that there must exist a more fundamental structure than codes. This is what makes this field so exciting right now. It's this push beyond traditional frameworks, toward something fundamentally new, that makes this a truly exciting moment for the field.

References

- P. Shor, Algorithms for quantum computation: discrete logarithms and factoring, in Proc. 35th Annu. Symp. Found. Comput. Sci. (Nov. 1994), pp. 124–134.
- [2] P. W. Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, SIAM J. Comput. 26, 1484–1509 (1997).
- [3] S. Lloyd, Universal Quantum Simulators, Science 273, 1073–1078 (1996).
- [4] P. Benioff, The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines, J Stat Phys 22, 563–591 (1980).
- [5] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. 21, 467– 488 (1982).
- [6] B. Bauer, S. Bravyi, M. Motta, and G. K.-L. Chan, Quantum Algorithms for Quantum Chemistry and Quantum Materials Science, Chem. Rev. 120, 12685– 12717 (2020).
- [7] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, Simulated Quantum Computation of Molecular Energies, Science 309, 1704–1707 (2005).
- [8] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Elucidating reaction mechanisms on quantum computers, Proc. Natl. Acad. Sci. 114, 7555–7560 (2017).
- [9] Y. Cao et al., Quantum Chemistry in the Age of Quantum Computing, Chem. Rev. 119, 10856–10915 (2019).
- [10] M. Motta and J. E. Rice, Emerging quantum computing algorithms for quantum chemistry, WIREs Comput. Mol. Sci. 12, e1580 (2022).
- [11] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Quantum computational chemistry, Rev. Mod. Phys. 92, 015003 (2020).
- [12] R. Santagati et al., Drug design on quantum computers, Nat. Phys. 20, 549–557 (2024).
- [13] M. E. Beverland, A. Kubica, and K. M. Svore, Cost of Universality: A Comparative Study of the Overhead of State Distillation and Code Switching with Color Codes, PRX Quantum 2, 020341 (2021).
- [14] C. Gidney and M. Ekerå, How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits, Quantum 5, 433 (2021).
- [15] D. Spielman and S.-H. Teng, Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time, in Proc. Thirty-Third Annu. ACM Symp. Theory Comput. STOC '01 (July 2001), pp. 296–305, arXiv:cs/0111050.

- [16] M. A. Nielsen and I. L. Chuang, Programmable Quantum Gate Arrays, Phys. Rev. Lett. 79, 321–324 (1997).
- [17] D. Zierler, Michael Nielsen (Postdoc '98-'00), Quantum Physicist, Author, and Open Science Advocate, Oct. 2022.
- [18] A. Y. Kitaev, Fault-tolerant quantum computation by anyons, Annals of Physics 303, 2–30 (2003).
- [19] S. B. Bravyi and A. Y. Kitaev, Quantum codes on a lattice with boundary, ArXivquant-Ph9811052 (1998), arXiv:quant-ph/9811052.
- [20] M. H. Freedman and D. A. Meyer, Projective Plane and Planar Quantum Codes, Found. Comput. Math. 1, 325–332 (2001).
- [21] N. P. Breuckmann and J. N. Eberhardt, Quantum Low-Density Parity-Check Codes, PRX Quantum 2, 040101 (2021).
- [22] P. Panteleev and G. Kalachev, Asymptotically good Quantum and locally testable classical LDPC codes, in Proc. 54th Annu. ACM SIGACT Symp. Theory Comput. STOC 2022 (June 2022), pp. 375–388.
- [23] I. Dinur, M.-H. Hsieh, T.-C. Lin, and T. Vidick, Good Quantum LDPC Codes with Linear Time Decoders, June 2022, arXiv:2206.07750 [quant-ph].
- [24] A. Leverrier and G. Zémor, Quantum Tanner codes, in 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS) (Oct. 2022), pp. 872–883, arXiv:2202.13641 [quant-ph].
- [25] S. Bravyi, A. W. Cross, J. M. Gambetta, D. Maslov, P. Rall, and T. J. Yoder, High-threshold and low-overhead fault-tolerant quantum memory, Nature 627, 778–782 (2024).
- [26] S. Yoshida, S. Tamiya, and H. Yamasaki, Concatenate codes, save qubits, Feb. 2024, arXiv:2402.09606 [quant-ph].
- [27] M. H. Shaw and B. M. Terhal, Lowering Connectivity Requirements for Bivariate Bicycle Codes Using Morphing Circuits, Phys. Rev. Lett. 134, 090602 (2025), arXiv:2407.16336 [quant-ph].
- [28] S. Wolanski and B. Barber, Introducing Ambiguity Clustering: An Accurate and Efficient Decoder for qLDPC Codes, in 2024 IEEE International Conference on Quantum Computing and Engineering (QCE) (Sept. 2024), pp. 402–403, arXiv:2406.14527 [quant-ph].
- [29] K. R. Ott, B. Hetényi, and M. E. Beverland, Decision-tree decoders for general quantum LDPC codes, Feb. 2025, arXiv:2502.16408 [quant-ph].
- [30] L. A. Beni, O. Higgott, and N. Shutty, Tesseract: A Search-Based Decoder for Quantum Error Correction, Mar. 2025, arXiv:2503.10988 [quant-ph].
- [31] K. Yin, X. Fang, J. Ruan, H. Zhang, D. Tullsen, A. Sornborger, C. Liu, A. Li, T. Humble, and Y. Ding, SymBreak: Mitigating Quantum Degeneracy Issues in QLDPC Code Decoders by Breaking Symmetry, Dec. 2024, arXiv:2412.02885 [quant-ph].
- [32] A. deMarti iOlius, I. E. Martinez, J. Roffe, and J. E. Martinez, An almost-linear time decoding algorithm for quantum LDPC codes under circuit-level noise, Sept. 2024, arXiv:2409.01440 [quant-ph].

