Technische Universität München

TUM School of Computation, Information and Technology

Quantum Fault-Tolerance with Continuous Variable Systems

Margret Hildegard Heinze

Vollständiger Abdruck der von der TUM School of Computation, Information and Technology der Technischen Universität München zur Erlangung des akademischen Grades einer

Doktorin der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitz:	Prof. Dr. Simone Warzel
Prüfer:innen der Dissertation:	
	1. Prof. Dr. Robert König
	2. Prof. Dr. Liang Jiang

Die Dissertation wurde am 30.11.2022 bei der Technischen Universität München eingereicht und durch die TUM School of Computation, Information and Technology am 28.03.2023 angenommen.

Zusammenfassung

Fehler aufgrund von Dekohärenz sind eine zentrale Hürde auf dem Weg zu zuverlässigen Quantencomputern. Diese Dissertation befasst sich mit experimentell zugänglichen Fehlertoleranzprotokollen in Quantensystemen mit kontinuierlichen Variablen (CV), welche in Anwendungen omnipräsent sind. Wir konstruieren neue Pulssequenzen, die ungewollte System-Umgebungs-Interaktionen unterdrücken, und untersuchen neue Codierungen von Quantenbits in CV-Systemen mit verbesserter Fehlertoleranz.

Abstract

Errors caused by decoherence are a major obstacle on the road towards robust quantum computation. This dissertation treats experimentally feasible fault-tolerance protocols in the context of continuous variable (CV) quantum systems which are ubiquitous in applications. We construct novel pulse sequences to suppress unwanted system-environment interactions and investigate new encodings of quantum bits into CV systems with improved noise tolerance.

Acknowledgements

First and foremost, I want to thank Robert König: Thank you for giving me the opportunity to work on such an exciting research topic. As my supervisor you shaped this work and my development during the last years. I appreciate the freedom and trust you gave me on the one hand and the support on the other. You were always easy-to-reach, answering questions and proof-reading impressively quickly, giving advice and much more. You dedicated a lot of time, ideas and effort to me and your other doctoral researchers: I am deeply grateful for your commitment.

I also want to express a heartfelt thanks to all the wonderful members of the M5 group at TUM; to Michael Wolf for leading this group and creating this enjoyable atmosphere, to Silvia Schulz for being the best support I can think of in all aspects of daily university life and to my colleagues, Andreas Bluhm, Anna-Lena Hashagen, Martina Gschwendtner, Stefan Huber, Alexander Kliesch, Matthias Caro, Lisa Hänggli, Daniel Stilck-Franca, Javier Cuesta, Cambyse Rouzé, Angela Capel, Markus Hasenöhrl, Yifan Jia and Vjosa Blakaj. You filled my past years at TUM with life: I will remember our entertaining coffee breaks, the large number of cakes to celebrate (any) achievement, our joint activities, that you were always willing to help out in case of need, that you encouraged and supported me in difficult situations and I loved to go on conferences with you. I also want to thank the people I had the pleasure to work with, i.e., Robert, Lisa, Martina and Michael (Keyl), for your ideas, your patience, discussions and everything you taught me. These collaborations have always been a great motivator for me.

The IMPRS QST did not only fund my PhD but also offered an exceptional qualification programme with summer schools, workshops, excursions, talks and lectures. I am grateful for these opportunities and for all the inspiring conversations with my fellow doctoral researches, and to our very dedicated coordinator Sonya Gzyl.

I am very lucky to have been embedded in such a supportive environment, so thanks go to the mathematics department of TUM (Simone Warzel, the deans office with Lydia Weber, the Fakultätsrat, the FACETS of mathematics team, and many more), the IMPRS QST, the MCQST for their career support, Thomas Schulte-Herbrüggen for integrating me into the ExQM programme, and to my master thesis supervisor and TUM mentor Michael Keyl. I am especially thankful to Isabella Wiegand: you have been the best supporter and advisor I can imagine and the heart and soul of ISAM.

My doctoral 'twin' Alex, and my friends Rita and Christian helped me a lot during the final phase before submission, e.g. by proof-reading or by hosting me for a week.

I was always motivated by the work as a students' or doctoral representative and I enjoyed working with my fellow students in the Physik-Fachschaft Frankfurt, with the members of ZaPF at 13 unforgettable conferences and with my fellow doctoral representatives at TUM and IMPRS.

Of course, besides work, the central support during the last years were my friends and family, from Munich, Frankfurt, Bonn and beyond. Without you, I would not have been able to write this thesis. I am so grateful to have you in my life, especially to the six of you, Christian, Lia, Rita, Lennart, and my parents, Elisabeth and Johannes.

Contents

1 Introduction

Quantum computing is a new paradigm for information-processing. It potentially provides novel, vastly more efficient ways of solving certain computational problems compared to what classical computers can achieve. Central to this paradigm are new information-processing primitives that build on quantum-mechanical effects such as non-locality. These can be observed if a quantum system can be manipulated reliably. In this sense, quantum computers exploit physical properties which are fundamentally different from classical computers: they build on the fact that the underlying physical system which carries the information is governed by quantum mechanics. The motivation for considering this approach is two-fold: First, such systems are by design more suitable to address computational problems associated with quantum-mechanical systems such as studying the properties and dynamics of e.g., molecules or new materials [63]. Second, there is evidence that certain computational problems of algebraic and combinatorial nature, such as factoring integers or search problems, can be solved more efficiently with quantum computers [48, 90, 72, 121]. In both cases, there is hope that the computational reach of quantum computers extends beyond classical computers in problems of practical relevance.

A crucial issue on the road towards building a quantum computer is the fact that manipulating quantum information in a robust manner is difficult: Inevitable and unwanted interactions with the environment typically lead to a rapid loss of quantum properties such as coherence and entanglement. Furthermore, limited control capabilities may restrict the set of operations that can be performed reliably, including state preparation, gate application and measurement. For these reasons, it is crucial to develop suitable fault-tolerance protocols and mechanisms to protect quantum systems against noise.

Quantum computing is often considered based on finite-dimensional systems, where the basic building block is a quantum bit, a two-level quantum system that replaces the notion of a classical bit and is therefore called a qubit. Similar to the way that continuous parameters of a physical system (such as voltage or magnetisation) are discretised in digital classical computers (e.g., to encode bit strings), the consideration of multi-qubit systems simplifies and unifies the search for algorithms in a device-independent fashion. In practice, this amounts to e.g., only using two energy levels of an atom or the two polarisation states of a photon. Error-correction techniques for multi-qubit systems have a long history, and show that – in principle – reliable and scalable quantum computers can be built based on imperfect components [2, 97, 92]. However, identifying the most resource-efficient procedures for achieving fault-tolerance remains a key research topic at the intersection of theory and experiment.

For fault-tolerance with qubits, several distinct approaches are being pursued. On the one hand, one attempts to engineer specific dynamics by e.g., applying pulse sequences, to suppress the effect of unwanted system-environment interactions. These techniques are well-established both theoretically and experimentally, and successfully reduce effective error rates.

On the other hand, the theory of quantum error-correcting codes aims to use redundant encodings to distribute relevant quantum information in many-body systems. Typically, corresponding proposed schemes provide increasing reliability with growing system size, provided that the initial physical noise level is below a certain threshold. For qubit systems, numerous families of quantum codes with various parameters and associated features are known.

While the abstraction of qubits is useful for theoretical considerations, many quantum systems used in practice are so-called continuous variable (CV) or bosonic quantum systems associated with a (countably) infinite number of energy levels. Such systems are omnipresent in nature since many physical quantities, such as the amplitudes of the electromagnetic field or position and momentum, are continuous. Correspondingly, they are also of high practical relevance since experimental techniques for their control are well developed in many systems. This motivates the development of fault-tolerance mechanisms that take the CV nature of such systems into account. In particular, this can result in more robust and thus resource-efficient approaches to quantum computation, and provide recipes that are more in line with available experimental capabilities. Unfortunately, however, the theory of quantum fault-tolerance with CV systems is much less developed compared to the multi-qubit setting.

This thesis contributes towards closing this gap between the present understanding of CV versus discrete quantum systems in the context of quantum fault-tolerance. Both approaches mentioned above are pursued. The main advances can be summarised as follows:

Novel dynamical decoupling protocols for continuous variable (CV) systems (a): We construct pulse sequences that combat dominant decoherence effects on CV quantum systems. The pulse sequences achieve the following: First, they approximately decouple the system from its environment, i.e., the effective evolution which is generated by rapidly applying pulses on the system at certain times and evolving under the uncontrolled decoherence Hamiltonian in between (approximately) involves no interaction between the system and the environment. Second, the system part of the effective evolution is approximately *homogenised*, i.e., reduced to that of non-interacting harmonic oscillators all rotating at the same averaged frequency. By 'approximately', we mean that these goals can be achieved up to any desired order in the total time of the pulse sequence. This homogenised evolution provides natural decoherence-free subspaces to encode quantum information. A remarkable feature of our constructed protocols is their simplicity and the fact that they are highly resource-efficient: They use no a priori knowledge on the exact form of the decoherence Hamiltonian and no feedback via measurements, i.e., the same pulse sequence works for all Hamiltonians that are quadratic in the system and environment mode operators. They hence target a wide range of practically relevant noise models such as thermal noise or photon loss. Furthermore, the involved pulses are very simple, i.e., products of single-mode phase rotations and beam splitters, and their number only scales polynomially in the suppression order. This scaling constitutes an exponential improvement over what can be achieved by concatenating previously considered first-order pulse sequences. To our knowledge, our pulse sequences are the first proposals for higher order bosonic decoupling or homogenisation.

A new approach to encoding qubits into CV systems (b): We propose a new protocol to robustly encode a logical qubit using several harmonic oscillator modes. The construction is based on the combination of two well-explored quantum error correcting codes, a qubit-into-CV encoding which is concatenated with a qubit-into-qubits encoding. The novelty of our approach relies on an additional squeezing unitary in the encoding procedure which has not been considered before. As the main advance of

this approach, we find that the introduction of this additional squeezing leads to an increased fault-tolerance threshold against random Gaussian phase space displacements. The encoded qubit is reliably protected from such noise with shift-error standard deviation up to $\sigma \approx 0.581$ in the limit of large code sizes compared to the state-of-the art threshold of $\sigma \approx 0.540$ which was achieved in similar constructions without the squeezing [169, 68]. At the cost of an additional squeezing of 4.77 dB, one can achieve this improvements in the noise tolerance.

To achieve these goals, we use the following methods.

For the first advance **(a)**, we relate multi-qubit systems to multi-mode CV system where the Hamiltonians are at most quadratic in the mode operators. More precisely, we establish a connection between the corresponding Lie groups, i.e., the unitary and the symplectic group, and similarly between the corresponding Lie algebras. This connection enables us to formally prove the validity of our CV pulse sequences by generalising the universality proof of the (nested) Uhrig dynamical decoupling scheme [86] to the CV setting. As a consequence, we can lift the most efficient multi-qubit dynamical decoupling schemes [161] – most efficient in the sense that they require a minimal number of pulses for a given suppression order – to the CV setting: while keeping the pulse timings of the multi-qubit protocols we replace the qubit operations by passive Gaussian unitaries. We note that our analysis is based on the fact that we work within the framework of symplectic matrices on the CV phase space. This is possible since – by assumption on the decoherence Hamiltonian – all considered unitaries are Gaussian.

For the second advance **(b)**, our encodings are constructed from the concatenation of two well-explored codes: using the surface code [30, 67, 47, 92], a logical qubit is encoded into many qubits each of which is again encoded into a CV mode using an asymmetric version of the Gottesman-Kitaev-Preskill (GKP) code [71]. We investigate our novel qubit-into-CV encodings employing analytical as well as numerical methods. For the latter, we simulate the quantum error correction process on a classical computer for different noise strengths: an efficient algorithm simulates an efficient surface code decoder [29] that uses a tensor network approximation. A Monte Carlo study yields the above mentioned estimates for noise thresholds. We emphasise that an asymmetric encoding of the GKP-qubits – using a squeezing unitary – is essential to the improved noise tolerance of our encodings. The key idea is that this squeezing renders the effective noise on the logical level of the GKP-qubits biased. The concatenated code can hence benefit from the enhanced resilience of the surface code against biased noise [158]. In this sense, our analysis conceptually illustrates that it can be beneficial to actively engineer a noise bias and benefit from the resilience of the surface code towards biased noise. We remark that throughout, we consider an idealised setup where no errors occur during the syndrome measurements of the error correction process.

Both advances **(a)** and **(b)** showcase the usefulness of connecting quantum fault-tolerance protocols in the qubit and the CV setting. In both cases, we only employ additional Gaussian resources to enhance the noise tolerance of the considered protocols: passive Gaussian pulses in the novel pulse sequences and an a Gaussian squeezing operation in the context of quantum error correcting codes. We note that such Gaussian resources are considered more feasible, e.g., realisable by passive or active linear optics, but also more limited in terms of their usefulness for quantum information processing. However here we show that such resources can be highly beneficial when combined with other concepts or resources. Similar constructions may also be interesting and beneficial for other protocols. Our results serve as a proof of principle; they exploits the enhanced noise tolerance of a given code by concatenation. This circumvents well-known no-go results on Gaussian CV-into-CV encoding [69, 56, 123, 169].

Outline of the thesis

The first two chapters of the main text are dedicated to the fundamentals: Chapter 2 presents the basic properties of quantum mechanics, bosonic quantum systems and quantum computing which are relevant to the work at hand. In Chapter 3, we discuss fundamentals of quantum fault-tolerance, i.e., quantum noise processes in the qubit as well as in the CV setting and error correction. We also discuss error suppression techniques. The two central results of this thesis on *dynamical decoupling for CV systems* and on *qubit-into-CV encodings* are presented on Chapters 4 and 5, respectively. The first one introduces the novel DD protocols for bosonic systems while the second one constructs an improvement on a specific DV-into-CV encoding. We conclude in Chapter 6 with a summary of our results and an outlook.

2 Fundamentals of quantum computing

Whereas a classical computer uses the classical states 0 and 1 as its basic data bits, a quantum computer uses quantum states. To consider the latter, it is therefore necessary to introduce the basic mathematical formalism which describes quantum mechanics: this is done in Section 2.1. The quantum systems considered in this thesis are so-called continuous variable (CV) or bosonic quantum systems where the basic information carriers are not two-level system, but bosonic quantum modes, i.e., systems with infinitely many levels. Section 2.2 presents the basic properties of such CV systems. In Section 2.3, we introduce the basics of the circuit model of quantum computation which uses quantum bits.

2.1 Introduction to quantum mechanics

The mathematical description of quantum systems relies on expressing quantum states and operations as elements of or operators on a Hilbert space. This section briefly recaps the central concepts of this description which are relevant to the work at hand. In Section 2.1.1, we set the stage by introducing Hilbert spaces and operators. The concepts of states, observables and evolutions are described in Sections 2.1.2 and 2.1.5, respectively. In Section 2.1.3, we illustrate the above concepts in terms of the simplest quantum system – the quantum bit – which despite its simplicity is of great importance, e.g., for quantum computation. The mathematical description of several composite quantum systems is given in Section 2.1.4 where we also elaborate on the related concept of entanglement.

2.1.1 Hilbert spaces and operators: basic definitions and notation

In quantum mechanics, one associates a complex separable Hilbert space $\mathcal{H}$ to every quantum system. A complex Hilbert space $\mathcal{H}$ is a complex inner product space $(\mathcal{H}, \langle\cdot,\cdot\rangle)$ which additionally is a complete metric space with respect to the norm $\|x\| \coloneqq \sqrt{\langle x, x\rangle}$ induced by the inner product. Note that we use the convention that the inner product $\langle\cdot,\cdot\rangle$ is antilinear in the first and linear in the second argument. Separability means that $\mathcal{H}$ admits a countable orthonormal basis. Let furthermore $\mathcal{H}^*$ denote the set of continuous, linear functionals on $\mathcal{H}$. In the following, we restrict our attention to complex separable Hilbert spaces unless mentioned otherwise. Note that in the context of quantum field theory, an alternative formulation using C* algebras of observables as a starting point is used [26]. However, for the concepts discussed in this thesis, the Hilbert space formulation is sufficient.

Braket-Notation

The *braket or Dirac notation* is a useful tool and prominent in physics: An element $\psi \in \mathcal{H}$ corresponds to the so-called *ket-vector* $|\psi\rangle$ whereas the bra-vector $\langle\psi| \in \mathcal{H}^*$ symbolises the corresponding dual element. Then the inner product $\langle\psi, \varphi\rangle$ between two vectors $\psi, \varphi \in \mathcal{H}$ can be written in braket-notation as $\langle\psi|\varphi\rangle$. Furthermore, for $\psi, \varphi \in \mathcal{H}$, the braket notation $|\psi\rangle\langle\varphi|$ stands for the linear map which acts on an element $\vartheta \in \mathcal{H}$ as $\vartheta \mapsto |\psi\rangle \langle\varphi|\vartheta\rangle := \langle\varphi, \vartheta\rangle\psi$. Given a linear map $A : \mathcal{H} \to \mathcal{H}$ one writes $\langle\varphi|A|\psi\rangle$ to denote $\langle\varphi, A\psi\rangle$.

Operators on Hilbert spaces

An *operator*[1] A on a Hilbert space $\mathcal{H}$ is a linear map $A : \mathcal{H} \to \mathcal{H}$. It is *bounded* if its operator norm $\|A\| := \sup_{\psi\in\mathcal{H}} \|A\psi\|/\|\psi\|$ is finite. We denote the set of operators on $\mathcal{H}$ by $\mathcal{L}(\mathcal{H})$ and those of bounded operators by $\mathcal{B}(\mathcal{H})$. The *adjoint* $A^\dagger$ of an operator $A \in \mathcal{B}(\mathcal{H})$ is defined via

$$\langle\psi, A\phi\rangle = \langle A^\dagger\psi, \phi\rangle \qquad \text{for all } \psi, \phi \in \mathcal{H} \ . \tag{2.1}$$

Certain subsets of $\mathcal{B}(\mathcal{H})$ are central to the description of quantum systems.

Definition 2.1 *(Operators on Hilbert spaces).* Let $\mathcal{H}, \mathcal{H}_1, \mathcal{H}_2$ be Hilbert spaces.

An operator $A \in \mathcal{B}(\mathcal{H})$ is called *self-adjoint* if it satisfies $A = A^\dagger$. Let

$$\mathcal{B}_{sa}(\mathcal{H}) := \{A \in \mathcal{B}(\mathcal{H}) \mid A = A^\dagger\}$$

denote the set of all self-adjoint operators.

A map $U \in \mathcal{B}(\mathcal{H})$ is called a *unitary* if $UU^\dagger = U^\dagger U = I$ where I is the identity map on $\mathcal{H}$. The set of unitary operators on $\mathcal{H}$ is denoted by $\mathsf{U}(\mathcal{H})$.

A linear map $V : \mathcal{H}_1 \to \mathcal{H}_2$ is called an *isometry* if for all $\varphi, \psi \in \mathcal{H}_1$, $\langle\varphi, \psi\rangle = \langle V\varphi V\Psi\rangle$.

A map $U \in \mathcal{B}(\mathcal{H})$ is called a *partial isometry* if its restriction onto the orthogonal complement of $\ker(U)$ is an isometry, i.e., for all $\psi \in \ker(U)^\perp$ we have $\|U\psi\| = \|\psi\|$.