- [33] A. Gong, S. Cammerer, and J. M. Renes, Toward Low-latency Iterative Decoding of QLDPC Codes Under Circuit-Level Noise, Mar. 2024, arXiv:2403.18901 [quant-ph].
- [34] A. Cowtan, SSIP: automated surgery with quantum LDPC codes, July 2024, arXiv:2407.09423 [quant-ph].
- [35] A. Cowtan, Z. He, D. J. Williamson, and T. J. Yoder, Parallel Logical Measurements via Quantum Code Surgery, Mar. 2025, arXiv:2503.05003 [quant-ph].
- [36] Z. He, A. Cowtan, D. J. Williamson, and T. J. Yoder, Extractors: QLDPC Architectures for Efficient Pauli-Based Computation, Mar. 2025, arXiv:2503.10390 [quant-ph].
- [37] B. Ide, M. G. Gowda, P. J. Nadkarni, and G. Dauphinais, Fault-tolerant logical measurements via homological measurement, Oct. 2024, arXiv:2410.02753 [quant-ph].
- [38] E. Swaroop, T. Jochym-O'Connor, and T. J. Yoder, Universal adapters between quantum LDPC codes, Oct. 2024, arXiv:2410.03628 [quant-ph].
- [39] A. Cross, Z. He, P. Rall, and T. Yoder, *Linear-Size Ancilla Systems for Logical Measurements in QLDPC Codes*, July 2024, arXiv:2407.18393 [quant-ph].
- [40] D. J. Williamson and T. J. Yoder, Low-overhead fault-tolerant quantum computation by gauging logical operators, Oct. 2024, arXiv:2410.02213 [cond-mat, physics:quant-ph].
- [41] R. Acharya et al., Quantum error correction below the surface code threshold, Nature 638, 920–926 (2025).
- [42] H. Putterman et al., Hardware-efficient quantum error correction via concatenated bosonic qubits, Nature 638, 927–934 (2025), arXiv:2409.13025 [quant-ph].
- [43] V. V. Sivak et al., Real-time quantum error correction beyond break-even, Nature 616, 50–55 (2023), arXiv:2211.09116 [quant-ph].
- S. Krinner et al., Realizing Repeated Quantum Error Correction in a Distance-Three Surface Code, ArXiv211203708 Cond-Mat Physicsquant-Ph (2021), arXiv:2112. 03708 [cond-mat, physics:quant-ph].
- [45] I. Besedin et al., Realizing Lattice Surgery on Two Distance-Three Repetition Codes with Superconducting Qubits, Jan. 2025, arXiv:2501.04612 [quant-ph].
- [46] R. S. Gupta et al., Encoding a magic state with beyond break-even fidelity, Nature 625, 259–263 (2024).
- [47] N. Sundaresan, T. J. Yoder, Y. Kim, M. Li, E. H. Chen, G. Harper, T. Thorbeck, A. W. Cross, A. D. Córcoles, and M. Takita, Demonstrating multi-round subsystem quantum error correction using matching and maximum likelihood decoders, Nat Commun 14, 2852 (2023).
- [48] D. Bluvstein et al., Logical quantum processor based on reconfigurable atom arrays, Nature **626**, 58–65 (2024).
- [49] P. S. Rodriguez et al., Experimental Demonstration of Logical Magic State Distillation, Dec. 2024, arXiv:2412.15165 [quant-ph].
- [50] B. W. Reichardt et al., Demonstration of quantum computation and error correction with a tesseract code, Sept. 2024, arXiv:2409.04628 [quant-ph].

- [51] N. Berthusen et al., Experiments with the four-dimensional surface code on a quantum charge-coupled device quantum computer, Phys. Rev. A 110, 062413 (2024), arXiv:2408.08865 [quant-ph].
- [52] V. G. Matsos, C. H. Valahu, T. Navickas, A. D. Rao, M. J. Millican, X. C. Kolesnikow, M. J. Biercuk, and T. R. Tan, Robust and Deterministic Preparation of Bosonic Logical States in a Trapped Ion, Phys. Rev. Lett. 133, 050602 (2024).
- [53] C. Ryan-Anderson et al., High-fidelity teleportation of a logical qubit using transversal gates and lattice surgery, Science 385, 1327–1331 (2024), arXiv:2404.16728 [quant-ph].
- [54] M. P. da Silva et al., Demonstration of logical qubits and repeated error correction with better-than-physical error rates, Apr. 2024, arXiv:2404.02280 [quant-ph].
- [55] S. A. Moses et al., A Race-Track Trapped-Ion Quantum Processor, Phys. Rev. X 13, 041052 (2023), arXiv:2305.03828 [quant-ph].
- [56] C. Ryan-Anderson et al., Realization of Real-Time Fault-Tolerant Quantum Error Correction, Phys. Rev. X 11, 041058 (2021).
- [57] A. J. Brady, A. Eickbusch, S. Singh, J. Wu, and Q. Zhuang, Advances in bosonic quantum error correction with Gottesman–Kitaev–Preskill Codes: Theory, engineering and applications, Progress in Quantum Electronics 93, 100496 (2024), arXiv:2308.02913 [quant-ph].
- [58] W. Cai, Y. Ma, W. Wang, C.-L. Zou, and L. Sun, Bosonic quantum error correction codes in superconducting quantum circuits, Fundamental Research 1, 50–67 (2021), arXiv:2010.08699.
- [59] A. Joshi, K. Noh, and Y. Y. Gao, Quantum information processing with bosonic qubits in circuit QED, Quantum Sci. Technol. 6, 033001 (2021), arXiv:2008.13471.
- [60] D. Gottesman, A. Kitaev, and J. Preskill, Encoding a qubit in an oscillator, Phys. Rev. A 64, 012310 (2001).
- [61] Z. Leghtas, G. Kirchmair, B. Vlastakis, R. J. Schoelkopf, M. H. Devoret, and M. Mirrahimi, Hardware-Efficient Autonomous Quantum Memory Protection, Phys. Rev. Lett. **111**, 120501 (2013).
- [62] Z. Leghtas et al., Confining the state of light to a quantum manifold by engineered two-photon loss, Science 347, 853–857 (2015).
- [63] S. Puri, S. Boutin, and A. Blais, Engineering the quantum states of light in a Kerr-nonlinear resonator by two-photon driving, npj Quantum Inf. 3, 18 (2017).
- [64] J. Guillaud and M. Mirrahimi, Repetition Cat Qubits for Fault-Tolerant Quantum Computation, Phys. Rev. X 9, 041053 (2019).
- [65] B. Royer, S. Singh, and S. M. Girvin, Stabilization of Finite-Energy Gottesman-Kitaev-Preskill States, Phys. Rev. Lett. 125, 260509 (2020), arXiv:2009.07941.
- [66] H. Putterman, J. Iverson, Q. Xu, L. Jiang, O. Painter, F. G. S. L. Brandão, and K. Noh, Stabilizing a Bosonic Qubit Using Colored Dissipation, Phys. Rev. Lett. 128, 110502 (2022), arXiv:2107.09198.
- [67] K. S. Dubovitskii, Bit-flip errors in dissipative cat qubits: Second-order perturbation theory, Phys. Rev. A 111, 012617 (2025), arXiv:2407.17299 [quant-ph].
- [68] D. Ruiz, J. Guillaud, A. Leverrier, M. Mirrahimi, and C. Vuillot, LDPC-cat codes for low-overhead quantum computing in 2D, Nat Commun 16, 1040 (2025).

- [69] Y. Xu, Y. Wang, C. Vuillot, and V. V. Albert, Letting the tiger out of its cage: bosonic coding without concatenation, Nov. 2024, arXiv:2411.09668.
- [70] C. T. Hann, K. Noh, H. Putterman, M. H. Matheny, J. K. Iverson, M. T. Fang, C. Chamberland, O. Painter, and F. G. S. L. Brandão, *Hybrid cat-transmon* architecture for scalable, hardware-efficient quantum error correction, Oct. 2024, arXiv:2410.23363.
- [71] T. Matsuura, N. C. Menicucci, and H. Yamasaki, Continuous-Variable Fault-Tolerant Quantum Computation under General Noise, Oct. 2024, arXiv:2410.12365 [quant-ph].
- [72] M. H. Shaw, A. C. Doherty, and A. L. Grimsmo, Logical Gates and Read-Out of Superconducting Gottesman-Kitaev-Preskill Qubits, Mar. 2024, arXiv:2403.02396 [quant-ph].
- [73] R. Gautier, M. Mirrahimi, and A. Sarlette, Designing High-Fidelity Gates for Dissipative Cat Qubits, Mar. 2023, arXiv:2303.00760 [quant-ph].
- [74] J. Hastrup and U. L. Andersen, All-optical cat-code quantum error correction, Phys. Rev. Res. 4, 043065 (2022), arXiv:2108.12225.
- [75] J. Hastrup, J. S. Neergaard-Nielsen, and U. L. Andersen, Deterministic generation of a four-component optical cat state, Opt. Lett., OL 45, 640–643 (2020), arXiv:1912. 09880.
- [76] J. Hastrup and U. L. Andersen, Improved readout of qubit-coupled Gottesman– Kitaev–Preskill states, Quantum Sci. Technol. 6, 035016 (2021), arXiv:2008.10531.
- [77] D. J. Weigand and B. M. Terhal, Generating grid states from Schrödinger-cat states without postselection, Phys. Rev. A 97, 022341 (2018).
- [78] D. J. Weigand and B. M. Terhal, Realizing modular quadrature measurements via a tunable photon-pressure coupling in circuit QED, Phys. Rev. A 101, 053840 (2020), arXiv:1909.10075 [quant-ph].
- [79] K. Duivenvoorden, B. M. Terhal, and D. Weigand, Single-mode displacement sensor, Phys. Rev. A 95, 012305 (2017).
- [80] R. Rousseau et al., Enhancing dissipative cat qubit protection by squeezing, Feb. 2025, arXiv:2502.07892 [quant-ph].
- [81] U. Réglade et al., Quantum control of a cat qubit with bit-flip times exceeding ten seconds, Nature 629, 778–783 (2024), arXiv:2307.06617 [quant-ph].
- [82] B. de Neeve, T.-L. Nguyen, T. Behrle, and J. P. Home, Error correction of a logical grid state qubit by dissipative pumping, Nat. Phys. 18, 296–300 (2022), arXiv:2010.09681 [quant-ph].
- [83] C. Berdou et al., One Hundred Second Bit-Flip Time in a Two-Photon Dissipative Oscillator, PRX Quantum 4, 020350 (2023), arXiv:2204.09128.
- [84] A. Marquet et al., Autoparametric Resonance Extending the Bit-Flip Time of a Cat Qubit up to 0.3 s, Phys. Rev. X 14, 021019 (2024).
- [85] A. Vanselow, B. Beauseigneur, L. Lattier, M. Villiers, A. Denis, P. Morfin, Z. Leghtas, and P. Campagne-Ibarcq, *Dissipating quartets of excitations in a super-conducting circuit*, Jan. 2025, arXiv:2501.05960 [quant-ph].
- [86] B. Vlastakis, G. Kirchmair, Z. Leghtas, S. E. Nigg, L. Frunzio, S. M. Girvin, M. Mirrahimi, M. H. Devoret, and R. J. Schoelkopf, Deterministically Encoding

Quantum Information Using 100-Photon Schrodinger Cat States, Science **342**, 607–610 (2013).