A map $A \in \mathcal{B}(\mathcal{H})$ is called *positive* if $\langle\psi, A\psi\rangle \geq 0$ for all $\psi \in \mathcal{H}$.

A map $P \in \mathcal{B}_{sa}(\mathcal{H})$ is called a *projection* if $P^2 = P$. The set of projections on $\mathcal{H}$ is denoted by $\mathcal{P}(\mathcal{H})$.

Note that the above definition directly implies that every projection is positive and that every positive operator is self adjoint. We write $A \geq 0$ to denote that A is a positive operator.

A complex number $\lambda \in \mathbb{C}$ belongs to the *spectrum* of a bounded operator A if $A - \lambda I$ is not a bijection with a bounded inverse. The *trace* of a positive operator $T \in \mathcal{B}(\mathcal{H})$ is defined by $\mathrm{tr}(T) := \sum_{k=1}^\infty \langle e_k|Te_k\rangle$ where $\{|e_k\rangle\}_{k\in\mathbb{N}}$ is a countable orthonormal basis of $\mathcal{H}$ and one sets $\mathrm{tr}(T) = \infty$ if $\sum_{k=1}^\infty \langle e_k|Te_k\rangle$ does not converge. By separability of $\mathcal{H}$, such a countable basis exists and the value of the trace is independent of the chosen basis (cf. e.g. [139, Theorem VI.18] for a proof).

[1] More generally, the domain and the range may be distinct Hilbert spaces: An operator A is a linear map $A : \mathcal{H}_1 \to \mathcal{H}_2$ for two Hilbert spaces $\mathcal{H}_1$ and $\mathcal{H}_2$. We denote the set of all operators from $\mathcal{H}_1$ to $\mathcal{H}_2$ by $\mathcal{L}(\mathcal{H}_1, \mathcal{H}_2)$ and use $\mathcal{L}(\mathcal{H})$ as short hand notation for $\mathcal{L}(\mathcal{H}, \mathcal{H})$.

Furthermore, we denote the square root of a positive operator $A \in \mathcal{B}(\mathcal{H})$ by $\sqrt{A}$ [2]. Since for every $A \in \mathcal{B}(\mathcal{H})$, $A^\dagger A$ is positive by Eq. (2.1), the operator $|A| := \sqrt{A^\dagger A}$ is well-defined.

Definition 2.2. The set of *trace-class operators* is defined as

$$\mathcal{T}(\mathcal{H}) := \{A \in \mathcal{B}(\mathcal{H}) \mid \operatorname{tr}(|T|) := \operatorname{tr}(\sqrt{T^\dagger T}) < \infty\} .$$

The sets we have introduced so far satisfy the following inclusion relations:

$$\mathcal{P}(\mathcal{H}) \subset \mathcal{B}_{sa}(\mathcal{H}) \subset \mathcal{B}(\mathcal{H}) \subset \mathcal{L}(\mathcal{H}) \quad , \quad \mathcal{T}(\mathcal{H}) \subset \mathcal{B}(\mathcal{H}) \quad , \quad \mathsf{U}(\mathcal{H}) \subset \mathcal{B}(\mathcal{H}) .$$

Let us add a few remarks on the differences between a finite-dimensional and an infinite-dimensional Hilbert space: If $\mathcal{H}$ is finite-dimensional, then $\mathcal{H} \cong \mathbb{C}^n$ for some $n \in \mathbb{N}$ and the linear operators on $\mathcal{H}$ can be represented by complex $n \times n$-matrices, implying that all linear operators on a finite-dimensional Hilbert space are bounded; this statement is no longer true if $\mathcal{H}$ is infinite-dimensional: operators may be unbounded.

Adaptions for infinite-dimensional Hilbert spaces

In the case of an infinite-dimensional Hilbert space, some of the above definitions are impractical – the assumptions are too strict for many purposes – and it is useful to adapt and relax certain of these. A mathematical precise description can be found in textbooks, e.g. by Simon and Reed [139].

The first adaptions concern *unbounded operators* since many physically interesting observables such as position and momentum of the quantum harmonic oscillator correspond to unbounded operators. By the Hellinger-Toeplitz theorem, an operator $A : \mathcal{H} \to \mathcal{H}$ that satisfies $\langle\varphi|A\psi\rangle = \langle A\varphi|\psi\rangle$ for all $|\psi\rangle, |\varphi\rangle \in \mathcal{H}$ must necessarily be bounded (cf. e.g. [Chapter III.5][139]). Hence one makes the following adaptions for self-adjoint operators.

Definition 2.3 *(Adjoint and self-adjointness).* A *densely defined operator* A is a linear map $A : D(A) \to \mathcal{H}$, where $D(A)$ is a dense linear subspace of $\mathcal{H}$ called the *domain* of the operator.

The *(Hilbert) adjoint* $A^\dagger$ of a densely defined operator $A : D(A) \to \mathcal{H}$ is defined by

$$\langle\psi|A\varphi\rangle = \langle A^\dagger\psi|\varphi\rangle \qquad \text{for all } |\varphi\rangle \in D(A) \, , |\psi\rangle \in D(A^\dagger)$$

where the domain of $A^\dagger$ is given by all $|\psi\rangle \in \mathcal{H}$ such that the map $|\varphi\rangle \mapsto \langle\psi|A\varphi\rangle$ is a (densely defined) continuous linear functional.

A densely defined operator $A : D(A) \to \mathcal{H}$ is *symmetric* if $\langle A\psi|\varphi\rangle = \langle\psi|A\varphi\rangle$ for all $|\psi\rangle, |\varphi\rangle \in D(A)$.

Furthermore $A : D(A) \to \mathcal{H}$ is *self-adjoint* if A is symmetric and $D(A^\dagger) = D(A)$.

A densely defined operator is specified by the action on its domain by extending the action from the dense subspace to the whole space using linearity. In slight abuse of notation, the domain is often omitted in its definition.

[2] By the square root Lemma (cf. e.g. [81]) such an operator exists, is unique and satisfies $(\sqrt{A})^2 = A$.

Note that a symmetric operator A with $D(A) = \mathcal{H}$ is self-adjoint and bounded. If $D(A) \neq \mathcal{H}$, then a self-adjoint operator does not need to be bounded any more. With this in mind, one can formally replace $\mathcal{B}_{sa}(\mathcal{H})$ by

$$\mathcal{L}_{sa}(\mathcal{H}) = \{A : D(A) \to \mathcal{H} \mid A \text{ self-adjoint}, \overline{D(A)} = \mathcal{H}\} \ .$$

Note that it is more subtle to define sums or products of densely defined operators A and B than for operators defined on the entire Hilbert space: for example, $D(A+B) \subset D(A) \cap D(B)$ might only include the zero vector; hence the set $\mathcal{L}_{sa}(\mathcal{H})$ will no longer be closed under addition. Of special interest in quantum mechanics is the spectrum of self-adjoint operators which is defined as follows: A number $\lambda \in \mathbb{C}$ belongs to the spectrum of an unbounded self-adjoint operator $A : D(A) \to \mathcal{H}$ if $A - \lambda I$ is not a bijection of $D(A)$ onto $\mathcal{H}$ with bounded inverse. If an element λ of the spectrum satisfies $A|\psi\rangle = \lambda|\psi\rangle$ for some $|\psi\rangle \in D(\mathcal{H})$, then it is called an eigenvalue and $|\psi\rangle$ the corresponding eigenvector (cf. [139, Chapter VIII.1]). The spectral theorem for unbounded self-adjoint operators.

The second adaption concerns the topology on operator spaces, e.g., on $\mathcal{L}(\mathcal{H})$ and subsets thereof. The *standard* or *norm topology* is defined by the following notion of convergence: A sequence of operators $(A_n)_{n\in\mathbb{N}} \subset \mathcal{B}(\mathcal{H})$ converges to $A \in \mathcal{B}(\mathcal{H})$ with respect to the *norm topology* if and only if

$$\lim_{n\to\infty} \|A_n - A\| \coloneqq \lim_{n\to\infty} \sup_{\|\psi\|=1} \|A_n\psi - A\psi\| = 0 \ .$$

However, the norm topology (determined by the operator norm) is too strong for many applications, e.g., when considering closures of subsets of the unitary operators. The two other topologies that are relevant for most applications in physics are the *weak operator topology* and the *strong operator topology*.

A sequence of operators $(A_n)_{n\in\mathbb{N}} \subset \mathcal{B}(\mathcal{H})$ converges to $A \in \mathcal{B}(\mathcal{H})$ *weakly* (i.e., with respect to the weak operator topology) if and only if

$$\lim_{n\to\infty} |\langle\varphi|A_n\psi\rangle - \langle\varphi|A\psi\rangle| = 0 \qquad \text{for all } \varphi, \psi \in \mathcal{H} \ .$$

A sequence of operators $(A_n)_{n\in\mathbb{N}} \subset \mathcal{B}(\mathcal{H})$ converges to $A \in \mathcal{B}(\mathcal{H})$ with respect to the *strong operator topology* if and only if

$$\lim_{n\to\infty} \|A_n\psi - A\psi\| = 0 \qquad \text{for all } \psi \in \mathcal{H} \ .$$

Note that (if $\mathcal{H}$ is infinite-dimensional) the weak operator topology is (strictly) weaker than the strong operator topology which again is (strictly) weaker than the norm topology[3].

2.1.2 Quantum states, observables and measurements

Density matrices and states

Pure states are rays in $\mathcal{H}$, i.e., in one-to-one correspondence with equivalence classes of unit vectors up to a global phase $e^{i\vartheta} \in \mathbb{C}$. In slight abuse of notation, we will usually neglect this phase and simply say $|\psi\rangle \in \mathcal{H}$ is a pure state if it has unit norm, and that $|\psi\rangle$ and $|\varphi\rangle$ describe the same state if $|\psi\rangle = e^{i\vartheta}|\varphi\rangle$ for some $\vartheta \in \mathbb{R}$.

There exists a useful generalisation of pure states: the density operators.

[3]Given two topologies $\mathcal{T}_1$, $\mathcal{T}_2$ on the same set, $\mathcal{T}_1$ is weaker than $\mathcal{T}_2$ if every sequence that converges with respect to $\mathcal{T}_2$ also converges with respect to $\mathcal{T}_1$.

Definition 2.4 *(Density operator).* A general state or *density operator* (or *density matrix*) is a positive unit trace operator $\rho \in \mathcal{B}(\mathcal{H})$. The set of density operators of a quantum system with associated Hilbert space $\mathcal{H}$ is given by

$$\mathcal{D}(\mathcal{H}) \coloneqq \{\rho \in \mathcal{B}(\mathcal{H}) \mid \rho \in \mathcal{B}(\mathcal{H}),\ \rho \geq 0,\ \mathrm{tr}(\rho) = 1\}\ .$$

Density matrices generalise the concept of pure states in the following sense: on the one hand, every pure state $|\psi\rangle \in \mathcal{H}$ corresponds to a density operator given by $\rho = |\psi\rangle\langle\psi|$, i.e., a pure state's density operator is the one-dimensional projection onto the subspace spanned by the pure state. On the other hand, every density matrix is the convex combination of one-dimensional projections: More precisely[4], every $\rho \in \mathcal{D}(\mathcal{H})$ can be written as

$$\rho = \sum_{i \in I} \lambda_i |\psi_i\rangle\langle\psi_i| \qquad \text{for } 0 \leq \lambda_i \leq 1, \quad \sum_i \lambda_i = 1\ , \tag{2.2}$$

for some $I \subset \mathbb{N}$. If $I = \mathbb{N}$, i.e., if there is a countably infinite number of summands λ_i, then the sum in (2.2) converges with respect to the trace norm $\|\rho\|_{tr} \coloneqq \mathrm{tr}(|\rho|)$. As a consequence, the set of density operators $\mathcal{D}(\mathcal{H})$ is convex and pure states correspond to extreme elements of the convex set $\mathcal{D}(\mathcal{H})$. The other elements are statistical mixtures of pure states and therefore called *mixed states*.

Note that there is also a different notion of quantum states:

Definition 2.5 *(Quantum state).* A *quantum state* is a linear functional $\omega : \mathcal{B}(\mathcal{H}) \to \mathbb{C}$ that is normalised, i.e., $\omega(I) = 1$, positive, i.e., $\omega(A) \geq 0$ for any positive operator A, and normal, i.e., if a sequence $(A_n)_{n\in\mathbb{N}} \subset \mathcal{B}(\mathcal{H})$ of increasing converges strongly to A, then $\lim_{n\to\infty} \omega(A_n) = \omega(A)$.

This is equivalent to Definition 2.4 by the following theorem.

Theorem 2.6. [26] A positive linear functional is normal if and only if there exists a positive unit-trace operator ρ such that $\omega(A) = \mathrm{tr}(A\rho)$ for all $A \in \mathcal{B}_{sa}(\mathcal{H})$. □

A prominent property of quantum mechanics which has no analogue in classical physics is that pure states of the system can occur in *superposition*. This concept is depicted mathematically as follows: Let us fix an orthonormal basis of the Hilbert space and note that each basis element corresponds to pure state. The sum (weighted with some complex number, and suitably normalised) of two or more such basis elements forms a state. More precisely, it is again a pure state, i.e., no statistical (or classical) ensemble of states and hence no mixed state. This new, purely quantum form of pure state combination is called superposition.

Observables

Let us consider the setup of a statistical experiment. It is useful to think of it as a two step process divided into two parts, the preparation and the measurement: So far, we have discussed the preparation, a procedure in which one associates a state to the quantum system. By another postulate of quantum mechanics, measurable quantities – *observables* – correspond to self-adjoint operators on the system's Hilbert space.

[4] This is a consequence of the spectral theorem for compact operators [139].

Definition 2.7 *(Observable)*. An *observable* is an element $A \in \mathcal{L}_{sa}(\mathcal{H})$.

Note that for a finite-dimensional Hilbert space, one may consider elements of $\mathcal{B}_{sa}(\mathcal{H})$, but for $\dim \mathcal{H} = \infty$, one allows for unbounded, densely defined operators.

Projective measurements on finite-dimensional Hilbert spaces

A measurement describes the procedure of extracting information about certain properties of a state. Here, the properties are specified by an observable. In contrast to classical physics, a quantum measurement is not simply a read-out of this information but it disturbs the quantum system, altering its state. As a consequence, the measurement associates to an observable-state pair a post-measurement state and measurement statistics of outcomes.

Let us start by considering a specific class of measurements, so-called *projective measurements on finite-dimensional Hilbert spaces*. They are essential to quantum error correcting codes for qubits (cf. Section 3.4).

Definition 2.8 *(Projective measurements)*. Let $\mathcal{H}$ be finite-dimensional. A *projective measurement* is described by a number of projections $P_1, \ldots, P_k \in \mathcal{P}(\mathcal{H})$ which are pairwise orthogonal, i.e., $P_i P_j = \delta_{i,j} P_i P_j$ for $i, j = 1, \ldots, k$, and sum up to the identity, i.e., $\sum_{i=1}^{k} P_i = I$.

Each P_i projects onto the subspace $P_i\mathcal{H}$ of $\mathcal{H}$. Since the projections are pairwise orthogonal, every pure state $|\psi\rangle \in \mathcal{H}$ can be decomposed uniquely as $|\psi\rangle = \sum_i P_i|\psi\rangle$, i.e., into a sum of elements $P_i|\psi\rangle \in P_i\mathcal{H}$ of these subspaces. Generally, a *projective measurement* $P_1, \ldots, P_k$ is in one-to-one correspondence with an *observable* $A \in \mathcal{B}_{sa}(\mathcal{H})$ in the following way: For the projective measurement $P_1, \ldots, P_k$ with distinct and non-zero outcomes $\lambda_1, \ldots, \lambda_k \in \mathbb{R}$, consider the operator

$$A = \sum_{i=1}^{k} \lambda_i P_i \ . \tag{2.3}$$

By construction, it is a self-adjoint operator, the associated *observable*. For the other direction, the spectral theorem for hermitian matrices (self-adjoint operators on a finite-dimensional Hilbert space) implies that $A \in \mathcal{B}_{sa}(\mathcal{H})$ has a complete orthonormal set of eigenvectors and hence admits the spectral decomposition of Eq. (2.3) where λ_i are the (real) eigenvalues of A, and P_i are the projections onto the eigenspaces corresponding to eigenvalues λ_i.

Consider a (potentially mixed) state $\rho \in \mathcal{D}(\mathcal{H})$ to which the measurement from Definition 2.8 is applied. The measurement outcome $\lambda_i = 1, \ldots, k$ occurs with probability $\mathrm{tr}(P_i\rho)$ and ρ is mapped to the post-measurement state

$$\rho \mapsto \frac{P_j \rho P_j}{\mathrm{tr}(P_j \rho)} \ .$$

A specific example of a projective measurement is the *measurement in the orthonormal basis* $\{|e_i\rangle\}_{i=1}^{k}$ of the k-dimensional Hilbert space $\mathcal{H} = \mathbb{C}^{\otimes k}$; the one-dimensional projections are given by $P_1 \coloneqq |e_1\rangle\langle e_1|, \ldots, P_k \coloneqq |e_k\rangle\langle e_k|$.

Projection-valued measure

Let us now generalise the concept of projective measurements to the setting of infinite-dimensional Hilbert spaces, where *observables* are given by (possibly unbounded) self-adjoint operators. To construct their measurements, we need the spectral theorem for unbounded self-adjoint operators; it guarantees a one-to-one correspondence between unbounded self-adjoint operators and so-called (unbounded) PVMs, i.e., here projective measurements are related to PVMs.

Definition 2.9 *(Projection valued measure).* Let $(\Omega, \mathcal{A})$ be a measure space where Ω is a set and $\mathcal{A}$ a σ-algebra of subsets of Ω. A *projection valued measure* (PVM) is a map $F : \mathcal{A} \to \mathcal{L}_{sa}(\mathcal{H})$ that satisfies:

(i) $F(\Omega) = I$ where I is the identity map on $\mathcal{H}$ and $F(\emptyset) = 0$.

(ii) $F(X)$ is an orthogonal projection for every $X \subset \Omega$.

(iii) For every sequence $(X_n)_{n\in\mathbb{N}} \subset \mathcal{A}$ of pairwise disjoint sets, the equation

$$F\left(\bigcup_{n=1}^{\infty} X_n\right) = \sum_{n=1}^{\infty} F(X_n)$$

holds, where the sum converges with respect to the strong operator topology.

There is a one-to-one correspondence between PVM's (for $\Omega \subset \mathbb{R}$) and self adjoint operators (observables) by the following theorem.

Theorem 2.10 *(Spectral theorem for unbounded self-adjoint operators).* Let $A \in \mathcal{L}_{sa}(\mathcal{H})$ with domain $D(A)$. Then there exists an associated PVM F such that for every $|\psi\rangle \in D(A), |\varphi\rangle \in \mathcal{H}$ one has

$$\langle\varphi|A|\psi\rangle = \int_{\mathbb{R}} \lambda \, d\langle\varphi|F(\lambda)|\psi\rangle \tag{2.4}$$

and where

$$D(A) = \left\{ |\psi\rangle \in \mathcal{H} \mid \int_{\mathbb{R}} \lambda^2 \, d\langle\psi|F(\lambda)|\psi\rangle < \infty \right\} . \tag{2.5}$$

Furthermore, the elements of the PVM are spectral projections in the sense that $\lambda \in \mathbb{R}$ is in the spectrum of A if and only if $F((\lambda - \epsilon, \lambda + \epsilon)) \neq 0$ for all $\epsilon > 0$.