- [87] N. Ofek et al., Extending the lifetime of a quantum bit with error correction in superconducting circuits, Nature 536, 441–445 (2016).
- [88] X. Pan, J. Schwinger, N.-N. Huang, P. Song, W. Chua, F. Hanamura, A. Joshi, F. Valadares, R. Filip, and Y. Y. Gao, *Protecting the quantum interference of cat states by phase-space compression*, Dec. 2022, arXiv:2212.01271 [quant-ph].
- [89] O. Milul, B. Guttel, U. Goldblatt, S. Hazanov, L. M. Joshi, D. Chausovsky, N. Kahn, E. Çiftyürek, F. Lafont, and S. Rosenblum, Superconducting Cavity Qubit with Tens of Milliseconds Single-Photon Coherence Time, PRX Quantum 4, 030336 (2023), arXiv:2302.06442 [quant-ph].
- [90] F. Arute et al., Quantum supremacy using a programmable superconducting processor, Nature 574, 505–510 (2019), arXiv:1910.11333.
- [91] A. Morvan et al., Phase transitions in random circuit sampling, Nature 634, 328– 333 (2024), arXiv:2304.11119 [quant-ph].
- [92] D. Gao et al., Establishing a New Benchmark in Quantum Computational Advantage with 105-qubit Zuchongzhi 3.0 Processor, Phys. Rev. Lett. 134, 090601 (2025).
- [93] J. S. Bell, On the Einstein Podolsky Rosen paradox, Physics Physique Fizika 1, 195–200 (1964).
- [94] A. Y. Kitaev, Quantum computations: algorithms and error correction, Russ. Math. Surv. 52, 1191 (1997).
- [95] C. M. Dawson and M. A. Nielsen, The Solovay-Kitaev algorithm, Quantum Info. Comput. 6, 81–95 (2006).
- [96] D. Gottesman, The Heisenberg representation of quantum computers, in 22nd Int. Colloq. Group Theor. Methods Phys. (July 1998), pp. 32–43, arXiv:quantph/9807006.
- [97] D. E. Deutsch, Quantum computational networks, Proc. R. Soc. Lond. Math. Phys. Sci. 425, 73–90 (1997).
- [98] R. Raussendorf and H. J. Briegel, A One-Way Quantum Computer, Phys. Rev. Lett. 86, 5188–5191 (2001).
- [99] R. Raussendorf, D. E. Browne, and H. J. Briegel, Measurement-based quantum computation on cluster states, Phys. Rev. A 68, 022312 (2003).
- [100] M. A. Nielsen, Quantum computation by measurement and quantum memory, Physics Letters A 308, 96–100 (2003).
- [101] D. W. Leung, Quantum computation by measurements, Int. J. Quantum Inform. 02, 33–43 (2004).
- [102] A. M. Childs, D. W. Leung, and M. A. Nielsen, Unified derivations of measurementbased schemes for quantum computation, Phys. Rev. A 71, 032318 (2005).
- [103] D. Litinski, A Game of Surface Codes: Large-Scale Quantum Computing with Lattice Surgery, Quantum 3, 128 (2019).
- [104] D. Horsman, A. G. Fowler, S. Devitt, and R. V. Meter, Surface code quantum computing by lattice surgery, New J. Phys. 14, 123011 (2012).
- [105] O. Goldreich, Short Locally Testable Codes and Proofs (Survey), tech. rep. TR05-014 (Electronic Colloquium on Computational Complexity (ECCC), Jan. 2005).

- [106] H. Bombín, Single-Shot Fault-Tolerant Quantum Error Correction, Phys. Rev. X 5, 031043 (2015).
- [107] F. MacWilliams and N. Sloane, The Theory of Error-Correcting Codes, 2nd ed. (North-holland Publishing Company, 1978).
- [108] T. Richardson and R. Urbanke, Modern Coding Theory (Cambridge University Press, Cambridge, 2008).
- [109] D. Poulin, J.-P. Tillich, and H. Ollivier, Quantum Serial Turbo Codes, IEEE Trans. Inf. Theory 55, 2776–2798 (2009).
- [110] P. W. Shor, Scheme for reducing decoherence in quantum computer memory, Phys. Rev. A 52, R2493–R2496 (1995).
- [111] A. M. Steane, Error Correcting Codes in Quantum Theory, Phys. Rev. Lett. 77, 793–797 (1996).
- [112] A. Steane, Multiple-particle interference and quantum error correction, Proc. R. Soc. Lond. Ser. Math. Phys. Eng. Sci. 452, 2551–2577 (1996).
- [113] D. Gottesman, Stabilizer Codes and Quantum Error Correction, May 1997, arXiv:quantph/9705052.
- [114] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, Quantum Error Correction and Orthogonal Geometry, Phys. Rev. Lett. 78, 405–408 (1997).
- [115] A. R. Calderbank and P. W. Shor, Good quantum error-correcting codes exist, Phys. Rev. A 54, 1098–1105 (1996).
- [116] A. Bolt, G. Duclos-Cianci, D. Poulin, and T. M. Stace, Foliated Quantum Error-Correcting Codes, Phys. Rev. Lett. 117, 070501 (2016).
- [117] R. Raussendorf, J. Harrington, and K. Goyal, Topological fault-tolerance in cluster state quantum computation, New J. Phys. 9, 199–199 (2007).
- [118] D. Poulin, Stabilizer Formalism for Operator Quantum Error Correction, Phys. Rev. Lett. 95, 230504 (2005).
- [119] B. J. Brown and S. Roberts, Universal fault-tolerant measurement-based quantum computation, Phys. Rev. Research 2, 033305 (2020).
- [120] A. Hatcher, Algebraic topology (Cambridge University Press, Cambridge, 2002), pp. xii+544.
- [121] R. G. Gallager, Low density parity check codes, Thesis (Massachusetts Institute of Technology, 1960).
- [122] R. Gallager, Low-density parity-check codes, IRE Trans. Inf. Theory 8, 21–28 (1962).
- [123] N. P. Breuckmann and J. N. Eberhardt, Balanced Product Quantum Codes, IEEE Trans. Inf. Theory 67, 6653–6674 (2021).
- [124] M. Sipser and D. Spielman, Expander codes, IEEE Trans. Inf. Theory 42, 1710– 1722 (1996).
- [125] B. Audoux and A. Couvreur, On tensor products of CSS codes, Ann. L'Institut Henri Poincaré D 6, 239–287 (2019).
- [126] J.-P. Tillich and G. Zémor, Quantum LDPC codes with positive rate and minimum distance proportional to $n^{1/2}$, in 2009 IEEE Int. Symp. Inf. Theory (June 2009), pp. 799–803.

- [127] J.-P. Tillich and G. Zémor, Quantum LDPC Codes With Positive Rate and Minimum Distance Proportional to the Square Root of the Blocklength, IEEE Trans. Inf. Theory 60, 1193–1202 (2014).
- [128] W. Zeng and L. P. Pryadko, Higher-Dimensional Quantum Hypergraph-Product Codes with Finite Rates, Phys. Rev. Lett. **122**, 230501 (2019).
- [129] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A 86, 032324 (2012).
- [130] P. Iyer and D. Poulin, Hardness of Decoding Quantum Stabilizer Codes, IEEE Trans. Inf. Theor. 61, 5209–5223 (2015), arXiv:1310.3235 [quant-ph].
- [131] E. Berlekamp, R. McEliece, and v. Tilborg H.C.A., On the inherent intractability of certain coding problems, IEEE Trans. Inf. Theory 24, 384–386 (1978).
- [132] B. M. Terhal, Quantum error correction for quantum memories, Rev. Mod. Phys. 87, 307–346 (2015).
- [133] A. Blais, S. M. Girvin, and W. D. Oliver, Quantum information processing and quantum optics with circuit quantum electrodynamics, Nat. Phys., 1–10 (2020).
- [134] H. Aghaee Rad et al., Scaling and networking a modular photonic quantum computer, Nature 638, 912–919 (2025).
- [135] S. Bartolucci et al., Fusion-based quantum computation, Nat Commun 14, 912 (2023).
- [136] D. Bacon, Software of QIP, by QIP, and for QIP, Pasadena, California, 2022.
- [137] D. Bacon, S. T. Flammia, A. W. Harrow, and J. Shi, Sparse Quantum Codes From Quantum Circuits, IEEE Trans. Inf. Theory 63, 2464–2479 (2017).
- [138] N. Delfosse and A. Paetznick, Spacetime codes of Clifford circuits, Apr. 2023, arXiv:2304.05943 [quant-ph].
- [139] T. Hillmann, G. Dauphinais, I. Tzitrin, and M. Vasmer, Single-shot and measurementbased quantum error correction via fault complexes, Oct. 2024, arXiv:2410.12963 [quant-ph].
- [140] D. Poulin, Optimal and efficient decoding of concatenated quantum block codes, Phys. Rev. A 74, 052333 (2006).
- [141] S. Bravyi, M. Suchara, and A. Vargo, Efficient algorithms for maximum likelihood decoding in the surface code, Phys. Rev. A 90, 032326 (2014).
- [142] A. J. Ferris and D. Poulin, Tensor Networks and Quantum Error Correction, Phys. Rev. Lett. 113, 030501 (2014).
- [143] C. T. Chubb, General tensor network decoding of 2D Pauli codes, Oct. 2021, arXiv:2101.04125 [quant-ph].
- [144] M.-H. Hsieh and F. Le Gall, NP-hardness of decoding quantum error-correction codes, Phys. Rev. A 83, 052331 (2011).
- [145] K.-Y. Kuo and C.-C. Lu, On the hardness of decoding quantum stabilizer codes under the depolarizing channel, in 2012 Int. Symp. Inf. Theory Its Appl. (Oct. 2012), pp. 208–211.
- [146] M. Wilde, "Quantum convolutional codes", in *Quantum Error Correction*, edited by D. A. Lidar and T. A. Brun (Cambridge University Press, Cambridge, 2013), pp. 231–260.

- [147] P. Fuentes, J. Etxezarreta Martinez, P. M. Crespo, and J. Garcia-Frías, Degeneracy and Its Impact on the Decoding of Sparse Quantum Codes, IEEE Access 9, 89093– 89119 (2021).
- [148] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, Topological quantum memory, J. Math. Phys. 43, 4452–4505 (2002).
- [149] J. Edmonds and E. L. Johnson, Matching, Euler tours and the Chinese postman, Mathematical Programming 5, 88–124 (1973).
- [150] A. G. Fowler, Minimum weight perfect matching of fault-tolerant topological quantum error correction in average O(1) parallel time, Quantum Info. Comput. 15, 145–158 (2015), arXiv:1307.1740 [quant-ph].
- [151] H. Bombin and M. A. Martin-Delgado, Optimal resources for topological twodimensional stabilizer codes: Comparative study, Phys. Rev. A 76, 012305 (2007).
- [152] A. Fischer and A. Miyake, Hardness results for decoding the surface code with Pauli noise, Quantum 8, 1511 (2024).
- [153] S. Aji and R. McEliece, The generalized distributive law, IEEE Trans. Inf. Theory 46, 325–343 (2000).
- [154] R. Tanner, A recursive approach to low complexity codes, IEEE Trans. Inf. Theory 27, 533–547 (1981).
- [155] F. Kschischang, B. Frey, and H.-A. Loeliger, Factor graphs and the sum-product algorithm, IEEE Trans. Inform. Theory 47, 498–519 (2001).
- [156] W. Ryan and S. Lin, Channel Codes: Classical and Modern (Cambridge University Press, Cambridge, 2009).
- [157] T. R. Scruby, T. Hillmann, and J. Roffe, High-threshold, low-overhead and singleshot decodable fault-tolerant quantum memory, June 2024, arXiv:2406.14445 [quant-ph].
- [158] D. J. C. MacKay and M. S. Postol, Weaknesses of Margulis and Ramanujan-Margulis Low-Density Parity-Check Codes, Electronic Notes in Theoretical Computer Science, MFCSIT 2002, The Second Irish Conference on the Mathematical Foundations of Computer Science and Information Technology 74, 97–104 (2003).
- [159] R. Koetter and P. O. Vontobel, Graph-covers and iterative decoding of finite length codes, in Proc 3rd Intern Symp Turbo Codes Relat. Top. (2003), pp. 1–5.
- [160] N. Raveendran and B. Vasic, Trapping Sets of Quantum LDPC Codes, Quantum 5, 562 (2021).
- [161] C. Di, D. Proietti, I. Telatar, T. Richardson, and R. Urbanke, Finite-length analysis of low-density parity-check codes on the binary erasure channel, IEEE Trans. Inf. Theory 48, 1570–1579 (2002).
- [162] N. Wiberg, Codes and decoding on general graphs (Univ, Linköping, 1996).
- [163] M. Fossorier and S. Lin, Soft-decision decoding of linear block codes based on ordered statistics, IEEE Trans. Inf. Theory 41, 1379–1396 (1995).
- [164] P. Panteleev and G. Kalachev, Degenerate Quantum LDPC Codes With Good Finite Length Performance, Quantum 5, 585 (2021).
- [165] M. Fossorier, S. Lin, and J. Snyders, Reliability-based syndrome decoding of linear block codes, IEEE Trans. Inf. Theory 44, 388–398 (1998).
- [166] J. Roffe, D. R. White, S. Burton, and E. Campbell, Decoding across the quantum low-density parity-check code landscape, Phys. Rev. Res. 2, 043423 (2020).