Conversely, for a PVM $F : \mathcal{A} \to \mathcal{L}_{sa}(\mathcal{H})$, the operator A defined by the spectral decomposition from Eq. (2.4) is self-adjoint on $D(A)$ from Eq. (2.5). □

The proof of this result can be found for example in the book by Reed and Simon [139, Theorem VIII.6]. Denote by F_A the PVM associated with the observable $a \in \mathcal{L}_{sa}(\mathcal{H})$. Note that one may integrate with respect to a PVM F_A since for $\rho \in \mathcal{D}(\mathcal{H})$

$$\mu_{F_A} : \mathcal{A} \otimes \mathcal{D}(\mathcal{H}) \to \mathbb{R} \qquad , \qquad X \mapsto \mu_F(X) = \mathrm{tr}(F_A(X)\rho)$$

is a well-defined measure on $\mathcal{A}$. Let us now state how a PVM describes the measurement of an observable.

Definition 2.11 *(Projective quantum measurement).* Let $\rho \in \mathcal{D}(\mathcal{H})$. A *projective measurement* (or von Neumann measurement) is described by a PVM $F : \mathcal{A} \to \mathcal{L}(\mathcal{H})$ where $\mathcal{A}$ is a σ-algebra of subsets of $\Omega \subset \mathbb{R}$. The outcome of the measurement F lies in $E \subset \Omega$ with probability $\operatorname{tr}(F(E)\rho)$ and then the post-measurement state is given by $F(E)\rho/\operatorname{tr}(F(E)\rho)$.

The measurable space Ω constitutes the possible values that the measurement outcomes of the observable A can take. The associated PVM defines the measurement statistics, more precisely the probabilities with which the outcomes are attained.

Note that projective measurements can be generalised even further to measurements which are not necessarily projective; described by so-called positive operator valued measures (POVMs) – cf. e.g. [121, Chapter 2.2.6] – where in contrast to Definition 2.9 (ii) not all elements have to be projections but only positive adding up to the identity. In contrast to PVMs, repeating a non-projective measurements may not produce the same outcome again. A POVM can be understood as the reduced effect of a PVM on a subsystem of a larger physical system. But since this concept is not needed in the work at hand, we will not further consider it.

The *expectation value of the observable* $A \in \mathcal{L}_{sa}(\mathcal{H})$ is given by $\operatorname{tr}(\rho A)$. Note that although a physically relevant quantity is described by an unbounded self-adjoint operator, the measurement statistics only depend on its spectral projections (defined by the PVM) which are bounded.

2.1.3 An example: the quantum bit

A very simple example of a quantum system which is ubiquitous in quantum computing is the *quantum bit* or simply the *qubit*. The associated Hilbert space $\mathcal{H} = \mathbb{C}^2$ is two-dimensional. Let us fix the canonical basis

$$|0\rangle := \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad , \quad |1\rangle := \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{2.6}$$

of $\mathcal{H}$ which is called the *computational basis*. In quantum computing, these basis vectors play analogous roles to their classical counterparts, the classical bit values 0 and 1. As illustrated in the preceding paragraphs, they describe pure states with corresponding density matrices $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$, respectively.

The bounded operators on $\mathbb{C}^2$ are 2×2-matrices with complex entries. To characterise the sets of observables (i.e., self-adjoint operators) and density operators (i.e., trace class operators), it is useful to introduce the *Pauli matrices*

$$\sigma_x := \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad , \quad \sigma_y := \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad , \quad \sigma_z := \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} .$$

Note that the computational basis consists of eigenvectors of σ_z with eigenvalues ± 1 and is therefore also called the z-eigenbasis. The eigenvectors to σ_x are

$$|+\rangle := \frac{1}{\sqrt{2}}\big(|0\rangle + |1\rangle\big) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad , \quad |-\rangle := \frac{1}{\sqrt{2}}\big(|0\rangle + |1\rangle\big) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} ,$$

forming the so-called polar or Hadamard basis, and the eigenvectors of σ_y are

$$|+i\rangle := \frac{1}{\sqrt{2}}\big(|0\rangle + i|1\rangle\big) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ i \end{pmatrix} \quad , \quad |-i\rangle := \frac{1}{\sqrt{2}}\big(|0\rangle + i|1\rangle\big) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -i \end{pmatrix} .$$

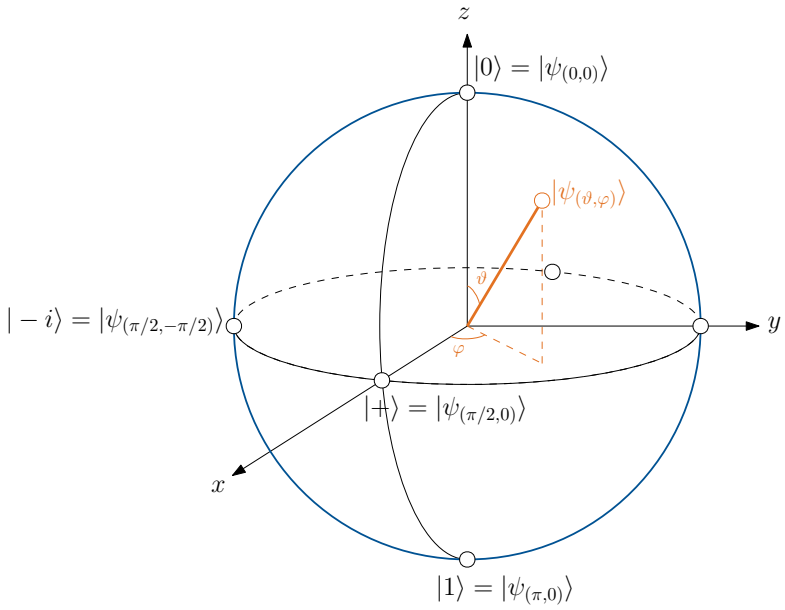


Figure 2.1: Bloch sphere: A qubit's pure state corresponds to a vector on the sphere. It can be either expressed in terms of Cartesian coordinates $\vec{a} = (a_x, a_y, a_z)$ or spherical coordinates (ϑ, φ) as $|\psi_{(\vartheta,\varphi)}\rangle$; examples are the σ_z-eigenstates $|0\rangle$ and $|1\rangle$ and the $(+1)$-eigenstate $|+\rangle$ of σ_x and the (-1)-eigenstate $|-i\rangle$ of σ_y.

The Pauli matrices are trace-less unitary and self-adjoint – or hermitian as self-adjoint operators on a finite-dimensional Hilbert space are represented by Hermitian matrices. They anticommute with each other, i.e., they satisfy $\{\sigma_i, \sigma_j\} = 0$ for all $i, j \in \{x, y, z\}$ such that $i \neq j$ where $\{A, B\} \coloneqq AB + BA$ denotes the anticommutator of two operators A and B. Together with the identity matrix they form a basis of $\mathcal{B}_{sa}(\mathbb{C}^2)$, where we regard $\mathcal{B}_{sa}(\mathbb{C}^2)$ as a vector space over the real numbers: More precisely, every complex Hermitian 2×2-matrix can be written uniquely as a linear combination (with real coefficients) of the identity and the three Pauli matrices: hence the set of self adjoint operators is given by

$$\mathcal{B}_{sa}(\mathbb{C}^2) = \left\{ \alpha I + \beta\sigma_x + \gamma\sigma_y + \delta\sigma_z \;\middle|\; \alpha, \beta, \gamma, \delta \in \mathbb{R} \right\} . \tag{2.7}$$

Additionally imposing the trace condition and positivity on the above hermitian operators yields that every density matrix $\rho \in \mathcal{D}(\mathbb{C}^2)$ of a qubit can be written as

$$\rho = \frac{1}{2}\left(I + \vec{a} \cdot \vec{\sigma}\right) = \frac{1}{2}(I + a_x\sigma_x + a_y\sigma_y + a_z\sigma_z) ,$$

where $\vec{\sigma} = (\sigma_x, \sigma_y, \sigma_z)$ and $\vec{a} = (a_x, a_y, a_z) \in \mathbb{R}^3$, i.e., the set of states is given by

$$\mathcal{D}(\mathbb{C}^2) = \left\{ \frac{1}{2}\left(I + \vec{a} \cdot \vec{\sigma}\right) \;\middle|\; \vec{a} \in \mathbb{R}^3,\ \|\vec{a}\| \leq 1 \right\} .$$

Hence, the density matrix of a qubit can be represented by a three-dimensional vector $\vec{a}$ which is an element of a three-dimensional unit ball (since $\|\vec{a}\| \leq 1$). The pure states correspond to its boundary (i.e., $\|\vec{a}\| = 1$), which is called the Bloch sphere and is depicted in Fig. 2.1.

Let us also express the unitary 2×2-matrices in terms of the Pauli matrices as they will be basic components of the circuit model of quantum computation in Section 2.3.1. To this end, recall that the complex unitary 2×2-matrices form the Lie group generated by the Lie algebra $\mathcal{B}_{sa}(\mathbb{C}^2)$ of self-adjoint matrices, i.e., e^{iA} is unitary for all $A \in \mathcal{B}_{sa}(\mathbb{C}^2)$ and

every unitary $U \in \mathsf{U}(\mathbb{C}^2)$ can be written as $U = e^{iA}$ for some $A \in \mathcal{B}_{sa}(\mathbb{C}^2)$. Recall the decomposition from Eq. (2.7) and rewrite it in terms of the parameter $\vartheta \in \mathbb{R}$ and a unit vector $\vec{n} = (n_x, n_y, n_z) \in \mathbb{R}^3$, $\|\vec{n}\| = 1$ as

$$e^{i(\alpha I + \beta\sigma_x + \gamma\sigma_y + \delta\sigma_z)} = e^{i\alpha} e^{i\vartheta\vec{n}\cdot\vec{\sigma}} = e^{i\alpha}\left(\cos(\vartheta) I + i\sin(\vartheta)\vec{n}\cdot\vec{\sigma}\right),$$

i.e., any complex unitary 2×2-matrix can be written – up to a global phase $e^{i\alpha}$ – as

$$e^{i\vartheta\vec{n}\cdot\vec{\sigma}} = \cos(\vartheta) I + i\sin(\vartheta)\vec{n}\cdot\vec{\sigma} = \cos(\vartheta) I + i\sin(\vartheta)(n_x\sigma_x + n_y\sigma_y + n_z\sigma_z)$$

for some $\vartheta \in \mathbb{R}$ and $\vec{n} = (n_x, n_y, n_z) \in \mathbb{R}^3$ such that $\|\vec{n}\| = 1$. When we keep the picture of the Bloch sphere in mind, such a unitary corresponds to a rotation of an angle ϑ around the axis specified by the vector $\vec{n}$.

Let us illustrate the concept of a measurement in the computational basis (2.6). This measurement has practical relevance, for example in the quantum circuit model, cf. Section 2.3.1. The two corresponding projections are $P_0 = |0\rangle\langle 0|$ and $P_1 = |1\rangle\langle 1|$. If the qubit is in the pure state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ for $\alpha, \beta \in \mathbb{C}$ such that $|\alpha|^2 + |\beta|^2 = 1$ then with probability $p_0 \coloneqq \mathrm{tr}(P_0|\psi\rangle\langle\psi|) = |\alpha|^2$ one obtains the post measurement state $|0\rangle$ with associated measurement outcome λ_0 and similarly, with probability $p_1 \coloneqq \mathrm{tr}(P_1|\psi\rangle\langle\psi|) = |\beta|^2$ the post-measurement state is $|1\rangle$ and the outcome is λ_1. Setting the two distinct measurement outcomes to $\lambda_0 = +1$ and $\lambda_1 = -1$, the observable corresponding to the measurement $\{P_0, P_1\}$ is given by the matrix $A = \lambda_0 P_0 + \lambda_1 P_1 = \sigma_z$.

2.1.4 Composite systems and entanglement

In many physical settings, one wants to describe the composition of several quantum systems: For example, for a quantum computation one wants to describe several qubits on which one may act jointly with multi-qubit operations. More generally, a quantum system may not be isolated perfectly from its environment and one has to take interactions between the quantum system and its environment into account.

Tensor product of Hilbert spaces

The mathematical description of *composite quantum systems* exploits the concept of tensor products of Hilbert spaces. The *tensor product* $\mathcal{H}_1 \otimes \mathcal{H}_2$ of two Hilbert spaces $\mathcal{H}_1$ and $\mathcal{H}_2$ is defined as follows. Elements of $\mathcal{H}_1 \otimes \mathcal{H}_2$ are of the form

$$\sum_{i=1}^{n} |\phi_i\rangle \otimes |\psi_i\rangle \qquad \text{for } |\phi_i\rangle \in \mathcal{H}_1,\ |\psi_i\rangle \in \mathcal{H}_2, \tag{2.8}$$

where the tensor product $\otimes$ is linear in both arguments and n is finite if both Hilbert spaces are finite-dimensional (more precisely, $\dim(\mathcal{H}_1 \otimes \mathcal{H}_2) = \dim(\mathcal{H}_1) \times \dim(\mathcal{H}_2)$) whereas $n \to \infty$ if one of the Hilbert spaces is infinite-dimensional; here, taking the completion of elements of the form (2.8) is necessary to ensure the completeness of $\mathcal{H}_1 \otimes \mathcal{H}_2$). We write $|\psi \otimes \phi\rangle = |\psi\rangle \otimes |\phi\rangle$. The inner product on $\mathcal{H}_1 \otimes \mathcal{H}_2$ is defined as $\langle \phi_1 \otimes \psi_1 | \phi_2 \otimes \psi_2 \rangle = \langle \phi_1 | \phi_2 \rangle \langle \psi_1 | \psi_2 \rangle$ for $|\phi_1\rangle, |\phi_2\rangle \in \mathcal{H}_1$, $|\psi_1\rangle, |\psi_2\rangle \in \mathcal{H}_2$ and extends to $\mathcal{H}_1 \otimes \mathcal{H}_2$ by linearity. The tensor product of two bases $\{|e_i\rangle\}_i$ of $\mathcal{H}_1$ and $\{f_j\}_j$ of $\mathcal{H}_2$ forms a basis of $\mathcal{H}_1 \otimes \mathcal{H}_2$.

Definition 2.12 *(Composite system).* Let two quantum systems with associated Hilbert spaces $\mathcal{H}_1$ and $\mathcal{H}_2$ be given. Then the Hilbert space of the joint system is the tensor product $\mathcal{H}_1 \otimes \mathcal{H}_2$.

Tensor products of operators are defined similarly: If $A \in \mathcal{L}(\mathcal{H}_1)$ and $B \in \mathcal{L}(\mathcal{H}_2)$, then $A \otimes B \in \mathcal{L}(\mathcal{H}_1 \otimes \mathcal{H}_2)$ is given by its action $(A \otimes B)(|\psi\rangle \otimes |\phi\rangle) = A|\psi\rangle \otimes B|\phi\rangle$ for $|\psi\rangle \in \mathcal{H}_1$, $|\phi\rangle \in \mathcal{H}_2$ and again extended to all of $\mathcal{H}_1 \otimes \mathcal{H}_2$ by linearity. For $n \in \mathbb{N}$, the expression $\mathcal{H}^{\otimes n}$ denotes the n-fold tensor product of $\mathcal{H}$, e.g., one writes $(\mathbb{C}^2)^{\otimes n}$ to denote the Hilbert space of an n-qubit quantum system which has dimension 2^n: Similarly, $(\sigma_x)^{\otimes n}$ denotes the operator acting as a Pauli-X on all n qubits. In slight abuse of notation, operators that only acts non-trivially on one of the subsystems are labelled by an index of the respective subsystem, implying tensoring identities on the other systems: for example $(\sigma_x)_2$ is short hand notation for $I \otimes \sigma_x \otimes I^{\otimes(n-2)}$.

Subsystems and entanglement

If we are given a system-environment state, one may ask how the 'substate' on the system part which is accessible by the experimentalist looks like. If the state is in tensor product form, e.g., $\rho = \rho_1 \otimes \rho_2 \in \mathcal{D}(\mathcal{H}_1 \otimes \mathcal{H}_2)$ for $\rho_1 \in \mathcal{D}(\mathcal{H}_1)$ and $\rho_2 \in \mathcal{D}(\mathcal{H}_2)$, then the answer is straightforward: Up to normalisation, ρ_1 and ρ_2 are the reduced states of ρ on the subsystems 1 and 2, respectively. For general state $\rho \in \mathcal{D}(\mathcal{H}_1 \otimes \mathcal{H}_2)$ this can answered using the concept of the *partial trace*.

Definition 2.13 *(Partial trace).* Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be Hilbert spaces. The *partial trace* over system 1 is the map $\mathrm{tr}_1 : \mathcal{T}(\mathcal{H}_1 \otimes \mathcal{H}_2) \to \mathcal{T}(\mathcal{H}_2)$ that satisfies

$$\mathrm{tr}\left(\mathrm{tr}_1(A)T\right) = \mathrm{tr}\left(A(I \otimes T)\right) \qquad \text{for all } A \in \mathcal{B}(\mathcal{H}_1 \otimes \mathcal{H}_2),\ T \in \mathcal{B}(\mathcal{H}_2)\ .$$

If $\rho \in \mathcal{D}(\mathcal{H}_1 \otimes \mathcal{H}_2)$ is a state on $\mathcal{H}_1 \otimes \mathcal{H}_2$, then the *reduced state of* ρ on subsystem 1, denoted by ρ_1, is given by $\mathrm{tr}_2(\rho)$. A state $\rho \in \mathcal{D}(\mathcal{H}_1 \otimes \mathcal{H}_2)$ is called *separable* if it can be written as a probability distribution over product states, i.e., for some index set K as $\sum_{k \in K} p_k \rho_k \otimes \sigma_k$ for a probability distribution $p_k : K \to [0,1]$ and $\rho_k \in \mathcal{D}(\mathcal{H}_1)$ and $\sigma_k \in \mathcal{D}(\mathcal{H}_2)$ for all $k \in K$. An *entangled* state is defined as a state which is not separable.

2.1.5 Quantum operations and dynamics

Let us now describe transformations of one quantum state to another. One relevant example might be the evolution of a system over time but also the so-called quantum gates which are operations associated with a quantum computation.

Recall the division of the statistical experiment into preparation and measurement. The transformation can be either regarded as part of the preparation, i.e., acting on the state, or as part of the measurement, i.e., acting on the observable. This gives rise to two different pictures: we call the former the Schrödinger picture and the latter the Heisenberg picture. Both pictures should model the same physics. Hence, they give rise to the same measurement statistics. More precisely, if a state $\rho \in \mathcal{D}(\mathcal{H})$ transforms as $\rho \mapsto \Phi(\rho)$ under a map Φ in the Schrödinger picture and an observable $A \in \mathcal{L}_{sa}(\mathcal{H})$ transforms as $A \mapsto \Psi(A)$ under a map Ψ in the Heisenberg picture, then

$$\mathrm{tr}(\Phi(\rho)A) = \mathrm{tr}(\rho\Psi(A)) \tag{2.9}$$

has to hold. We will make this assumption more precise later.