- [167] E. T. Campbell, A theory of single-shot error correction for adversarial noise, Quantum Sci. Technol. 4, 025006 (2019).
- [168] K. Husimi, Some Formal Properties of the Density Matrix, Proc. Phys.-Math. Soc. Jpn. 3rd Ser. 22, 264–314 (1940).
- [169] R. J. Glauber, Coherent and Incoherent States of the Radiation Field, Phys. Rev. 131, 2766–2788 (1963).
- [170] E. C. G. Sudarshan, Equivalence of Semiclassical and Quantum Mechanical Descriptions of Statistical Light Beams, Phys. Rev. Lett. 10, 277–279 (1963).
- [171] E. Wigner, On the Quantum Correction For Thermodynamic Equilibrium, Phys. Rev. 40, 749–759 (1932).
- [172] V. Gorini, A. Kossakowski, and E. C. G. Sudarshan, Completely positive dynamical semigroups of N-level systems, J. Math. Phys. 17, 821–825 (1976).
- [173] G. Lindblad, On the generators of quantum dynamical semigroups, Commun. Math. Phys. 48, 119–130 (1976).
- [174] J. S. Ivan, K. K. Sabapathy, and R. Simon, Operator-sum representation for bosonic Gaussian channels, Phys. Rev. A 84, 042311 (2011).
- [175] K.-E. Hellwig and K. Kraus, Operations and measurements. II, Commun.Math. Phys. 16, 142–147 (1970).
- [176] E. C. G. Sudarshan, P. M. Mathews, and J. Rau, Stochastic Dynamics of Quantum-Mechanical Systems, Phys. Rev. 121, 920–924 (1961).
- [177] M.-D. Choi, Completely positive linear maps on complex matrices, Linear Algebra and its Applications 10, 285–290 (1975).
- [178] W. F. Stinespring, Positive functions on C*-algebras, Proc. Amer. Math. Soc. 6, 211–216 (1955).
- [179] M. O. Scully and M. S. Zubairy, Quantum Optics (Cambridge University Press, Cambridge, 1997).
- [180] C. W. Gardiner, Quantum Noise (Springer-Verlag, 1991).
- [181] H.-P. Breuer and F. Petruccione, The Theory of Open Quantum Systems (Oxford University Press, Oxford, New York, June 2002).
- [182] S. Lloyd and S. L. Braunstein, Quantum Computation over Continuous Variables, Phys. Rev. Lett. 82, 1784–1787 (1999).
- [183] F. Arzani, R. I. Booth, and U. Chabaud, Can effective descriptions of bosonic systems be considered complete?, Jan. 2025, arXiv:2501.13857 [quant-ph].
- [184] S. L. Braunstein and P. van Loock, Quantum information with continuous variables, Rev. Mod. Phys. 77, 513–577 (2005).
- [185] C. Weedbrook, S. Pirandola, R. García-Patrón, N. J. Cerf, T. C. Ralph, J. H. Shapiro, and S. Lloyd, Gaussian quantum information, Rev. Mod. Phys. 84, 621– 669 (2012).
- [186] P. Marek, R. Filip, and A. Furusawa, Deterministic implementation of weak quantum cubic nonlinearity, Phys. Rev. A 84, 053802 (2011).
- [187] M. Yukawa, K. Miyata, H. Yonezawa, P. Marek, R. Filip, and A. Furusawa, Emulating quantum cubic nonlinearity, Phys. Rev. A 88, 053816 (2013).
- [188] F. Arzani, N. Treps, and G. Ferrini, Polynomial approximation of non-Gaussian unitaries by counting one photon at a time, Phys. Rev. A 95, 052352 (2017).

- [189] K. Miyata, H. Ogawa, P. Marek, R. Filip, H. Yonezawa, J.-i. Yoshikawa, and A. Furusawa, Implementation of a quantum cubic gate by an adaptive non-Gaussian measurement, Phys. Rev. A 93, 022301 (2016).
- [190] R. Yanagimoto, T. Onodera, E. Ng, L. G. Wright, P. L. McMahon, and H. Mabuchi, Engineering a Kerr-Based Deterministic Cubic Phase Gate via Gaussian Operations, Phys. Rev. Lett. **124**, 240503 (2020), arXiv:1912.11408.
- [191] P. Faist, S. Nezami, V. V. Albert, G. Salton, F. Pastawski, P. Hayden, and J. Preskill, Continuous Symmetries and Approximate Quantum Error Correction, Phys. Rev. X 10, 041018 (2020).
- [192] B. Eastin and E. Knill, Restrictions on Transversal Encoded Quantum Gate Sets, Phys. Rev. Lett. 102, 110502 (2009).
- [193] L. Hänggli and R. König, Oscillator-to-Oscillator Codes Do Not Have a Threshold, IEEE Trans. Inf. Theory 68, 1068–1084 (2022), arXiv:2102.05545.
- [194] A. P. Lund, T. C. Ralph, and H. L. Haselgrove, Fault-Tolerant Linear Optical Quantum Computing with Small-Amplitude Coherent States, Phys. Rev. Lett. 100, 030503 (2008).
- [195] H. Jeong and M. S. Kim, Efficient quantum computation using coherent states, Phys. Rev. A 65, 042305 (2002).
- [196] T. C. Ralph, A. Gilchrist, G. J. Milburn, W. J. Munro, and S. Glancy, Quantum computation with optical coherent states, Phys. Rev. A 68, 042319 (2003).
- [197] M. Mirrahimi, Z. Leghtas, V. V. Albert, S. Touzard, R. J. Schoelkopf, L. Jiang, and M. H. Devoret, Dynamically protected cat-qubits: a new paradigm for universal quantum computation, New J. Phys. 16, 045014 (2014).
- [198] V. V. Albert, Lindbladians with multiple steady states: theory and applications, PhD thesis (Yale University, Jan. 2018), arXiv:1802.00010.
- [199] E. Knill and R. Laflamme, Theory of quantum error-correcting codes, Phys. Rev. A 55, 900–911 (1997).
- [200] M. Mirrahimi, Cat-qubits for quantum computation, Comptes Rendus Phys. 17, 778–787 (2016).
- [201] S. Puri et al., Bias-preserving gates with stabilized cat qubits, Sci. Adv. 6, eaay5901 (2020), arXiv:1905.00450.
- [202] A. L. Grimsmo, J. Combes, and B. Q. Baragiola, Quantum Computing with Rotation-Symmetric Bosonic Codes, Phys. Rev. X 10, 011058 (2020).
- [203] M. H. Michael, M. Silveri, R. T. Brierley, V. V. Albert, J. Salmilehto, L. Jiang, and S. M. Girvin, New Class of Quantum Error-Correcting Codes for a Bosonic Mode, Phys. Rev. X 6, 031006 (2016).
- [204] T. Matsuura, H. Yamasaki, and M. Koashi, Equivalence of approximate Gottesman-Kitaev-Preskill codes, Phys. Rev. A 102, 032408 (2020).
- [205] V. V. Albert et al., Performance and structure of single-mode bosonic codes, Phys. Rev. A 97, 032346 (2018).
- [206] K. Noh and C. Chamberland, Fault-tolerant bosonic quantum error correction with the surface–Gottesman-Kitaev-Preskill code, Phys. Rev. A 101, 012316 (2020).
- [207] N. C. Menicucci, P. van Loock, M. Gu, C. Weedbrook, T. C. Ralph, and M. A. Nielsen, Universal Quantum Computation with Continuous-Variable Cluster States, Phys. Rev. Lett. 97, 110501 (2006).