Quantum channel

Let us describe the most general transformation on a quantum system and start in the Schrödinger picture, i.e., with a map from quantum states to quantum states. More precisely, since states have unit trace, the map $\Phi : \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H})$ must be trace preserving, i.e., $\mathrm{tr}(\Phi(\rho)) = \mathrm{tr}(\rho)$ for all $\rho \in \mathcal{D}(\mathcal{H})$, and it must be positive, i.e., $\Phi(\rho) \geq 0$ for all $\rho \geq 0$. We will ask for a little more: a map $\Phi : \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H})$ is *completely positive* if it is positive and the map $\Phi \otimes I_n$ is positive for all $n \in \mathbb{N}$, where I_n is the identity operator on $\mathcal{B}(\mathbb{C}^n)$.

Definition 2.14 *(Quantum channel).* A *quantum channel* or *quantum operation* is a completely positive and trace preserving (CPTP) map $\Phi : \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H})$.

The expression 'quantum channel' originates from quantum information theory where a CPTP map is viewed as a communication channel that can transmit quantum information.

Note that there are maps which are positive but not completely positive, e.g., the transposition map on $\mathcal{B}(\mathbb{C}^k)$ for $k \in \mathbb{N}$, implying that complete positivity is a restriction of positivity. The restriction to complete positivity (instead of only positivity) has an insightful physical interpretation: Let $\rho \in \mathcal{D}(\mathcal{H}_S)$ be q state on a quantum system $\mathcal{H}_S$ and regard it as the reduced state of a state $\sigma \in \mathcal{D}(\mathcal{H}_S \otimes \mathcal{H}_2)$ on a larger Hilbert space $\mathcal{H}_S \otimes \mathcal{H}_2$. Then $(\Phi \otimes \mathbb{I}_{\mathcal{B}(\mathcal{H}_2)})(\sigma)$ should again be a state on the larger Hilbert space, implying that the map $\Phi \otimes \mathbb{I}_{\mathcal{B}(\mathcal{H}_2)}$ must be positive. Since this should hold for any (finite-dimensional) auxiliary Hilbert space $\mathcal{H}_2$, the map Φ must be completely positive.

We note that every quantum channel Φ, in the Schrödinger picture, induces a transformation Φ^* in the Heisenberg picture which is defined by the relation $\mathrm{tr}(\Phi(\rho)A) = \mathrm{tr}(\rho\Phi^*(A))$ for $\rho \in \mathcal{D}(\mathcal{H})$ and $A \in \mathcal{B}_{sa}(\mathcal{H})$. As a consequence, the quantum channel in the Heisenberg picture Φ^* is completely positive and unital, i.e., $\Phi^*(I_{\mathcal{H}}) = I_{\mathcal{H}}$.

In the finite-dimensional setting, i.e., if $\mathcal{H} \cong \mathbb{C}^n$ for some $n \in \mathbb{N}$, a quantum channel can be characterised by a finite set of operators, the Kraus operators.

Theorem 2.15 *(Kraus representation [101, 100]).* A linear map $\Phi : \mathcal{B}(\mathbb{C}^n) \to \mathcal{B}(\mathbb{C}^n)$ is a CPTP map if and only if it admits the following representation

$$\Phi(A) = \sum_{i=1}^{k} K_i A K_i^\dagger \ ,$$

where the *Kraus operators* $\{K_i\}_{i=1}^k$ satisfy $\sum_{i=1}^k K_i^\dagger K_i = I_n$. The Kraus rank r, which is defined as the minimal value of $k \in \mathbb{N}$, is upper bounded by n^2. There is a set of r Kraus operators such that $K_i^\dagger K_j = \delta_{ij}$. Two sets of Kraus operators $\{K_i\}_{i=1}^k$ and $\{\widetilde{K}_i\}_{i=1}^l$ represent the same CPTP map Φ if and only if they are unitarily equivalent. □

Quantum instrument

Recall the quantum measurement (cf. Section 2.1.2) of an observable. It does not only reveal the outcome, but also alters the quantum state (cf. Definition 2.11) on which the measurement

is performed. Hence its full mathematical description includes the transformation of the quantum state. The concept of a quantum instrument captures both the quantum – the post-measurement state – as well as the classical output – the measurement outcome – of this process.

Definition 2.16 *(Quantum instrument).* A *quantum instrument* is a map

$$\mathcal{I} : \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H}) \otimes \mathcal{B}(\mathbb{C}^{|X|}) \quad , \quad \rho \mapsto \sum_{x \in X} \Phi_x(\rho) \otimes |x\rangle\langle x| \ ,$$

where X is a finite set of measurement outcomes and $\{\Phi_x\}_{x \in X}$ is a collection of quantum channels.

A quantum instrument is connected to a measurement in the following sense: the set X constitutes the possible measurement outcomes and $\Phi_x(\rho)$ the possible post-measurement states. Assume that the system is initially in the state $\rho \in \mathcal{D}(\mathcal{H})$ and the measurement outcome $x \in X$ occurs with probability $\mathrm{tr}(\Phi_x(\rho))$. If the outcome is $x \in X$ then the post-measurement quantum state is given by the normalised image under the associated CPTP map Φ_x, i.e., by $\Phi_x(\rho)/\mathrm{tr}(\Phi_x(\rho))$.

Reversible evolutions

Physically, a closed quantum system involves no interaction with its environment, it is isolated and no information can leak from the system to its environment and vice versa. In contrast, a quantum system which is not fully isolated from its environment is called open. Closed quantum systems undergo reversible evolutions.

Definition 2.17 *(Unitary evolution).* The *reversible evolution of a closed quantum system* is characterised by a unitary operator $U \in \mathsf{U}(\mathcal{H})$. In the Schrödinger picture, the quantum states are transformed as

$$\Phi : \mathcal{D}(\mathcal{H}) \to \mathcal{D}(\mathcal{H}) \quad , \quad \rho \mapsto \Phi(\rho) = U\rho U^\dagger \ , \tag{2.10}$$

and in the Heisenberg picture, the observables are transformed as

$$\Psi : \mathcal{B}_{sa}(\mathcal{H}) \to \mathcal{B}_{sa}(\mathcal{H}) \quad , \quad A \mapsto \Psi(A) = U^\dagger A U \ . \tag{2.11}$$

Note that these evolutions (2.10) and (2.11) are compatible with Eq. (2.9). The two maps Ψ and Φ are mutually dual maps, i.e., $\Phi = \Psi^*$ and one says that the Schrödinger picture and the Heisenberg pictures are dual to each other.

A unitary $U \in \mathsf{U}(\mathcal{H})$ obviously defines a quantum channel Ψ_U by the relation

$$\Psi_U : \mathcal{D}(\mathcal{H}) \to \mathcal{D}(\mathcal{H}) \quad , \quad \Psi_U(\rho) = U\rho U^\dagger \ ,$$

i.e., reversible evolutions (in the Schrödinger as well as the Heisenberg picture) can be described by quantum channels. The converse is generally not true; there are quantum channels which are not reversible. But due to the following theorem, every quantum channel is related to the reversible evolution of a larger system.

Theorem 2.18 *(Stinespring dilation [153])*. Let $\mathcal{H}$ be a Hilbert space. The linear map $\Phi : \mathcal{D}(\mathcal{H}) \to \mathcal{D}(\mathcal{H})$ is completely positive if and only if there exists a Hilbert space $\mathcal{H}_E$ and a bounded operator $V : \mathcal{B}(\mathcal{H}) \to \mathcal{B}(\mathcal{H} \otimes \mathcal{H}_E)$ such that

$$\Phi(\rho) = \mathrm{tr}_E(V \rho V^\dagger) \ .$$

Furthermore, the map V is an isometry if and only if Ψ is trace preserving. □

In this sense, the case of a reversible quantum channel is still generic. Every (irreversible) evolution of a quantum state corresponds to the reduced state of a reversible evolution on a larger system and an open system can be regarded as a subsystem of a closed larger quantum system.

Continuous time evolution and Schrödinger equation

Time is a special physical quantity: in contrast to many others it is not an observable (an neither the outcome of one) but described by a real parameter. The continuous-time evolution of a closed quantum system is mathematically modelled by a one-parameter group of unitaries.

Definition 2.19 *(One-parameter group)*. A *strongly continuous one-parameter unitary group* is a family $(U(t))_{t\in\mathbb{R}}$ of unitary operators $U(t) \in \mathsf{U}(\mathcal{H})$ for all $t \in \mathbb{R}$ such that the following holds:

(i) the family forms a group, i.e., $U(0) = I_\mathcal{H}$, $U(t)U(s) = U(t+s)$ for all $s, t \in \mathbb{R}$, and $U(-t)U(t) = I_\mathcal{H}$ for all $t \in \mathbb{R}$.

(ii) The map $t \mapsto U(t)$ is strongly continuous, i.e., $(U(t) - U(0))|\psi\rangle \to 0$ as $t \to 0$ for all $|\psi\rangle \in \mathcal{H}$.

The time evolution is related to a self-adjoint operator by the following theorem.

Theorem 2.20 *(Stone's theorem)*. Let $(U(t))_{t\in\mathbb{R}}$ be strongly continuous one-para-meter group of unitary operators on a Hilbert space $\mathcal{H}$. Then there exists a unique self-adjoint densely defined operator $A : D(A) \to \mathcal{H}$ which satisfies

$$U(t) = e^{-itA} \ . \tag{2.12}$$

Conversely, let $A : D(A) \to \mathcal{H}$ be a self-adjoint densely defined operator. Then the unitaries $U(t)$ defined by Eq. (2.12) form a strongly continuous one-parameter group of unitary operators $(U(t))_{t\in\mathbb{R}}$. □

For a proof of this Theorem, we refer to the standard literature [139, Section VIII.4]. The operator A is called an infinitesimal generator of $(U(t))_{t\in\mathbb{R}}$. Note that A is bounded if and only if $t \mapsto U(t)$ is norm-continuous. If A is bounded then (2.12) is defined by the power series expansion but if A is unbounded then (2.12) can be made sense of by functional calculus (using the spectral theorem for unbounded operators).

In quantum mechanics, one calls the generator of the time evolution the *Hamiltonian* and uses the letter H. Its eigenstates describe the energy levels or states of a system and the

eigenvalues are the associated energies. A pure quantum state $|\psi\rangle \in \mathcal{H}$ evolves according to the *Schrödinger equation*

$$\hbar \frac{d}{dt}|\psi(t)\rangle = -iH|\psi(t)\rangle \quad , \quad \text{where } |\psi(t_0)\rangle = |\psi(0)\rangle \ , \tag{2.13}$$

for a Hamiltonian $H \in \mathcal{L}_{sa}(\mathcal{H})$ and where $\hbar \approx 1.054571817 \times 10^{-34} Js$ is the Planck constant. Note that for convenience, we work within natural units throughout the remainder of this thesis, i.e., setting $\hbar = 1$. As a consequence of the Schrödinger equation (2.13), a general state $\rho \in \mathcal{D}(\mathcal{H})$ evolves as described by the equation

$$\frac{d}{dt}\rho(t) = -i[H, \rho(t)] \ .$$

The *propagator* or the *time evolution operator* $U(t, t_0)$ from time t_0 to time t is defined by the relation

$$|\psi(t)\rangle \coloneqq U(t, t_0)|\psi_0\rangle \quad , \quad \text{for all } |\psi_0\rangle \in \mathcal{H} \ , \tag{2.14}$$

where $|\psi(t)\rangle$ denotes the state at time t and $|\psi_0\rangle = |\psi(t_0)\rangle$ the initial state at time t_0. If a Hamiltonian H is given that is not time-dependent then by the Schrödinger equation (2.13) one has $U(t, t_0) = U(t - t_0) = e^{-i(t-t_0)H}$ as in Eq. (2.12). The term propagator refers to the observation that an initial state $|\psi\rangle$ evolves from time t_0 to time t as given by Eq. (2.14). We says that $U(t, t_0)$ is generated by H and that the quantum system evolves under the Hamiltonian. In slight abuse of notation we sometimes neglect the second variable of the propagator and write $U(t)$ to denote $U(t) = U(t, 0)$.

Note that if the Hamiltonian is time-dependent, i.e., if $H : \mathbb{R} \to \mathcal{L}_{\text{sa}}(\mathcal{H})$, $t \mapsto H(t)$ then the propagator $U(t, t_0)$ from Eq. (2.14) is not simply given by the exponential of the Hamiltonian. By inserting the propagator (2.14) into the Schrödinger equation (2.13), one finds that it is defined by initial value problem

$$\frac{d}{dt}U(t, t_0) = -iH(t)U(t, t_0) \qquad , \qquad U(t_0, t_0) = I \ . \tag{2.15}$$

Let us briefly describe the strategy to compute the solution to (2.15) numerically. One splits the time interval into N equal parts $\delta t = (t - t_0)/N$. Integrating the Schrödinger equation over t up to first order in δt one finds that

$$|\psi(\delta t)\rangle = |\psi(t_0)\rangle + \delta t \frac{d}{dt}|\psi(t)\rangle|_{t=t_0} + O\big((\delta t)^2\big) = (1 - i\delta t H(t))|\psi(t)\rangle + O\big((\delta t)^2\big) \ .$$

Note that in the finite-dimensional case ($\dim(\mathcal{H}) < \infty$) and if $H(t)$ commutes with $H(t')$ for all $t, t' \geq t_0$, then the limit $N \to \infty$ yields

$$|\psi(t)\rangle = \lim_{N\to\infty} \prod_{k=0}^{N-1} (1 - i\delta t H(k\delta t))|\psi(t_0)\rangle = e^{-i\int_{t_0}^{t} H(\tau)d\tau}|\psi(t_0)\rangle \ .$$

In general, this is not true since the Hamiltonian $H(t)$ may not commute with $H(t')$ at a different time t'. Formally, we write the solution to (2.15) as

$$U(t, t_0) = \mathcal{T}_+ \left[e^{-i\int_{t_0}^{t} H(\tau)d\tau}\right] \ ,$$

where $\mathcal{T}_+$ is called the *time ordering operator* and it satisfies

$$\mathcal{T}_+ \left[H(\tau_1)H(\tau_2)\cdots H(\tau_n)\right] = H(\tau_1)H(\tau_2)\cdots H(\tau_n) \quad \text{for all} \quad \tau_1 \leq \tau_2 \leq \cdots \leq \tau_n \ .$$

2.2 Bosonic quantum systems

Throughout this thesis, the focus is on *continuous variable* (CV) quantum systems. The term 'continuous variable' refers to the fact that such systems have certain associated observables with a continuous spectrum. Their Hilbert space is always infinite-dimensional.

The most prominent example of such a continuous variable quantum system is the quantum harmonic oscillator of a single bosonic mode where the continuous variables correspond to the eigenvalues of the position and momentum operators. More generally, this thesis treats systems of $n \in \mathbb{N}$ bosonic modes. Such a system may be used to describe n modes of the (quantised) electromagnetic field. But the same mathematics also model other quantum systems in certain parameter regimes such as atomic ensembles, Josephson junctions, trapped ions or nuclear spins of quantum dots.

To set the stage, we define the operators with continuous spectra giving the CV systems their name – the position and momentum operators – and the quantum harmonic oscillator in Section 2.2.1. In Section 2.2.2, we discuss the phase space description of an n-mode bosonic system, where instead of the unbounded operators with continuous spectrum, one considers so-called Weyl displacement operators in the $2n$-dimensional phase space. Of special interest are certain quantum states on such an n-mode system which are ubiquitous in quantum optics – the Gaussian states – as well as the quantum channels which preserve the Gaussianity of states – the Gaussian channels, both presented in Section 2.2.3. In Section 2.2.4, we present a representation of Gaussian reversible channels in terms in terms of $(2n \times 2n)$-matrices (real symplectic matrices) on the phase space, which is mathematically advantageous compared to the Schrödinger picture description with unbounded operators. We also give several examples of such channels in which we use in Chapters 4 and 5.

2.2.1 Schrödinger representation: Position and momentum operators

The description in this section is based on the one in the textbook by Reed and Simon [139]. The Hilbert space associated with a single bosonic mode $\mathcal{H} := L^2(\mathbb{R})$ consists of square integrable complex-valued functions. More precisely, its elements are equivalence classes of such functions up to differences on sets with Lebesgue measure zero. In slight abuse of notation we will consider functions as elements of $\mathcal{H}$, but they should always be understood as a representative of the corresponding equivalence class. Elements $\psi \in L^2(\mathbb{R})$ with norm 1 are associated with the pure states of the single-mode quantum system.

Position and momentum operators

Let us start with a single bosonic mode with Hilbert space $\mathcal{H} = L^2(\mathbb{R})$ and define two operators on $\mathcal{H}$ which play a major role throughout this thesis: the so-called quadratures of *position and momentum operators*.

Note that as they are unbounded operators on an infinite-dimensional Hilbert space, one has to pay attention at their domains.

Definition 2.21 *(Position and momentum).* The *position operator* on $\mathcal{H}$ is defined by $Q : D(Q) \to \mathcal{H}$ where $(Qf)(x) = xf(x)$ for all $x \in \mathbb{R}$ and where the domain of Q is given by

$$D(Q) \coloneqq \left\{ f \in \mathcal{H} \;\middle|\; Qf \in \mathcal{H} \right\} = \left\{ f \in \mathcal{H} \;\middle|\; \int_{\mathbb{R}} x^2 |f(x)|^2 d\mu(x) < \infty \right\} .$$

The *momentum operator* on $\mathcal{H}$ is defined as $P : D(P) \to \mathcal{H}$ where $(Pf)(x) = -i\frac{d}{dx}f(x)$ for all $x \in \mathbb{R}$ with domain

$$D(P) \coloneqq \left\{ f \in \mathcal{H} \mid Pf \in \mathcal{H} \right\} = \left\{ f \in \mathcal{H} \;\middle|\; f \in C^1(\mathbb{R}),\ \int_{\mathbb{R}} |f'(x)|^2 d\mu(x) < \infty \right\} .$$

Note that although for every $f \in \mathcal{H}$ the image $x \mapsto (Qf)(x)$ is a well-defined function, there are $f \in \mathcal{H}$ such that $Qf \notin \mathcal{H}$. Hence the restriction to the domain is necessary.

In order to define sums and products of Q and P, one has to construct a joint dense domain for both of the above operators. This is the *Schwartz space*

$$\mathcal{S}(\mathbb{R}) \coloneqq \left\{ f \in C^\infty(\mathbb{R}) \;\middle|\; \text{for all } \alpha, \beta \in \mathbb{N}_0 :\ \sup_{x \in \mathbb{R}} |x^\alpha D^\beta f(x)| < \infty \right\}$$

of functions with derivatives that decrease faster than any polynomial at infinity. Its topological dual is the space of *tempered distributions*

$$\mathcal{S}'(\mathbb{R}) \coloneqq \{ f : \mathcal{S}(\mathbb{R}) \to \mathbb{C} \mid f \text{ linear, continuous} \} .$$

These sets satisfy $\mathcal{S}(\mathbb{R}) \subset L^2(\mathbb{R}) \subset \mathcal{S}'(\mathbb{R})$ where the inclusion $L^2(\mathbb{R}) \subset \mathcal{S}'(\mathbb{R})$ should be understood as follows: let $f \in L^2(\mathbb{R})$ be a square integrable function, then the integration against the measure $f d\mu$ gives a tempered distribution $g \mapsto \int_{\mathbb{R}} f(x)g(x)d\mu(x)$. Since the Schwartz space $\mathcal{S}(\mathbb{R})$ is a subset of both $D(Q)$ and $D(P)$ and it is dense in $L^2(\mathbb{R})$, the operators Q and P can be redefined on the Schwartz space as their domain – on which they are essentially self-adjoint[5]. We will denote (in slight abuse of notation) their self-adjoint extensions by the same symbol.

The tempered distributions $\mathcal{S}'(\mathbb{R})$ can be used to give the eigenvectors of Q and P meaning. One can formally write down the eigenvalue equations for Q and P as $Q|q\rangle = q$ and $P|p\rangle = p$, respectively. Here, $q \in \mathbb{R}$ and $p \in \mathbb{R}$ are the continuous eigenvalues and $|q\rangle$ and $|p\rangle$ the eigenvectors. Although the latter are not square integrable (not even Schwartz functions), they can be given meaning as distributions, i.e., as elements of $\mathcal{S}'(\mathbb{R})$. This motivates a generalisation of the notion of states.

Definition 2.22 *(States).* *Pure states* are tempered distributions $\mathcal{S}'(\mathbb{R})$. Pure states that satisfy $|\psi\rangle \in L^2(\mathbb{R})$ are called *normalisable pure states* whereas $|\psi\rangle \in \mathcal{S}'(\mathbb{R})$ which are not in $L^2(\mathbb{R})$ are called *non-normalisable pure states*.

The families $\{|q\rangle\}_{q \in \mathbb{R}}$ and $\{|p\rangle\}_{p \in \mathbb{R}}$ are two bases of $\mathcal{S}'(\mathbb{R})$ and related to each other via Fourier transform

$$|q\rangle = \frac{1}{\sqrt{2}} \int_{\mathbb{R}} e^{-ipq} |p\rangle dp \qquad , \qquad |p\rangle = \frac{1}{\sqrt{2}} \int_{\mathbb{R}} e^{-ipq} |q\rangle dq \ .$$

[5]The Schwartz space is a joint domain of analytic vectors; the definition of analytic vectors and a proof of this property can be found in Chapter X.6 of [140].

We call $\{|q\rangle\}_{q\in\mathbb{R}}$ the *position basis* and $\{|p\rangle\}_{p\in\mathbb{R}}$ the *momentum basis* of $\mathcal{S}'(\mathbb{R})$. The Fourier transform is defined on $\mathcal{S}(\mathbb{R})$ and its inverse on $\mathcal{S}'(\mathbb{R})$. It extends to a unitary isomorphism on $L^2(\mathbb{R})$ which diagonalises both operators Q and P. One can write a pure state $|\psi\rangle \in \mathcal{S}'(\mathbb{R})$ in the position and momentum basis as $\psi(q) \coloneqq \langle q|\psi\rangle$ and $\psi(p) = 1/(\sqrt{2\pi}) \int_{\mathbb{R}} \psi(q) e^{iqp} dq$, respectively.

Commutation relations

The operators Q and P do not commute[6]: They satisfy

$$[Q, P] \coloneqq QP - PQ = iI \ , \tag{2.16}$$

where I is the identity on $\mathcal{S}(\mathbb{R})$. Eq. (2.16) is called the *canonical commutation relation*. In contrast to the unbounded operators Q an P, the generated one-parameter groups of unitaries $\{U(t) = e^{itQ}\}_{t\in\mathbb{R}}$ and $\{V(s) = e^{isP}\}_{s\in\mathbb{R}}$ (cf. Definition 2.19) contain only bounded operators and are therefore easier to study. These unitaries satisfy the commutation relations

$$U(t)V(s) = e^{-its} V(s)U(t) \qquad \text{for all } s, t \in \mathbb{R} \tag{2.17}$$

as a consequence of the commutation relation (2.16). For many purposes, it is sufficient to study a system described by the quadrature operators Q and P at the level of their generated unitary groups $U(t)$ and $V(s)$ due to the following result:

For every pair of strongly continuous one-parameter groups of unitaries $U(t), V(s)$ that satisfy Eq. (2.17), there are two unbounded self adjoint operators which are generators of $U(t)$ and $V(s)$, respectively, and satisfy the canonical commutation relation (2.16). Note that the converse inclusion does not hold in general and hence both 'commutation relations' are not equivalent: there are operators which satisfy (2.16) but their generated unitary groups do not obey (2.17). A proof of both of these results can be found e.g. in Chapter VIII.5 of [139].

Number states, creation and annihilation operators

A useful countable basis of $\mathcal{H} \coloneqq L^2(\mathbb{R})$ is given by the so called Hermite functions. In more detail, for $k \in \mathbb{N}_0 \coloneqq \mathbb{N} \cup \{0\}$ let

$$h_k : \mathbb{R} \to \mathbb{R} \qquad , \qquad h_k(x) \coloneqq (-1)^k e^{x^2} \frac{d^k}{dx^k} e^{-x^2} \qquad \text{for all } x \in \mathbb{R} \ ,$$

be the kth Hermite polynomial. Then, the kth Hermite function is defined as

$$f_k(x) \coloneqq (2^k k! \sqrt{\pi})^{-1/2} e^{-x^2/2} h_k(x) \qquad \text{for all } x \in \mathbb{R} \ . \tag{2.18}$$

The Hermite functions $\{f_k\}_{k\in\mathbb{N}}$ are Schwartz functions. They are orthonormal, i.e.,

$$\int_{\mathbb{R}} \overline{f_k(x)} f_j(x) dx = \delta_{k,j} \qquad \text{for all } k, j \in \mathbb{N}_0 \ ,$$

[6]Computing the commutator of two unbounded self-adjoint operators is in fact non-trivial because of domain issues. In contrast to popular belief, it is not sufficient for two operators to commute that the two operators commute on a joint dense domain, cf. Chapter VIII.5 of [139]. Instead, two operators commute if and only if all their spectral projections commute. Here, the domain $\mathcal{S}(\mathbb{R})$ is a sufficiently well-behaved joint domain (consisting of so called analytic vectors of Q and P) such that the commutator of the operators can be computed as $[Q, P]\psi$ for $\psi \in \mathcal{S}(\mathbb{R})$.

and satisfy a completeness relation in the sense of tempered distributions, i.e.,

$$\sum_{k=0}^{\infty} f_k(x) f_k(y) = \delta(x-y) \qquad \text{for all } x, y \in \mathbb{R} \ ,$$

where $\delta \in \mathcal{S}'(\mathbb{R})$ is the Dirac-Delta distribution. The Hermite functions form a countable basis of $L^2(\mathbb{R})$.

Definition 2.23 *(Number states).* The *number state basis* or *Fock basis* of the infinite-dimensional Hilbert space $L^2(\mathbb{R})$ is defined as

$$\{|n\rangle\}_{n\in\mathbb{N}_0} = \{|0\rangle, |1\rangle, |2\rangle, \ldots\} \ ,$$

where $|n\rangle = f_n$ is the nth Hermite function from Eq. (2.18).

Let us furthermore introduce the bosonic field operators a and $a^\dagger$. On the Schwartz functions they are defined via their relation to the position and momentum operators

$$a^\dagger \coloneqq \frac{1}{\sqrt{2}}(Q - iP) \qquad , \qquad a \coloneqq \frac{1}{\sqrt{2}}(Q + iP) \ .$$

These operators are adjoint, justifying the notation $a^\dagger$. As a consequence of the canonical commutation relations (2.16) between Q and P, they satisfy the commutation relations $[a, a^\dagger] = I$ where I is the identity operator on $\mathcal{H}$. The operator $a^\dagger a$ is diagonal in the number basis. Its eigenvalues $k \in \mathbb{N}_0$ are given by the relation $a^\dagger a|k\rangle = k|k\rangle$. Therefore, the operator $a^\dagger a$ is called the *number operator* and its eigenvalues $k \in \mathbb{N}$ the *(particle/photon/excitation) number*. Using the commutation relations, one can show that the bosonic field operators act as

$$a|0\rangle = 0 \ , \quad a|k\rangle = \sqrt{k}|k-1\rangle \text{ for } k \geq 1 \ , \quad a^\dagger|k\rangle = \sqrt{k+1}|k+1\rangle \text{ for } k \geq 0 \ ,$$

on the number basis states (i.e., on the Hermite functions): The operator a decreases the number basis state by one, the operator $a^\dagger$ increases it, i.e., a is said to destroy a photon or bosonic excitation and $a^\dagger$ to create one. They are therefore called the *bosonic annihilation and creation operators*, respectively, or ladder operators.

Let us motivate the name quadrature for the operators Q and P. Up to a factor of $\frac{1}{\sqrt{2}}$, the position and momentum operator correspond to the real and imaginary part of the bosonic field amplitude, respectively. That is why in analogy to the quadratures of the classical electromagnetic field, the operators Q and P are also called quadrature operators.

The quantum harmonic oscillator

The Hamiltonian of the *quantum harmonic oscillator* on a single mode is given by

$$H \coloneqq \frac{\widetilde{P}^2}{2m} + \frac{m\omega^2}{2}\widetilde{Q}^2 \ .$$

Here m denotes the quantum particle's mass, ω the oscillation frequency and the operators $\widetilde{Q}$ and $\widetilde{P}$ are related to the dimensionless operators Q and P from Definition 2.21 as follows: $Q = \sqrt{m\omega/\hbar}\,\widetilde{Q}$ and $P = 1/\sqrt{m\omega\hbar}\,\widetilde{P}$, such that

$$H = \frac{\omega}{2}(Q^2 + P^2) \qquad , \qquad D(H) \coloneqq \mathcal{S}(\mathbb{R}) \ .$$

In analogy to the classical harmonic oscillator, where the total energy is given by

$$p^2/(2m) + q^2 m\omega^2/2$$

for the classical position q and momentum p, it is called the quantum harmonic oscillator. One difference is that the quantum operators for position and momentum are unbounded operators that do not commute. The quantum harmonic oscillator is the prototype of an infinite-dimensional quantum system. One can show that the Hamiltonian H is essentially self-adjoint on its domain and that $Hf_k = \omega\,(k + 1/2)\,f_k$ for all $k \in \mathbb{N}_0$, implying that H is unbounded and that the Hermite functions are the eigenfunctions of H. Hence H is diagonal in the number state basis from Definition 2.23 and the quantum harmonic oscillator has energy eigenstates $\{|k\rangle\}_{k\in\mathbb{N}_0}$ with corresponding equidistant energy levels (eigenvalues) $\{\omega(k + 1/2)\}_{k\in\mathbb{N}}$.

Multiple bosonic modes

For $n \in \mathbb{N}$ bosonic modes, the Hilbert space is given by the n-fold tensor product of the single-mode Hilbert space, i.e., it is

$$\mathcal{H} \coloneqq \bigotimes_{k=1}^{n} \mathcal{H}_k \cong \left(L^2(\mathbb{R}^n)\right) \qquad \text{where} \qquad \mathcal{H}_k = L^2(\mathbb{R}) \ .$$

The index $k = 1, \ldots, n$ labels the modes. In slight abuse of notation, we will write Q_k to denote the operator $I^{\otimes(k-1)} \otimes Q \otimes I^{\otimes(n-k)}$. One defines a_k, $a_k^\dagger$, and P_k analogously.

2.2.2 Phase space representation

To overcome domain issues when dealing with the unbounded quadrature operators, there is an alternative description of bosonic modes: the phase space representation. The key observation is that instead of the unbounded quadrature operators Q and P, exponentials of these (self-adjoint) operators are bounded.

Definition 2.24 *(Phase space representation).* The *phase space* of $n \in \mathbb{N}$ bosonic modes is $\mathbb{R}^{2n}$ equipped with the symplectic form

$$\mathbb{R}^{2n} \times \mathbb{R}^{2n} \to \mathbb{R} \qquad , \qquad (x, y) \mapsto \langle x, Jy\rangle \coloneqq x^T J y \ ,$$

where J is a non-singular, antisymmetric matrix. Here,

$$J \coloneqq \begin{pmatrix} 0 & I_n \\ -I_n & 0 \end{pmatrix} \ , \tag{2.19}$$

where I_n is the $n \times n$-identity matrix and we order the quadratures or mode operators in the vector

$$R^T = (Q_1, \ldots, Q_{2n}, P_1, \ldots, P_{2n}) \ . \tag{2.20}$$

The symplectic matrix J from Eq. (2.19) realises the canonical commutation relations

$$[R_j, R_k] = iJ_{j,k} \qquad \text{for } j, k = 1, \ldots, 2n \ . \tag{2.21}$$

There exist different conventions. Alternatively, the ordering $Q_1, P_1, \ldots, Q_n, P_n$ of mode operators requires the symplectic matrix

$$\widetilde{J} \coloneqq \bigoplus_{k=1}^{n} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

(instead of J) to realise the symplectic form $\mathbb{R}^{2n} \times \mathbb{R}^{2n} :\to \mathbb{R},\ (x, y) \mapsto \langle x, \widetilde{J}y\rangle \coloneqq x^T \widetilde{J} y$ on the phase space. We will choose J as in Eq. (2.19) if not mentioned otherwise.

Displacement operators and Weyl relations

Definition 2.25 *(Weyl displacement).* Let $\xi \in \mathbb{R}^{2n}$ be a phase space element and R the quadrature vector from Eq. (2.20). The (Weyl) *displacement operator* is defined as

$$D(\xi) \coloneqq e^{iR^T J \xi} \ . \tag{2.22}$$

On a single mode, Eq. (2.22) simply becomes $D(\xi) = e^{i(\xi_2 Q - \xi_1 P)}$. As a consequence of the canonical commutation relations (2.21), the displacement operators satisfy the *Weyl relations*

$$\begin{aligned} D(\xi)D(\eta) &= e^{\frac{i}{2}\xi^T J \eta} D(\xi + \eta) \qquad \text{for all } \xi, \eta \in \mathbb{R}^{2n} \ , \\ D(\xi)D(\eta) &= e^{i\xi^T J \eta} D(\eta)D(\xi) \qquad \text{for all } \xi, \eta \in \mathbb{R}^{2n} \ . \end{aligned} \tag{2.23}$$

The terminology 'displacement' for $D(\xi)$ becomes clear when considering its action on the quadratures R_j: Using the Baker-Campbell-Hausdorff formula[7] and the commutation relations between the quadratures, this action can be determined as

$$D(\xi)^\dagger R_j D(\xi) = R_j + \xi_j I \qquad \text{for all } \xi \in \mathbb{R}^{2n},\ j = 1, \ldots, 2n \ .$$

i.e., the displacement $D(\xi)$ shifts the quadratures by an amount of $\xi \in \mathbb{R}^{2n}$.

What we have seen in the previous section for a single bosonic mode can be generalised to the n-mode system: Every family of unitaries $\{U_\xi\}_{\xi \in \mathbb{R}^{2n}}$ that satisfies the Weyl relations (2.23) (for $U_\xi = D(\xi)$) and is strongly continuous can be expressed in the form of Eq. (2.22) for unbounded self-adjoint operators R that satisfy the canonical commutation relations (on a suitably chosen domain of essential self-adjointness). Such a family is called a Weyl system. Hence, one can take the Weyl system as a starting point and deduce the corresponding position and momentum operators from it. Furthermore, the Weyl relations (2.23) imply that the displacement operators yield a representation $(\xi, \alpha) \mapsto e^{-i\alpha} D(\xi)$ of the Heisenberg-Weyl H_n group (cf. Definition 3.10) on $\mathcal{S}'(\mathbb{R}^n)$. This representation restricts to a unitary irreducible representation on $L^2(\mathbb{R}^n)$.

Wigner functions

The idea of the Wigner function [178] is to find (quasi-)probability distributions in phase space which are in one-to-one correspondence with quantum states in the sense that expectation values of quantum observables should be describable in terms of this distribution.

[7] The Baker-Campbell-Hausdorff (BCH) formula states that if two operators X, Y satisfy $[[X, Y], X] = 0 = [[X, Y], Y]$, one has $e^X e^Y = e^Y e^X e^{[X,Y]}$ and $e^{X+Y} = e^X e^Y e^{-[X,Y]/2}$.

To every density operator $\rho \in \mathcal{D}\left(\bigotimes_{k=1}^{n} L^2(\mathbb{R})\right)$, let us associate its *characteristic function* defined as

$$\chi_\rho(\xi) \coloneqq \operatorname{tr}\left(\rho D(\xi)\right) = \operatorname{tr}\left(\rho\, e^{iR^T J\xi}\right) \qquad \text{for } \xi \in \mathbb{R}^{2n} \ .$$

Note that the characteristic function is well-defined for trace-class operators and that the corresponding state can be recovered from $\chi_\rho(\xi)$ via

$$\rho = \frac{1}{(2\pi)^n} \int_{\mathbb{R}^{2n}} \chi_\rho(\xi) D(\xi)^\dagger \, d^{2n}\xi \ .$$

The characteristic function itself is the symplectic Fourier transform of the *Wigner function* W_ρ, i.e., they satisfy

$$\chi_\rho(\xi) = \int_{\mathbb{R}^{2n}} e^{i\xi^T J\eta} W_\rho(\eta)\, d^{2n}\eta \ , \quad \text{and} \quad W_\rho(\xi) \coloneqq \frac{1}{(2\pi)^{2n}} \int_{\mathbb{R}^{2n}} e^{-i\xi^T J\eta} \chi_\rho(\eta)\, d^{2n}\eta$$

for $\xi \in \mathbb{R}^{2n}$. This function shares certain properties with classical probability distributions: The condition $\operatorname{tr}(\rho) = 1$ implies $\chi_\rho(0) = 1$ and hence that $W_\rho(\xi)$ is normalisable.

A useful property of the Wigner function is that it produces the correct marginal distributions of the state ρ. But the Wigner function does only define a quasi-probability distribution on the phase space (not a probability distribution) since it may take negative values. Many quantities of the bosonic quantum system are obtained in this way from the Wigner functions. For a detailed overview of this so-called Weyl calculus, the reader is referred to the literature, e.g. de Gosson's book on symplectic geometry and quantum mechanics [44].

The properties of the Wigner function $W_\rho(\xi)$ – and thereby of the respective quantum state ρ – are characterised by its moments, i.e., the derivatives of its characteristic function. The most important will be the first moment and second moments.

Definition 2.26 *(First and second moments).* Let $\mathcal{H} = L^2(\mathbb{R}^n)$ for some $n \in \mathbb{N}$ and $\rho \in \mathcal{D}(\mathcal{H})$ be a quantum state. The first moment $\overline{R}(\rho) \in \mathbb{R}^{2n}$ is called the *displacement vector* and given by

$$\overline{R}_j(\rho) \coloneqq \langle R_j \rangle_\rho = \operatorname{tr}(\rho R_j) \quad , \quad \text{for } j = 1, \ldots, 2n \ .$$

The second moment is a matrix $V(\rho) \in \mathbb{R}^{2n\times 2n}$ defined as

$$V_{jk}(\rho) \coloneqq \langle \{R_j - \overline{R}_j(\rho), R_k - \overline{R}_k(\rho)\} \rangle_\rho = \operatorname{tr}\left(\rho\{R_j - \overline{R}_j(\rho), R_k - \overline{R}_k(\rho)\}\right) \tag{2.24}$$

for $j, k = 1, \ldots, 2n$. Here $\{\cdot,\cdot\}$ denotes the anticommutator $\{X, Y\} \coloneqq XY + YX$. The matrix $V(\rho)$ is called the *covariance matrix* of state ρ.