- [208] N. C. Menicucci, Fault-Tolerant Measurement-Based Quantum Computing with Continuous-Variable Cluster States, Phys. Rev. Lett. 112, 120504 (2014).
- [209] M. Jafarzadeh, J. Conrad, R. N. Alexander, and B. Q. Baragiola, Logical channels in approximate Gottesman-Kitaev-Preskill error correction, Apr. 2025, arXiv:2504. 13383 [quant-ph].
- [210] Byron Bay Quantum Computing Workshop, *GKP retrospective & bosonic codes* panel discussion with Gottesman, Devoret, Divincenzo, & Girvin, Nov. 2020.
- [211] C. Flühmann, T. L. Nguyen, M. Marinelli, V. Negnevitsky, K. Mehta, and J. P. Home, Encoding a qubit in a trapped-ion mechanical oscillator, Nature 566, 513– 517 (2019).
- [212] P. Campagne-Ibarcq et al., Quantum error correction of a qubit encoded in grid states of an oscillator, Nature 584, 368–372 (2020), arXiv:1907.12487.
- [213] B. L. Brock, S. Singh, A. Eickbusch, V. V. Sivak, A. Z. Ding, L. Frunzio, S. M. Girvin, and M. H. Devoret, *Quantum Error Correction of Qudits Beyond Break-even*, Sept. 2024, arXiv:2409.15065 [quant-ph].
- [214] V. G. Matsos, C. H. Valahu, M. J. Millican, T. Navickas, X. C. Kolesnikow, M. J. Biercuk, and T. R. Tan, Universal Quantum Gate Set for Gottesman-Kitaev-Preskill Logical Qubits, Sept. 2024, arXiv:2409.05455 [quant-ph].
- [215] S. Konno et al., Logical states for fault-tolerant quantum computation with propagating light, Science **383**, 289–293 (2024), arXiv:2309.02306.
- [216] S. Lloyd and J.-J. E. Slotine, Analog Quantum Error Correction, Phys. Rev. Lett. 80, 4088–4091 (1998).
- [217] V. V. Albert, Bosonic coding: introduction and use cases, Nov. 2022, arXiv:2211. 05714 [quant-ph].
- [218] J. Niset, J. Fiurášek, and N. J. Cerf, No-Go Theorem for Gaussian Quantum Error Correction, Phys. Rev. Lett. 102, 120501 (2009).
- [219] B. D. Josephson, Possible new effects in superconductive tunnelling, Phys. Lett. 1, 251–253 (1962).
- [220] J. Clarke, A. N. Cleland, M. H. Devoret, D. Esteve, and J. M. Martinis, Quantum Mechanics of a Macroscopic Variable: The Phase Difference of a Josephson Junction, Science 239, 992–997 (1988).
- [221] M. H. Devoret, J. M. Martinis, and J. Clarke, Measurements of Macroscopic Quantum Tunneling out of the Zero-Voltage State of a Current-Biased Josephson Junction, Phys. Rev. Lett. 55, 1908–1911 (1985).
- [222] J. M. Martinis, M. H. Devoret, and J. Clarke, Energy-Level Quantization in the Zero-Voltage State of a Current-Biased Josephson Junction, Phys. Rev. Lett. 55, 1543–1546 (1985).
- [223] Y. Nakamura, Y. A. Pashkin, and J. S. Tsai, Coherent control of macroscopic quantum states in a single-Cooper-pair box, Nature **398**, 786–788 (1999).
- [224] D. P. DiVincenzo, Fault-tolerant architectures for superconducting qubits, Phys. Scr. T137, 014020 (2009).
- [225] J. M. Martinis, S. Nam, J. Aumentado, and C. Urbina, Rabi Oscillations in a Large Josephson-Junction Qubit, Phys. Rev. Lett. 89, 117901 (2002).
- [226] J. E. Mooij, T. P. Orlando, L. Levitov, L. Tian, C. H. van der Wal, and S. Lloyd, Josephson Persistent-Current Qubit, Science 285, 1036–1039 (1999).