The covariant matrix V is a symmetric $2n \times 2n$ matrix and its diagonal elements correspond to the variance of the quadrature operators

$$V_{kk}(\rho) = \langle R_k^2 \rangle_\rho - \langle R_k \rangle_\rho^2 \coloneqq \operatorname{Var}(R_j) \ .$$

It is conversely true [145] that a symmetric matrix $V \in \mathbb{R}^{2n\times 2n}$ is the covariance matrix of a quantum state if and only if it satisfies the operator inequality $V(\rho) + iJ \geq 0$. This inequality implies that for all $k = 1, \ldots, n$, we it satisfies $V_{kk}(\rho)V_{k+N,k+N}(\rho) \geq 1$, i.e., that $\operatorname{Var}(Q_k)\operatorname{Var}(P_k) \geq 1$ which corresponds to the usual Heisenberg uncertainty relation between position and momentum.

2.2.3 Gaussian states and channels

Let us describe the most prominent Gaussian states and channels in this section. Section 2.2.4 focuses on the special case of Gaussian unitary channels and establishes the connection to symplectic operations. We give a few relevant examples of Gaussian unitary channels and dilations thereof in Section 2.2.5. For a more detailed overview on the topic and applications for quantum information processing tasks, we refer the interested reader to the review articles [176, 62, 57].

Gaussian states

Definition 2.27 *(Gaussian states).* A quantum state $\rho \in \mathcal{D}(L^2(\mathbb{R}^n))$ is *Gaussian* if its Wigner characteristic function is of the form

$$\chi_\rho(\xi) := e^{-\frac{1}{4}\xi^T(JVJ^T)\xi - i\xi^T J\overline{R}} \qquad \text{for } \xi \in \mathbb{R}^{2n} , \tag{2.25}$$

i.e., it is a Gaussian function.

By definition, a Gaussian state is fully characterised by its first and second moments $\overline{R}(\rho)$ and $V(\rho)$, respectively. One writes $\rho(\overline{R}, V)$ to denote the Gaussian state with displacement vector $\overline{R}$ and covariance matrix V. Let us discuss a few examples of Gaussian states on a single mode.

Coherent states and vacuum state

For a single bosonic mode (i.e., $n = 1$), the coherent state $|\alpha\rangle \in L^2(\mathbb{R})$ is defined via the equation $a|\alpha\rangle = \alpha|\alpha\rangle$, i.e., as the eigenstate of the annihilation operator a with eigenvalue $\alpha \in \mathbb{C}$. In the number basis $\{|k\rangle\}_{k\in\mathbb{N}}$ it can therefore be written as

$$|\alpha\rangle := e^{-|\alpha|^2/2} \sum_{k=0}^{\infty} \frac{\alpha^k}{\sqrt{k!}} |k\rangle \ .$$

By the Baker-Campbell Hausdorff formula[8] for the displacement operators, its characteristic function is Gaussian (cf. Eq. (2.25)) with covariance matrix $V(|\alpha\rangle\langle\alpha|) = I_2$ and displacement vector $\overline{R}(|\alpha\rangle\langle\alpha|) = (\sqrt{2}\mathsf{Re}(\alpha), \sqrt{2}\mathsf{Im}(\alpha))$.

The *vacuum state* is given by the pure state $|0\rangle$ in the number basis. It is easy to see that this is a coherent state (for $\alpha = 0$) with displacement vector $\overline{R}(|0\rangle\langle 0|) = (0\ 0)^T$ and covariance matrix $V(|0\rangle\langle 0|) = I_2$, respectively. It is easy to see that a coherent state for general $\alpha \in \mathbb{C}$ is created from the vacuum state by a displacement in phase space. More precisely, $|\alpha\rangle = D(\xi)|0\rangle$ where $\xi^T = \big(\sqrt{2}\mathsf{Re}(\alpha)\ ,\ \sqrt{2}\mathsf{Im}(\alpha)\big)$.

[8] The Baker-Campbell-Hausdorff (BCH) formula states that for two operators X, Y satisfy $[[X,Y],X] = 0 = [[X,Y],Y]$ then the relations $e^X e^Y = e^Y e^X e^{[X,Y]}$ and $e^{X+Y} = e^X e^Y e^{-[X,Y]/2}$ hold.

Thermal state

On a finite-dimensional system with Hilbert space dimension $d \in \mathbb{N}$, one may define a maximally mixed state as

$$\rho_{\text{mix}} \coloneqq \frac{1}{d} \sum_{k=1}^{d} |k\rangle\langle k| \ .$$

One cannot simply consider $d \to \infty$ since this 'state' would have infinite energy expectation value (which would be equal to $d/2$ plus its number operator expectation value) and hence be unphysical. To generalise this state to the infinite-dimensional Hilbert space $L^2(\mathbb{R})$, one fixes its mean photon number $\overline{n} \coloneqq \text{tr}(\rho a^\dagger a)$ (also called the state's average energy). The *thermal state*

$$\rho_{\text{th},\overline{n}} \coloneqq \frac{1}{1+\overline{n}} \sum_{k=1}^{\infty} \left(\frac{\overline{n}}{1+\overline{n}} \right)^k |k\rangle\langle k| \ , \tag{2.26}$$

is a well-defined state in the sense that its mean photon number $\overline{n}$ is finite. One can write $\rho_{\text{th},\overline{n}}$ as a Gaussian mixture of coherent states or in terms of the inverse temperature β as

$$\rho_{\text{th},\overline{n}} \coloneqq \frac{1}{\pi\overline{n}} \int |\alpha\rangle\langle\alpha| e^{-\frac{|\alpha|^2}{\overline{n}}} d^2\alpha = Z^{-1} \sum_{k=1}^{\infty} e^{-\beta k} |k\rangle\langle k|$$

for a suitable normalisation constant Z. Its characteristic function is given by

$$\chi_{\rho_{\text{th}}}(\xi_1, \xi_2) = \text{tr}\left[\frac{1}{\pi\overline{n}} \int |\alpha\rangle\langle\alpha| e^{-\frac{|\alpha|^2}{\overline{n}}} D(\xi) d^2\alpha \right] = e^{-\frac{1}{4}(\xi_1^2+\xi_2^2)(1+2\overline{n})} \ .$$

Gaussian channels: general formalism

On a system of $n \in \mathbb{N}$ bosonic modes let us the notions of a general quantum channel and a Gaussian quantum channel.

By Definition 2.14 a quantum channel on n bosonic modes is a CPTP map. Using the Stinespring dilation theorem (Definition 2.18 and [153]), it can be expressed in terms of a unitary evolution U of an input state $\rho \in \mathcal{D}(L^2(\mathbb{R}^n))$ and a pure state $|\psi\rangle\langle\psi|_E$ on n_E additional modes as

$$\Phi : \mathcal{D}(L^2(\mathbb{R}^n)) \to \mathcal{D}(L^2(\mathbb{R}^n)) \quad , \quad \Phi(\rho) = \text{tr}_E \left[U \left(\rho \otimes |\psi\rangle\langle\psi|_E \right) U^\dagger \right] \ . \tag{2.27}$$

We note that this representation is unique up to partial isometries [128]. Since Gaussian states are ubiquitous in quantum optics we are primarily interested in channels which map Gaussian states to Gaussian states.

Definition 2.28 *(Gaussian channels).* Let $\mathcal{H} = L^2(\mathbb{R}^n)$ be the Hilbert space of $n \in \mathbb{N}$ bosonic modes. A *Gaussian channel* is a quantum channel $\Phi : \mathcal{D}(\mathcal{H}) \to \mathcal{D}(\mathcal{H})$, such that $\Phi(\rho)$ is Gaussian for every Gaussian state ρ.

A Gaussian channel can be fully characterised by its action on Gaussian states. It acts as $\Phi(\rho(\overline{R}, V)) = \rho(\overline{R}', V')$ where the covariance matrix V and the displacement vector $\overline{R}$ of the Gaussian state are transformed as [83]

$$\overline{R}' = X\overline{R} + v \ , \tag{2.28}$$

$$V' = XVX^T + Y \ . \tag{2.29}$$

Here, $X, Y \in \mathbb{R}^{2n\times 2n}$ are two real matrices, Y is symmetric, and $v \in \mathbb{R}^{2n}$ is a vector. The map Φ is completely positive if and only if the matrices X, Y are related to each other via

$$Y + i(J - X^T J X) \geq 0 \ . \tag{2.30}$$

Conversely, every tuple (X, Y, v) which satisfies $Y = Y^T$ and (2.30) defines a Gaussian quantum channel, cf. [57].

2.2.4 Gaussian unitary channels and symplectic operations

Of special interest are those Gaussian channels which are reversible. They describe linear optics, are hence ubiquitous in quantum optics, and such operations will be used later in this thesis as resources for our pulse sequences and in the encoding of a qubit into a harmonic oscillator. Furthermore, they admit a simplified representation on the phase space associated with the bosonic quantum system.

Consider a Gaussian unitary channels Φ_U which can be described by a unitary operator $U \in \mathsf{U}(L^2(\mathbb{R}^{2n}))$ such that $\Phi_U(\rho) = U\rho U^\dagger$. Since the unitary is generated by some Hamiltonian, it can be written as $U = e^{-iH}$ for some $H \in \mathcal{L}_{\mathrm{sa}}(L^2(\mathbb{R}^{2n}))$. In order to preserve the Gaussian nature of states, the Hamiltonian H must be a second order polynomial in the bosonic field operators $a \coloneqq (a_1, \ldots, a_n)$ and $a^\dagger \coloneqq (a_1^\dagger, \ldots, a_n^\dagger)$, i.e., its most general form is

$$H = \alpha a + (a^\dagger)^T F a + (a^\dagger)^T G a^\dagger + cI + \text{h.c.} \tag{2.31}$$

for $\alpha \in \mathbb{C}$, and $F, G \in \mathbb{C}^{n\times n}$ symmetric and where h.c. denotes the complex conjugate of the previous terms. Then, U (or more precisely the dual channel of Φ_U) transforms the annihilation operators as $U^\dagger a U = Aa + Ba^\dagger + \alpha$ where $A, B \in \mathbb{C}^{n\times n}$ are matrices which satisfy $AB^T = BA^T$ and $AA^\dagger - BB^\dagger = I$ as a consequence of the commutation relations between creation and annihilation operators. The action of the dual channel Φ_U on the quadrature operators R from Eq. (2.20) can be computed to be (cf. e.g. [62, 176])

$$U^\dagger R_j U = \sum_{k=1}^{2n} S_{jk} R_k + d_j \tag{2.32}$$

where $S \in \mathbb{R}^{2n\times 2n}$, and $d \in \mathbb{R}^{2n}$. If the unitarily transformed quadratures (2.32) satisfy the commutation relations (2.16), the matrix S in Eq. (2.32) is an element of the symplectic group $\mathsf{Sp}(2n, \mathbb{R})$.

Definition 2.29 *(Symplectic group and Lie algebra).* The *symplectic group*

$$\mathsf{Sp}(2n, \mathbb{R}) \coloneqq \left\{ S \in \mathbb{R}^{2n\times 2n} \mid SJS^T = J \right\}$$

consists of all real $2n \times 2n$ matrices that preserve the symplectic form. The *symplectic Lie algebra* is given by

$$\mathfrak{sp}(2n, \mathbb{R}) \coloneqq \left\{ X \in \mathbb{R}^{2n\times 2n} \mid X^T J + JX = 0 \right\} \ .$$

By Eq. (2.32), a Gaussian unitary $U \in \mathsf{U}(L^2(\mathbb{R}^n))$ can be described by an affine symplectic map (d, S) where $d \in \mathbb{R}^{2n}$ and $S \in \mathsf{Sp}(2n, \mathbb{R})$. We write $U_{(d,S)}$ to denote a Gaussian unitary corresponding to the affine symplectic map (d, S) and simply U_S if the affine symplectic map is of the form $(0, S)$. Note that the correspondence $S \mapsto U_S$ is not unique because Eq. (2.32)

(for $d = 0$) leaves freedom for a S-dependent phase factor. There is actually only a one-to-one correspondence $S \mapsto \pm U_S$, yielding a two-valued unitary representation of $\mathsf{Sp}(2n, \mathbb{R})$. But note that this sign ambiguity in U_S is irrelevant when considering a unitary conjugation e.g. for the evolution of states $\rho \mapsto U\rho U^\dagger$ or operators $A \mapsto U^\dagger A U$ and at the level of the unitary Weyl displacements 2.25.

Theorem 2.30 *(Metaplectic representation and further properties).* There is a faithful unitary representation of the metaplectic group $\mathsf{Mp}(2n, \mathbb{R})$ – a two-fold covering of $\mathsf{Sp}(2n, \mathbb{R})$ – called the *metaplectic representation*.

Furthermore, the exponential map $\mathfrak{sp}(2n, \mathbb{R}) \to \mathsf{Sp}(2n, \mathbb{R})$ is not surjective, but every $S \in \mathsf{Sp}(2n, \mathbb{R})$ can be written as the product of two exponentials of $\mathfrak{sp}(2n, \mathbb{R})$.

The group of symplectic orthogonal matrices satisfies

$$\mathsf{Sp}(2n, \mathbb{R}) \cap \mathrm{O}(2n, \mathbb{R}) \cong \mathsf{U}(n) \tag{2.33}$$

where $\mathrm{O}(2n, \mathbb{R}) := \{O \in \mathbb{R}^{2n \times 2n} \mid O^T O = O O^T = I_{2n}\}$ is the group of orthogonal $(2n \times 2n)$-matrices and $\mathsf{U}(n)$ is the Lie group of unitary $(n \times n)$-matrices.

A symplectic matrix $S \in \mathsf{Sp}(2n, \mathbb{R})$ can be decomposed as

$$S = O_1 Z O_2 \quad \text{where} \quad O_1, O_2 \in \mathsf{Sp}(2n, \mathbb{R}) \cap \mathrm{O}(2n, \mathbb{R}),\ Z = D \oplus D^{-1} \quad ,$$

where D is positive-definite and diagonal. □

For the proof as well as an overview of further properties of the symplectic group, we refer to a review by Arvind, Dutta, Mukanda, and Simon [53]. In slight abuse of notation the map $S \mapsto U_S$ by Eq. (2.32) (for $d = 0$) is called the *metaplectic representation* of $\mathsf{Sp}(2n, \mathbb{R})$.

Let us discuss a few properties of Gaussian unitaries. The Gaussian unitary $U_{(d,S)}$ transforms the displacement operator $D(\xi)$ from Eq. (2.22) as

$$\begin{aligned} U_{(d,S)}^\dagger D(\xi) U_{(d,S)} &= e^{i(SR+d)^T J\xi} = D(S^{-1}\xi) e^{id^T J\xi} \ , \\ U_{(d,S)} D(\xi) U_{(d,S)}^\dagger &= e^{iR^T (S^{-1})^T J\xi - id^T (S^{-1})^T J\xi} = D(S\xi) e^{-id^T JS\xi} . \end{aligned} \tag{2.34}$$

To compute this we used that $S^T J = J S^{-1}$ and $(S^{-1})^T J = JS$ which follows from the property that S is symplectic. Similarly, we can ask how the Wigner function and the characteristic function of the evolved state $U_{(d,S)} \rho U_{(d,S)}^\dagger$ look like. Direct computation shows that for $\xi \in \mathbb{R}^{2n}$

$$\chi_{U_{(d,S)}\rho U_{(d,S)}^\dagger}(\xi) = \mathrm{tr}\left[D(\xi) U_{(d,S)} \rho U_{(d,S)}^\dagger\right] = \mathrm{tr}\left[D(S^{-1}\xi) e^{id^T J\xi} \rho\right] = \chi_\rho(S^{-1}\xi) e^{id^T J\xi}$$

and similarly that

$$W_{U_{(d,S)}\rho U_{(d,S)}^\dagger}(\xi) = \frac{1}{(2\pi)^{2n}} \int_{\mathbb{R}^{2n}} e^{-i\xi^T J\eta}\ e^{id^T J\eta}\ \chi_\rho(S^{-1}\eta)\, d^{2n}\eta = W_\rho(S^{-1}(\xi - d)) \ .$$

If an initial state $\rho(\overline{R}, V)$ is Gaussian that undergoes a Gaussian unitary channel characterised by the affine symplectic map (d, S) then the transformed state

$$\rho(\overline{R}', V') = U_{(d,S)} \rho(\overline{R}, V) U_{(d,S)}^\dagger$$

has the following first and second moments:

$$\begin{aligned}\overline{R}'_j &\coloneqq \mathrm{tr}\left(U_{(d,S)}\rho(\overline{R},V)U^\dagger_{(d,S)}R_j\right) = \sum_k S_{jk}\overline{R}_k + d_j \ ,\\ V'_{jk} &\coloneqq \mathrm{tr}\left(U_{(d,S)}\rho(\overline{R},V)U^\dagger_{(d,S)}\{R_j-\overline{R}'_j, R_k\overline{R}'_k\}\right) = \sum_{l,m} S_{jl}V_{lm}S_{km} \ ,\end{aligned}$$

for $j,k = 1,\dots 2n$, i.e., one finds

$$\overline{R}' = S\overline{R} + d \qquad , \qquad V' = SVS^T \ . \tag{2.35}$$

Comparing Eq. (2.28) and (2.29) to Eqs. (2.35) shows that a unitary Gaussian channel corresponds to a Gaussian channel for the parameters $Y = 0$, $X = S$ symplectic and the vector $v = d$.

A variant of the formula (2.32), which is commonly used throughout this thesis, can be derived in the special case of quadratic Hamiltonians, i.e., where H from Eq. (2.31) depends at most quadratically on the quadrature operators R.

Lemma 2.31. Assume that the Hamiltonian is quadratic in the mode operators, i.e.,

$$H = \frac{1}{2}R^TAR = \frac{1}{2}\sum_{j,k=1}^{2n} A_{jk}R_jR_k \ . \tag{2.36}$$

Then the unitary $U = e^{-iH}$ transforms the quadrature operators as

$$U^\dagger R_j U = \sum_{k=1}^{2n} \left(e^{JA}\right)_{jk} R_k \qquad \text{for } j = 1,\dots,2n \ . \tag{2.37}$$

□

Proof. For the Hamiltonian (2.36), the associated unitary is $U \coloneqq e^{-iH} = e^{-i\frac{1}{2}R^TAR}$. In order to compute $U^\dagger R_j U$, we apply an important lemma to the Baker-Campbell-Hausdorff formula (called the Hadamard lemma or Lie expansion formula, cf. e.g. [75, Proposition 3.3.5]) which states that for linear operators B, C

$$e^BCe^{-B} = C + [B,C] + \frac{1}{2!}[B,[B,C]] + \cdots = \sum_{m=0}^{\infty}\frac{[B,C]_m}{m!} \tag{2.38}$$

where $[B,C]_m = [B,[B,C]_{m-1}]$ and $[B,C]_0 = C$. Here, one sets $B = \frac{i}{2}R^TAR$ and $C = R_l$. By direct computation and induction one obtains that

$$\begin{aligned}[B,C] &= \frac{i}{2}\sum_{jk} A_jk[R_jR_k,R_l] = -\sum_{jk} A_{jk}R_jJ_{kl} = (-R^TAJ)_l \ ,\\ [B,C]_m &= (R^T(-AJ)^m)_l \ ,\end{aligned} \tag{2.39}$$

which implies, together with Eq. (2.38), that

$$U_S^\dagger R_l U_S \coloneqq e^{i\frac{1}{2}R^TAR}R_le^{-i\frac{1}{2}R^TAR} = \sum_{n=0}^{\infty}\frac{\left(R^T(-AJ)^n\right)_l}{n!} = \left(R^Te^{-AJ}\right)_l \ .$$

This is equal to $U_S^\dagger R_l U_S = \sum_j S_{lj} R_j$ if and only if $S^T = e^{-AJ}$, i.e., if $S = e^{JA}$ as claimed. Let us remark that if the Hamiltonian contains an additional linear term in the mode operators, i.e., if $H = \frac{1}{2} R^T A R + b^T R$ for $b \in \mathbb{R}^{2n}$, then the same calculation gives

$$U_S^\dagger R_l U_S = \sum_j S_{lj} R_j + d_j \qquad \text{where} \qquad d_j = \sum_{m=0}^{\infty} \frac{\left(b^T J (-AJ)^m\right)_j}{(m+1)!}$$

and $S = e^{JA}$ as before. ■

We note that Eq. (2.37) corresponds to Eq. (2.32) for the special case of $d = 0$ and $S = e^{JA}$ for A symmetric. Lemma 2.31 shows that if $d = 0$, then the symplectic matrix S in Eq. (2.32) can be written as the exponential of $JA \in \mathfrak{sp}(2n, \mathbb{R})$ which is related to the unitary U_S via $U_S = e^{-\frac{i}{2} R^T A R}$. Note that not all symplectic matrices can be written in the form e^{JA} but that all e^{JA} for $A = A^T$ are symplectic.

One furthermore distinguishes two classes of Gaussian unitaries.

Definition 2.32 *(Passive and active Gaussians).* A Hamiltonian H of the form of Eq. (2.36) is called *passive* if and only if it commutes with the 'free' Hamiltonian

$$H_0 \coloneqq \frac{1}{2} R^T R = \frac{1}{2} \sum_{j=1}^{n} (Q_j^2 + P_j^2) \,. \tag{2.40}$$

The generated unitary $U = e^{-iH}$ is called a *passive Gaussian unitary*. All other Gaussian unitaries are called *active*.

Since a passive Hamiltonian commutes with H_0, it preserves the photon number/energy. The symplectic matrix S associated with a passive Gaussian unitary is orthogonal, i.e., it belongs to $\mathsf{Sp}(2n, \mathbb{R}) \cap \mathrm{O}(2n, \mathbb{R})$ from (2.33). As a consequence, it preserves the trace of the covariance matrix $\mathrm{tr}(SVS^T) = \mathrm{tr}(V)$. Passive Gaussian unitaries can be constructed from beam splitters and phase rotations only [138], two examples of Gaussian channels that will be presented in the next section.

2.2.5 Examples of Gaussian unitary channels

In this section, we give a few examples of pairs of Gaussian unitaries U_S and associated symplectic matrices $S = e^{JA}$ that are related to each other as described in Lemma 2.31, i.e., via Eq. (2.37). To compute the symplectic matrices S, we follow the same steps as in the above mentioned Lemma 2.31, i.e., we use Eqs. (2.38) and (2.39) to compute $U^\dagger R_j U$. The presented examples are prominent Gaussian unitaries ubiquitous in quantum optics and they are used in Chapters 4 and 5 of this thesis. More precisely, the pulses in our novel CV pulse sequences as well as the additional GKP encoding are constructed from these unitaries.

Single-mode phase space rotations

The first example is a phase space rotation unitary $U_{\mathrm{rot}}(\zeta)$ about the angle ζ. Here the associated Hamiltonian $H_{\mathrm{rot}}(\zeta)$ is proportional to the free Hamiltonian H_0 from Eq. (2.40)

for $n = 1$, i.e., $H_{\mathrm{rot}}(\zeta) = \zeta/2(Q^2 + P^2)$ for $\zeta \in \mathbb{R}$ and

$$U_{\mathrm{rot}}(\zeta) = e^{-i\frac{\zeta}{2}(Q^2+P^2)} \ . \tag{2.41}$$

Direct computation using Eqs. (2.38) and (2.39) shows that the symplectic matrix associated with (2.41) is given by

$$S_{\mathrm{rot}}(\zeta) = \begin{pmatrix} \cos(\zeta) & \sin(\zeta) \\ -\sin(\zeta) & \cos(\zeta) \end{pmatrix} \ . \tag{2.42}$$

This is a rotation by the angle ζ in the phase space. Two specific examples are the *sign flip* – the π-rotation – with associated symplectic matrix and unitary

$$S_{\mathrm{rot}}(\pi) = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{and} \quad U_{\mathrm{rot}}(\pi) = e^{-i\frac{\pi}{2}(Q^2+P^2)} \ , \tag{2.43}$$

respectively and the $(\zeta = \pi/2)$-rotation for which one has

$$S_{\mathrm{rot}}(\pi/2) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \text{and} \quad U_{\mathrm{rot}}(\pi/2) = e^{-i\frac{\pi}{4}(Q^2+P^2)}$$

We note that all such $U_{\mathrm{rot}}(\zeta)$ are passive single-mode Gaussian unitaries.

Single-mode active gates

For $\varphi \in \mathbb{R}$, consider the unitary $U_{\mathrm{act,Q}}(\varphi) = e^{-i\frac{\varphi}{2}Q^2}$. By the Baker-Campbell-Hausdorff formula (Eqs. (2.38) and (2.39)) one finds that the corresponding symplectic matrix

$$S_{\mathrm{act,Q}}(\varphi) = \begin{pmatrix} 1 & 0 \\ -\varphi & 1 \end{pmatrix} \ . \tag{2.44}$$

Similarly, one obtains for $U_{\mathrm{act,P}}(\varphi) \coloneqq e^{-i\varphi/2P^2}$ that

$$S_{\mathrm{act,P}}(\varphi) = \begin{pmatrix} 1 & \varphi \\ 0 & 1 \end{pmatrix} \ .$$

The unitaries $U_{\mathrm{act,P}}(\varphi)$ and $U_{\mathrm{act,P}}(\varphi)$ are active for $\varphi \neq 0$ since $[Q^2, Q^2 + P^2] \neq 0$.

Single-mode squeezing

Another relevant example of a single-mode Gaussian unitary is the so-called squeezing. Consider the Hamiltonian

$$H_{\mathsf{sq}}(\vartheta) = \frac{\vartheta}{2}\,(QP + PQ)$$

for $\vartheta \in \mathbb{R}$ which generates the unitary

$$U_{\mathsf{sq}}(\vartheta) = e^{-iH_{\mathsf{sq}}(\vartheta)} = e^{-i\frac{\vartheta}{2}(QP+PQ)} \ . \tag{2.45}$$

Again, by the Baker-Campbell-Hausdorff formula (Eqs. (2.38) and (2.39)) one finds that the associated symplectic matrices is given by

$$S_{\mathsf{sq}}(\vartheta) = \begin{pmatrix} e^{\vartheta} & 0 \\ 0 & e^{-\vartheta} \end{pmatrix} \ .$$

Such matrices $S_{\mathsf{sq}}(\vartheta)$ and unitaries $U_{\mathsf{sq}}(\vartheta)$ are usually called (pure) *squeezing matrices* or *squeezing unitaries*, respectively, since they extend a phase space region along one axis and shrink it along the other axis while keeping the volume constant. The squeezing is in terms of the squeezing parameter $r = e^{\vartheta}$, i.e., as

$$S_{\mathsf{sq}}\big(\ln(r)\big) = \begin{pmatrix} r & 0 \\ 0 & \frac{1}{r} \end{pmatrix} . \tag{2.46}$$

This is also a single-mode active unitary.

Two mode beam splitter

Let us consider two examples of two-mode Gaussian unitaries. The *beam splitter* unitary

$$U_{\mathsf{bs}}(\beta) = e^{-i\beta(Q_1P_2 - Q_2P_1)} \tag{2.47}$$

for $\beta \in \mathbb{R}$ is generated by the beam splitter Hamiltonian $H_{\mathsf{bs}}(\beta) = \beta\,(Q_1P_2 - Q_2P_1)$. The matrix $A_{\mathsf{bs}}(\beta)$ such that $H_{\mathsf{bs}}(\beta) = \frac{1}{2}R^T A_{\mathsf{bs}}(\beta)R$ is given by

$$A_{\mathsf{bs}}(\beta) = \beta \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} .$$

Direct calculation (using Eqs. (2.38) and (2.39)) shows that the mode operators transform as

$$\begin{aligned} U_{\mathsf{bs}}(\beta)^\dagger Q_1 U_{\mathsf{bs}}(\beta) &= \cos(\beta)Q_1 - \sin(\beta)Q_2 \ , \\ U_{\mathsf{bs}}(\beta)^\dagger Q_2 U_{\mathsf{bs}}(\beta) &= \cos(\beta)Q_2 + \sin(\beta)Q_1 \ , \\ U_{\mathsf{bs}}(\beta)^\dagger P_1 U_{\mathsf{bs}}(\beta) &= \cos(\beta)P_1 - \sin(\beta)P_2 \ , \\ U_{\mathsf{bs}}(\beta)^\dagger P_2 U_{\mathsf{bs}}(\beta) &= \cos(\beta)P_2 + \sin(\beta)P_1 \ . \end{aligned}$$

One often quantifies the beam splitter by its transmittivity η which is related to the parameter $\beta \in \mathbb{R}$ by $\cos(\beta) = \sqrt{\eta}$ and $\sin(\beta) = \sqrt{1-\eta}$. In terms of the transmittivity, the beam splitter matrix $S_{\mathsf{bs}}(\beta)$ associated with the unitary (2.47) is

$$S_{\mathsf{bs}}\big(\arccos(\sqrt{\eta})\big) = \begin{pmatrix} \sqrt{\eta} & -\sqrt{1-\eta} & 0 & 0 \\ \sqrt{1-\eta} & \sqrt{\eta} & 0 & 0 \\ 0 & 0 & \sqrt{\eta} & -\sqrt{1-\eta} \\ 0 & 0 & \sqrt{1-\eta} & \sqrt{\eta} \end{pmatrix} . \tag{2.48}$$

Dilations of Gaussian channels

Recall from Section 2.2.3 that a general n-mode Gaussian channel

$$\mathcal{N} : \mathcal{D}\left(L^2(\mathbb{R}^n)\right) \to \mathcal{D}\left(L^2(\mathbb{R}^n)\right)$$

is fully characterised by the triple (X, Y, v). Written as a Stinespring dilation of the form of Eq. (2.27), one can easily see that this channel is Gaussian if U is Gaussian and the environment state is Gaussian as well. This is actually sufficient: an arbitrary n-mode Gaussian channel can be represented by a dilation Eq. (2.27) where U is a Gaussian unitary, $|\psi\rangle_E$ is pure Gaussian and furthermore $n_E \leq 2n$, cf. [35, 36].

Let us restrict to a single bosonic mode and present a few relevant examples of Gaussian unitary dilations (for a full characterisation of single-mode Gaussian channels we refer to the survey [176]): Define the channel

$$\mathcal{N}(\rho) = \mathrm{tr}_E \left(U_S(\rho \otimes \sigma) U_S^\dagger \right) \quad , \quad \text{for all } \rho \in \mathcal{D}(L^2(\mathbb{R}^n)) \ .$$

for a Gaussian unitary U_S associated with the symplectic matrix $S \in \mathsf{Sp}(2n_S + 2n_E, \mathbb{R})$ and a thermal state (cf. Eq. (2.26)) $\sigma \in \mathcal{D}(L^2(\mathbb{R}^{n_E}))$ of mean photon number $\overline{n}$.

If U_S is the beam splitter unitary (with associated symplectic matrix $S_{\mathsf{bs}}(\arccos(\sqrt{\eta}))$ for $0 < \eta < 1$ from Eq. (2.48)), then this channel $\mathcal{N}_{\eta,\overline{n}}$ is called the thermal attenuation channel and fully characterised by

$$X = \sqrt{\eta} I \quad , \quad Y = (1-\eta)(2\overline{n}+1) I \ .$$

2.3 Quantum computing with qubits

The idea of quantum computing has been pioneered in the 1980es with proposals of a quantum Turing machine by Benioff [18] and a quantum simulator by Feynman [63]. Since then, the field has attracted immense theoretical and experimental interest as it bears the potential of substantially outperforming classical computers for different relevant tasks. Currently, major obstacles such as errors caused by decoherence processes have hindered the realisation of a physical quantum computer with a sufficiently large number of reliable and controllable qubits on which useful computations can be performed. We are hence said to be in the noisy intermediate-scale era of quantum computing where first approaches of small devices with a limited amount of control are realistic in the near future. Recent advances include the famous claim of quantum advantage by Google [13] from 2019.

A quantum computer is a device which uses the properties of quantum mechanics to perform computations. Information is encoded in a quantum state on which one may run a quantum algorithm which is a sequence of instructions performed in order to find the solution to a particular computational problem. Such algorithms promise speed-ups compared to known algorithms that can be run on a classical computer. How they are incorporated in the physical processes, i.e., in the state preparation, in the transformations or in the measurement, depends on the model of computation. The most prominent model in which quantum computation is considered is the quantum circuit model. Here we focus on this model, although there exist other proposals: The quantum Turing machine proposed by Benioff [18] as mentioned above generalises the mathematical description of a (classical) Turing machine. As its name suggests, in measurement-based quantum computation [136], an algorithm is incorporated in the measurement: one prepares the system in a highly entangled quantum state, a so-called cluster state (note that one picks the same cluster state for all possible algorithms) and a sequence of measurements, which is performed on this cluster state, defines the specific algorithm. In contrast, adiabatic quantum computation [61] uses a potentially very complicated target Hamiltonian to describe the desired algorithm or computation: here, one starts with a (usually easy to prepare) ground state of an initial Hamiltonian and adiabatically drives the system from this state to the ground state of the target Hamiltonian, which is possible under certain conditions due to the adiabatic theorem. For a more detailed presentation of these approaches to quantum computation we refer to the literature such as [121].

Quantum computation is closely related to quantum information theory – a field where one studies how to quantify quantum information and describe its transformation when the un-

derlying carrier – a quantum state – is transformed. But this thesis does not go into details of quantum information and the interested reader is referred to the standard literature such as the textbook by Nielsen and Chuang [121].

2.3.1 Quantum circuit model

Let us return to the circuit model of quantum computation which we describe in more detail. One usually picks the qubit as a basic building block of quantum computation, i.e., one considers the circuit model on $n \in \mathbb{N}$ qubits.

One initially prepares the system in an n-qubit state. We think of the quantum computer as a closed quantum system where all quantum channels that are used for the computation act on the system only and are reversible, i.e., the computation is modelled by a unitary operation – a quantum gate – acting on the n qubits. The read-out of the computation result consists of measuring every single qubit in the computational basis. The computation, the n-qubit gate, can equivalently be written as the product of many gates, each of which may act on less than n qubits. In many realisations, the hardware cannot implement any n-qubit gate but only a specific subset thereof. This gate set is typically small and may only include gates which act non-trivially on a small number of qubits. Then, the computation is modelled by a sequence of such quantum gates.

One starts by initialising a pure n-qubit state $|\psi\rangle = |0\rangle^{\otimes n} \in (\mathbb{C}^2)^{\otimes n}$. which encodes the quantum information. We call this initial state the n-qubit register. Then, a sequence of gates $\{U_i\}_{i=1}^k \subset \mathsf{U}_{\text{gate set}}\big((\mathbb{C}^2)^{\otimes n}\big)$ is applied to the initial state, i.e.,

$$|\psi\rangle \mapsto U_k U_{k-1} \cdots U_1 |\psi\rangle \ .$$

These gates are taken from the set $\mathsf{U}_{\text{gate set}}\big((\mathbb{C}^2)^{\otimes n}\big) \subset \mathsf{U}\big((\mathbb{C}^2)^{\otimes n}\big)$ of possibly implementable gates with the considered hardware. At every step of the computation, the quantum state is pure. The output of the computation is given by the measurement outcome of a projective measurement in the computational basis on the state $U_k U_{k-1} \cdots U_1 |\psi\rangle$ at the end of the circuit.

Quantum gates and universality

As mentioned before, a central part of the toolkit of the quantum circuit model consists of quantum gates. We are especially interested in sets of quantum gates whose products approximately generate the entire unitary group on n qubits.

Definition 2.33 *(Gates and universality).* Consider a quantum system with associated Hilbert space $\mathcal{H}_S$. A *quantum gate* is a unitary operator $U \in \mathsf{U}(\mathcal{H}_S)$.

A family of quantum gates $\{U_i\}_{i\in I}$ is *universal* if every unitary $U \in \mathsf{U}(\mathcal{H}_S)$ can be approximated using a finite number of these gates with arbitrary precision, i.e., if for all $U \in \mathsf{U}(\mathcal{H}_S)$ and for all $\epsilon > 0$, there exists a family of indices $i_1, i_2, \ldots, i_k \in I$ such that

$$\|U - U_{i_1} U_{i_2} \cdots U_{i_k}\| < \epsilon \ . \tag{2.49}$$

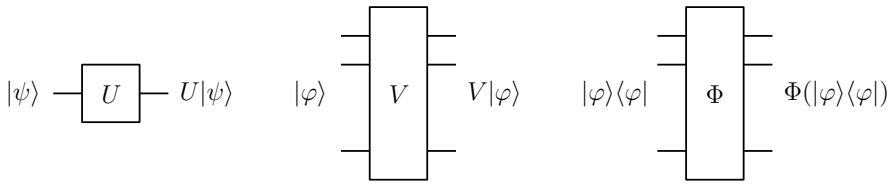


Figure 2.2: Quantum circuits of a single-qubit gate U, a multi-qubit gate V and a quantum channel Φ (from left to right): the single-qubit unitary $U \in \mathsf{U}(\mathbb{C}^2)$ transforms the pure state $|\psi\rangle \in \mathbb{C}^2$ as $U|\psi\rangle$; the multi-qubit unitary $V \in \mathsf{U}\big((\mathbb{C}^2)^{\otimes n}\big)$ transforms the state $|\varphi\rangle \in (\mathbb{C}^2)^{\otimes n}$ to the pure state $V|\varphi\rangle \in (\mathbb{C}^2)^{\otimes n}$. Pure input states – depicted as horizontal lines – are transformed by gates into pure output states. The output of a multi-qubit gate needs not to be in a tensor product form even if the input was. In slight abuse of notation, one may include general quantum channels Φ in the circuit model as on the right, even though they are not realised by gates and the output may no longer be in a pure state.

The expression on the left hand side of (2.49) is given by the operator norm of the difference of U and $U_{i_1}U_{i_2}\cdots U_{i_k}$, i.e., as

$$\|U - U_{i_1}U_{i_2}\cdots U_{i_k}\| \coloneqq \max_{|\psi\rangle\in\mathcal{H},\|\psi\|=1} \|(U - U_{i_1}U_{i_2}\cdots U_{i_k})|\psi\rangle\| \ . \tag{2.50}$$

The approximate implementation of the desired unitary has the following operational interpretation. Assume that instead of a (desired) unitary U, we have implemented the unitary V. We can interpret the norm difference $\|U - V\|$ with the norm from Eq. (2.50) as a computation error measure in the sense that if $\|U - V\|$ is small, then the measurement statistics on the states $U|\psi\rangle$ and $V|\psi\rangle$ are similar. More precisely, let M be an element of an arbitrary PVM with associated outcome m and p_U and p_V denote the probabilities of obtaining the outcome m of the PVM measured in the state $U|\psi\rangle$ and $V|\psi\rangle$, respectively. Then $|p_U - p_V| \leq 2\|U - V\|$.