- [227] I. M. Pop, K. Geerlings, G. Catelani, R. J. Schoelkopf, L. I. Glazman, and M. H. Devoret, Coherent suppression of electromagnetic dissipation due to superconducting quasiparticles, Nature 508, 369–372 (2014).
- [228] R. Barends et al., Coherent Josephson Qubit Suitable for Scalable Quantum Integrated Circuits, Phys. Rev. Lett. 111, 080502 (2013).
- [229] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, A quantum engineer's guide to superconducting qubits, Appl. Phys. Rev. 6, 021318 (2019).
- [230] S. Rasmussen, K. Christensen, S. Pedersen, L. Kristensen, T. Bækkegaard, N. Loft, and N. Zinner, Superconducting Circuit Companion—an Introduction with Worked Examples, PRX Quantum 2, 040204 (2021), arXiv:2103.01225.
- [231] U. Vool and M. H. Devoret, Introduction to Quantum Electromagnetic Circuits, Int. J. Circ. Theor. Appl. 45, 897–934 (2017), arXiv:1610.03438.
- [232] R.-P. Riwar and D. P. DiVincenzo, Circuit quantization with time-dependent magnetic fields for realistic geometries, npj Quantum Inf 8, 1–12 (2022), arXiv:2103. 03577.
- [233] X. You, J. A. Sauls, and J. Koch, Circuit quantization in the presence of timedependent external flux, Phys. Rev. B 99, 174512 (2019), arXiv:1902.04734.
- [234] S. E. Nigg, H. Paik, B. Vlastakis, G. Kirchmair, S. Shankar, L. Frunzio, M. H. Devoret, R. J. Schoelkopf, and S. M. Girvin, Black-Box Superconducting Circuit Quantization, Phys. Rev. Lett. 108, 240502 (2012).
- [235] M. H. Devoret, Quantum Fluctuations in Electrical Circuits, in Proc. Houches Summer Sch. Sess. LXIII (1997).
- [236] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, Charge-insensitive qubit design derived from the Cooper pair box, Phys. Rev. A 76, 042319 (2007).
- [237] H. Paik et al., Observation of High Coherence in Josephson Junction Qubits Measured in a Three-Dimensional Circuit QED Architecture, Phys. Rev. Lett. 107, 240501 (2011).
- [238] M. H. Devoret and R. J. Schoelkopf, Superconducting Circuits for Quantum Information: An Outlook, Science 339, 1169–1174 (2013).
- [239] D. F. Walls and G. J. Milburn, Quantum Optics, 2nd ed. (Springer-Verlag, Berlin Heidelberg, 2008).
- [240] W. Schleich, Quantum Optics in Phase Space (Wiley-VCH, Berlin; New York, 2001).
- [241] W. Wustmann and V. Shumeiko, Parametric resonance in tunable superconducting cavities, Phys. Rev. B 87, 184501 (2013).
- [242] A. Vrajitoarea, Z. Huang, P. Groszkowski, J. Koch, and A. A. Houck, Quantum control of an oscillator using a stimulated Josephson nonlinearity, Nat. Phys. 16, 211–217 (2020).
- [243] R. Lescanne, M. Villiers, T. Peronnin, A. Sarlette, M. Delbecq, B. Huard, T. Kontos, M. Mirrahimi, and Z. Leghtas, Exponential suppression of bit-flips in a qubit encoded in an oscillator, Nat. Phys. 16, 509–513 (2020).
- [244] H. J. Carmichael, Statistical Methods in Quantum Optics 2, Theoretical and Mathematical Physics (Springer, Berlin, Heidelberg, 2008).

- [245] F. Reiter and A. S. Sørensen, Effective operator formalism for open quantum systems, Phys. Rev. A 85, 032111 (2012).
- [246] J. R. Schrieffer and P. A. Wolff, Relation between the Anderson and Kondo Hamiltonians, Phys. Rev. 149, 491–492 (1966).
- [247] J. M. Luttinger and W. Kohn, Motion of Electrons and Holes in Perturbed Periodic Fields, Phys. Rev. 97, 869–883 (1955).
- [248] F. Wegner, Flow-equations for Hamiltonians, Ann. Phys. 506, 77–91 (1994).
- [249] J. Venkatraman, X. Xiao, R. G. Cortiñas, A. Eickbusch, and M. H. Devoret, Static Effective Hamiltonian of a Rapidly Driven Nonlinear System, Phys. Rev. Lett. 129, 100601 (2022).
- [250] N. E. Frattini, U. Vool, S. Shankar, A. Narla, K. M. Sliwa, and M. H. Devoret, 3-wave mixing Josephson dipole element, Appl. Phys. Lett. 110, 222603 (2017).
- [251] H.-K. Lin and L. P. Pryadko, Quantum two-block group algebra codes, Phys. Rev. A 109, 022407 (2024), arXiv:2306.16400 [math-ph, physics:quant-ph].
- [252] A. Kubica and N. Delfosse, Efficient color code decoders in $d \ge 2$ dimensions from toric code decoders, Quantum 7, 929 (2023).
- [253] B. J. Brown, Conservation Laws and Quantum Error Correction: Toward a Generalized Matching Decoder, IEEE BITS Inf. Theory Mag. 2, 5–19 (2022), arXiv:2207.06428 [cond-mat, physics:quant-ph].
- [254] K. Sahay and B. J. Brown, Decoder for the Triangular Color Code by Matching on a Möbius Strip, PRX Quantum 3, 010310 (2022).
- [255] C. Piveteau, C. T. Chubb, and J. M. Renes, Tensor-Network Decoding Beyond 2D, PRX Quantum 5, 040303 (2024), arXiv:2310.10722 [quant-ph].
- [256] M. S. Alam, J. Chen, and T. R. Scruby, Bacon-Shor Board Games, Apr. 2025, arXiv:2504.02749 [quant-ph].
- [257] M. McEwen, D. Bacon, and C. Gidney, Relaxing Hardware Requirements for Surface Code Circuits using Time-dynamics, Quantum 7, 1172 (2023), arXiv:2302. 02192 [quant-ph].
- [258] H. Bombin, D. Litinski, N. Nickerson, F. Pastawski, and S. Roberts, Unifying flavors of fault tolerance with the ZX calculus, Quantum 8, 1379 (2024), arXiv:2303.08829 [quant-ph].
- [259] C. Gidney and D. Bacon, Less Bacon More Threshold, May 2023, arXiv:2305.12046 [quant-ph].

Paper I

Universal Gate Set for Continuous-Variable Quantum Computation with Microwave Circuits

Paper II

Designing Kerr Interactions for Quantum Information Processing via Counterrotating Terms of Asymmetric Josephson-Junction Loops

Paper III

Universal control of a bosonic mode via drive-activated native cubic interactions
Paper IV

Performance of Teleportation-Based Error-Correction Circuits for Bosonic Codes with Noisy Measurements

Paper V

Quantum error correction with dissipatively stabilized squeezed-cat qubits

Paper VI

Analog Information Decoding of Bosonic Quantum Low-Density Parity-Check Codes

Paper VII

Localized statistics decoding: A parallel decoding algorithm for quantum low-density parity-check codes

Paper VIII

High-threshold, low-overhead and single-shot decodable fault-tolerant quantum memory

Paper IX

Single-shot and measurement-based quantum error correction via fault complexes