Let us consider universal gate sets on an n-qubit system. They can be constructed from single-qubit and two-qubit gates. The term single-qubit gate stands for a unitary which acts non-trivially only on a single qubit and as identities on the others. Similarly, unitaries which act non-trivially on two qubits are called two-qubit gates. Within the quantum circuit model, we depict gates as shown in Fig. 2.2.

Single-qubit gates

Let us consider single-qubit gates. They correspond to unitary 2×2 matrices such as the Pauli matrices σ_x, σ_y and σ_z from Section 2.1.3. These are prominent single-qubit gates which we also call X, Y, and Z, respectively. Another relevant single qubit gate is the *Hadamard gate*

$$H \coloneqq \frac{1}{\sqrt{2}}(X + Z) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \ . \tag{2.51}$$

One can easily see it is unitary and hermitian. It transforms the Z-eigenvectors into the X-eigenvectors and vice-versa. Computation shows that

$$HXH^\dagger = Z \qquad , \qquad HYH^\dagger = -Y \qquad , \qquad HZH^\dagger = X \ . \tag{2.52}$$

Phase rotations by an angle $\vartheta \in [0, 2\pi)$ are given by

$$M_\vartheta \coloneqq e^{i\vartheta}\begin{pmatrix} e^{-i\vartheta} & 0 \\ 0 & e^{i\vartheta} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i2\vartheta} \end{pmatrix} \tag{2.53}$$

(up to a complex phase $e^{i\vartheta}$). The phase rotation M_ϑ for $\vartheta = \pi/4$

$$S \coloneqq M_{\pi/4} \coloneqq \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/2} \end{pmatrix} \tag{2.54}$$

transforms X, Y and Z as

$$SXS^\dagger = Y = iXZ \quad , \quad SYS^\dagger = iY \quad , \quad SZS^\dagger = Z \ , \tag{2.55}$$

and is called the *phase gate*. The $(\pi/8)-$*gate* is given by

$$T \coloneqq M_{\pi/8} \coloneqq \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix} \ . \tag{2.56}$$

Multi-qubit gates

Generally, an n-qubit gate is given by a unitary $U \in \mathsf{U}\big((\mathbb{C}^2)^{\otimes n}\big)$ on n qubits. One example are so-called controlled operations. Such gates come into play whenever one wants to apply a gate on a qubit conditioned on the state of another qubit. Their classical counterparts are logical operations of the form "if one bit takes the value 1 (i.e., the bit value is true), then do a certain operation to another bit". For $U \in \mathsf{U}(\mathbb{C}^2)$, the controlled-$U$ gate – abbreviated CU – has one control qubit and one target qubit. Its matrix representation in the computational basis is given by $\mathrm{CU} \coloneqq |0\rangle\langle 0| \otimes I_{2\times 2} + |1\rangle\langle 1| \otimes U = I_{2\times 2} \oplus U$, i.e., it acts as the identity (or U) on the target if the control qubit is in state $|0\rangle$ (or $|1\rangle$, respectively). The first qubit corresponds to the control qubit and the second one to the target qubit (and we write UC if the roles of control and target qubit are reversed). One of the most prominent controlled gates is the *controlled-NOT* or simply the *CNOT gate*

$$\mathsf{CNOT} \coloneqq |0\rangle\langle 0| \otimes I_{2\times 2} + |1\rangle\langle 1| \otimes \sigma_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \ ,$$

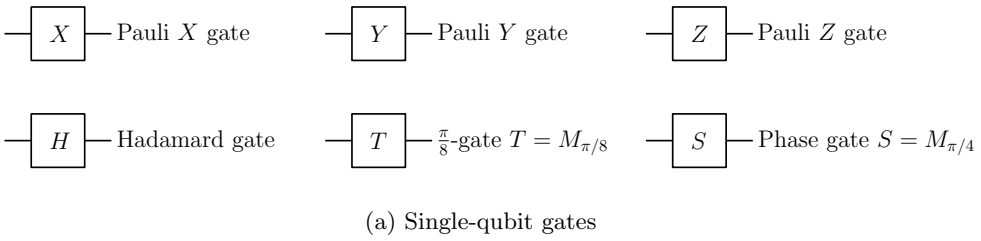
i.e., it corresponds to the controlled-X gate. The state of the target qubit is inverted if the control qubit is in the $|1\rangle$-state. This gate transforms the Pauli matrices X and Z as

$$\begin{aligned} \mathsf{CNOT} X_1 \mathsf{CNOT}^\dagger &= X_1 X_2 \quad , \quad \mathsf{CNOT} Z_1 \mathsf{CNOT}^\dagger = Z_1 \\ \mathsf{CNOT} X_2 \mathsf{CNOT}^\dagger &= X_2 \quad , \quad \mathsf{CNOT} Z_2 \mathsf{CNOT}^\dagger = Z_1 Z_2 \ , \end{aligned} \tag{2.57}$$

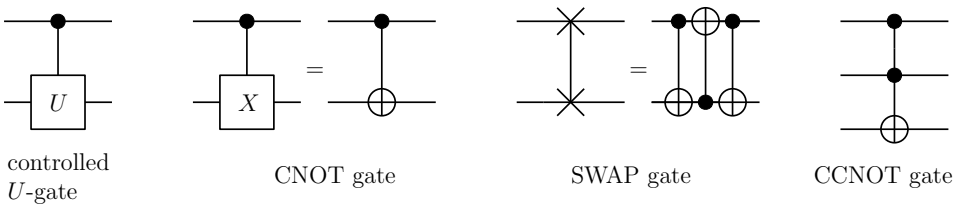
where 1 and 2 label the control and target qubit, respectively. It is also possible to condition on more than one qubit state. The controlled-controlled-X (CCX or CCNOT) gate – also called the *Toffoli gate* – has two control qubits and one target qubit. It inverts the state of the third qubit if and only if both control qubits are in the $|1\rangle$-state. The *SWAP gate* interchanges the states of two qubits, i.e., its matrix representation in the computational basis is

$$\mathsf{SWAP} \coloneqq \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \ . \tag{2.58}$$

Fig. 2.3b shows the circuits implementing several multi-qubit gates.



(a) Single-qubit gates



(b) Multiqubit gates

Figure 2.3: Circuit representation for the most relevant single- and multi-qubit gates.

Universal gate sets

Let us now consider families of gates which are universal according to Definition 2.33.

Theorem 2.34 *(Universal gate set [25])***.** Consider the circuit model of quantum computation on an n-qubit quantum system $\mathcal{H}_S = \left(\mathbb{C}^2\right)^{\otimes n}$. The family of gates consisting of

(i) CNOT gates between pairs of qubits,

(ii) Hadamard gates on every qubit,

(iii) phase gates S from Eq. (2.54) on every qubit,

(iv) $(\pi/8)$-gates on every qubit,

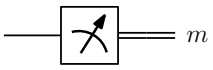
is universal. □

The gates (i), (ii) and (iii) of the above gates (CNOT, Hadamard and phase gates) form the so-called Clifford group. For a proof of this theorem, we refer to standard textbooks [121, Chapter 5.4.3].

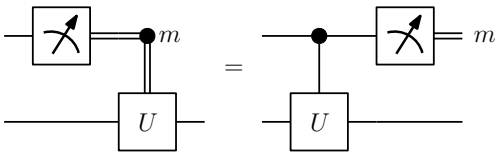
Measurements

To read out the result of the computation, one performs a measurement at the end of the circuit. If not mentioned otherwise, we consider projective measurements of single qubits in the computational basis. The corresponding circuit is depicted in Fig. 2.4. The outcome of such each qubit measurement can take one of two values ± 1, i.e., it contains one bit of classical information.

In the case of quantum error correction, measurements may be also applied in the middle of a circuit, i.e., in between quantum gates. Usually, the post-measurement state is not used in later parts of the circuit since such a measurement generally describes a non-reversible quantum channel, i.e., not a gate. In contrast, the measurement outcome may be used as a control input of a gate applied in the circuit later, the corresponding circuit representation



(a) Projective measurement of a qubit in the computational basis



(b) Gate application conditioned on measurement outcome: the gate U is applied to the second qubit if $m = -1$.

Figure 2.4: Quantum circuit representation of projective measurements in the computational basis. The measurement outcome $m = -1, 1$ associated with the (-1) or $(+1)$-eigenstate of σ_z, respectively, is depicted by a double line (bits carrying classical are depicted by double lines in order to distinguish them from qubits carrying quantum information). (b) This circuit is equivalent to the right hand-side where a controlled-U gate is followed by a projective measurement of the control qubit.

is shown in Fig. 2.4. Note that without loss of generality one can assume to perform measurements only at the end of the circuit: a quantum circuit C with a measurement in the middle of the computation is equivalent to a circuit C' where the measurement is shifted to the circuit's end. This is possible even if the measurement outcome is used as the classical input of a certain gate later in the computation, cf. e.g. [121].

2.3.2 Physical implementations

One may also distinguish different physical systems on which such a model is practically realised. Following a proposal by DiVincenzo [50] and a similar list of requirements from the textbook [121], a physical system has to fulfil the following in order to be used for the implementation of a quantum computer in the circuit model.

(i) Robustly encode quantum information,

(ii) Represent a scalable system of quantum bits,

(iii) Perform a universal family of unitary transformations,

(iv) Prepare an initial state,

(v) Measure the output result.

There exists a plethora of different candidates for such physical realisations of a quantum computer. The pursued proposals – which build on qubits – include for example superconducting circuits, trapped ions, quantum dots of semiconductors, nitrogen-vacancies (so-called NV centres in crystals) and the nuclear spin of molecules (NMR). A different approach is pursued in *continuous variable quantum computation* where the underlying physical system is described by continuous variables (CV), as usually considered in a quantum optical setup. Examples for this approach include a CV analogue of the circuit model for qubits [109] as well as a measurement-based quantum computation model [113], for which CV cluster states have been experimentally demonstrated in 2013 [185]. Both of these approaches model a universal quantum computer. Furthermore, a non-universal computing scheme is boson sampling [1] which is based on a finite number of indistinguishable CV modes. Another direction to use

CV systems for quantum computation relies on embedding or encoding a finite-dimensional system into a CV system, an approach which we discuss in more detail in Section 3.5. We remark that this list of physical realisations is non-exhaustive and that the interested reader is referred to the literature. In order to meet the first requirement (i), all these different implementations have to incorporate some fault-tolerance mechanisms to combat decoherence processes and to reliably operate, the mathematical description of which is discussed in the next chapter.

3 Quantum fault-tolerance

Coherence in quantum systems is fragile and usually decays quickly, a process referred to as decoherence. But for many quantum information processing tasks, coherence is needed, e.g., to reliably store quantum information in a quantum state. The goal of quantum fault-tolerance is to combat these decoherence processes and to protect quantum systems from errors.

One possible strategy would be to avoid errors by physically isolating the system from possible error sources. Unfortunately, this counteracts the ability to interact with the system in order to perform quantum operations, e.g., in quantum computing, so simply perfectly isolating the quantum system from its environment is impractical. In contrast, practically relevant *error avoiding* strategies make use of certain a priori knowledge of the decoherence processes to identify parts of the quantum system which are not affected by decoherence. To protect a quantum state from errors, one stores or encodes it cleverly in this naturally protected part of the system. In contrast, *active* fault tolerance protocols protect the encoded quantum information by actively counteracting the effect of errors while allowing them to (rarely) happen. Here, one may interact with the noisy system, e.g., by pulse application in *dynamical decoupling*. The pulse sequence is chosen using only a priori information on the noise but without involving feedback, i.e., without any extraction of error information during the protocol. Even more evolved fault tolerance protocols are the famous *quantum error correction codes*. They *prevent the encoded quantum information* from decaying by detecting the errors and then undoing their effects. In contrast to classical physics, one has to tackle two main issues here. First, creating redundancy works differently: Simply copying the quantum state several times is fundamentally prohibited due to the no-cloning theorem[1]. One has to think of more clever ways of introducing redundancy in the encoding of information. Second, detecting possible errors is different to classical physics, as well: Since every quantum measurement affects the quantum state, the detection of errors has to be done in a way that leaves the encoded information unchanged.

Organisation of this chapter

This chapter presents the fundamentals of quantum fault-tolerance. We start with an introduction of noise, errors and decoherence models in Section 3.1, including examples of noise channels in the qubit as well as in the CV setting. Section 3.2 briefly recaps the framework of quantum control theory which distinguishes open-loop and closed-loop control protocols, the second allowing for feedback such as measurements in the fault-tolerance protocol whereas the first does not. The remainder of this chapter is organised with respect to this distinction.

[1]The famous no-cloning theorem found by Wootters and Zurek [181] and by Dieks [49] states that there exists no quantum channel such that $|\psi\rangle \mapsto |\psi\rangle \otimes |\psi\rangle$ for all $|\psi\rangle \in \mathcal{H}$.

First, Section 3.3 presents the control techniques of decoherence-free subspaces and dynamical decoupling both for multi-qubit systems. Second, the technique of quantum error correcting codes (for finite-dimensional systems) is discussed in Section 3.4. Finally, Section 3.5 extends the discussion to continuous-variable systems: it summarises the fundamentals of quantum error correcting codes in the bosonic setting. This lays foundation for the novel CV fault-tolerance protocols presented in the remainder of this thesis: the CV dynamical decoupling protocols developed in Chapter 4 and the asymmetric qubit-into-CV encodings analysed in Chapter 5.

3.1 Quantum noise and error models

Let us start with the question of how to mathematically model decoherence, noise and errors on a quantum system. This section presents different models for these noise processes on qubit as well as bosonic systems. Their mathematical description is an important basis for formulating strategies to combat such errors and for analysing and quantifying their performance.

3.1.1 CPTP maps and the Hamiltonian formulation

We will consider two different ways of modelling quantum noise processes, with the first using *CPTP noise channels*s which map an original state to a corrupted state while the second uses *decoherence Hamiltonians* between the system and its environment. In the former case, errors occur in discrete steps of time, where every step corresponds to an application of the noise quantum channel. In the latter case, errors occur over continuous time by the unitary evolution under the decoherence Hamiltonian between the system under consideration and its environment. The term *decoherence* refers to the process of losing coherence, i.e., an originally well-prepared system state may become entangled to its environment such that the superposition properties of the system's state (cf. Section 2.1.2 for the introduction of superposition) are lost.

For the *CPTP map formulation*, recall that the most general transformation of quantum states is given by a quantum channel, cf. Definition 2.14 in Section 2.1.5. It is therefore straightforward to model noise on a quantum system $\mathcal{H}_S$ by a quantum channel

$$\mathcal{N} : \mathcal{D}(\mathcal{H}_S) \to \mathcal{D}(\mathcal{H}_S) \quad , \quad \rho \mapsto \mathcal{N}(\rho) \ , \tag{3.1}$$

which one also calls *noise channel* and is typically denoted by $\mathcal{N}$. In the specific case where $\mathcal{N}(\rho) = E\rho E^\dagger$, one calls E an *error* or *error operator*. More generally, one may refer to the Kraus operators $E_i : \mathcal{H}_S \to \mathcal{H}_S$ (cf. Theorem 2.15) of a noise channel $\mathcal{N}(\rho) = \sum_i E_i \rho E_i^\dagger$ as error operators.

It is sometimes useful to consider the case of *stochastic noise in the CPTP map formulation*. Let $\mathfrak{E}$ denote a set of possible errors on the system – here we assume that $\mathfrak{E} \subset \mathsf{U}(\mathcal{H}_S)$ – and let $\pi : \mathfrak{E} \to [0,1]$ be a probability distribution on $\mathfrak{E}$. The noise channel

$$\mathcal{N} : \mathcal{D}(\mathcal{H}_S) \to \mathcal{D}(\mathcal{H}_S) \qquad , \qquad \rho \mapsto \mathcal{N}(\rho) = \sum_{E \in \mathfrak{E}} \pi(E) E \rho E^\dagger \ ,$$

models stochastic noise, where an error $E \in \mathfrak{E}$ occurs with probability $\pi(E)$.

The Stinespring dilation Theorem 2.18 implies that a noise channel (3.1) arises from a unitary channel on the system and some environment, i.e., for every $\mathcal{N}$ from Eq. (3.1) there are an environment Hilbert space $\mathcal{H}_E$, a state $\sigma \in \mathcal{D}(\mathcal{H}_E)$ and $U \in \mathsf{U}(\mathcal{H}_S \otimes \mathcal{H}_E)$ such that

$$\mathcal{N}(\rho) = \mathrm{tr}_E \left(U(\rho \otimes \sigma)U^\dagger \right) .$$

This has the following interpretation: noise is introduced to the system by the interaction with the environment $\mathcal{H}_E$. This connects the CPTP formulation to the Hamiltonian formulation.

In the *Hamiltonian formulation*, the system's state undergoes noise via interaction with its environment over continuous time $t \in \mathbb{R}$. Let an environment $\mathcal{H}_E$ and a Hamiltonian $H^{\mathrm{orig}} \in \mathcal{L}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$ be given which may include system-environment interactions. The system's state $\rho \in \mathcal{D}(\mathcal{H}_S)$ evolves as

$$\rho(t) = \mathrm{tr}_E \left(U(t)(\rho \otimes \sigma)U(t)^\dagger \right) ,$$

where the respective one-parameter group of unitaries $\{U(t)\}_{t\in\mathbb{R}}$ is generated by H^{orig}. Let us refer to H^{orig} is the *original noise or decoherence Hamiltonian*. By convention, one often splits this noise Hamiltonian into different terms and writes it as the sum of a pure-system part, a pure-environment part and an interaction part:

$$H^{\mathrm{orig}} = H_S \otimes I_E + I_S \otimes H_E + H_{SE} \in \mathcal{L}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E) . \tag{3.2}$$

Here $H_S \otimes I_E$ and $I_S \otimes H_E$ act non-trivially on the system and environment only, respectively, and $H_{SE} \in \mathcal{L}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$ involves system-environment interactions. These interactions entangle system and environment and hence lead to the decoherence of system states (which are initially not entangled with the environment). More generally, the decoherence Hamiltonian (3.2) might be time-dependent, i.e., $t \mapsto H^{\mathrm{orig}}(t)$ where

$$H^{\mathrm{orig}}(t) = H_S(t) \otimes I_E + I_S \otimes H_E(t) + H_{SE}(t) . \tag{3.3}$$

3.1.2 Noise channels for multi-qubit systems

Consider a system of $n \in \mathbb{N}$ qubits with associated Hilbert space $\mathcal{H}_S \coloneqq (\mathbb{C}^2)^{\otimes n}$. Recall from Section 2.1.3 that the Pauli-matrices $\{I, \sigma_x, \sigma_y, \sigma_z\}$ form a basis of the real vector space of complex Hermitian 2×2 matrices as well as the complex vector space of complex 2×2 matrices. To describe n-qubit errors, it is useful to consider a group of $2^n \times 2^n$ matrices which can be seen as a generalisation of the group generated by the single-qubit Pauli matrices.

Definition 3.1 *(Pauli group).* The n-qubit *Pauli group* $\mathcal{P}_n$ is defined as the group of operators which are tensor products of single-qubit Pauli matrices up to an overall phase ±1 or $\pm i$, i.e., elements $E \in \mathcal{P}_n$ are given by

$$E = \alpha E_1 \otimes E_2 \otimes \cdots \otimes E_n ,$$

where $\alpha \in \{\pm1, \pm i\}$ and $E_k \in \{I, \sigma_x, \sigma_y, \sigma_z\}$ for $k = 1, 2, \ldots, n$.

We will usually neglect the complex phase α of operators $E \in \mathcal{P}_n$ with rare exceptions (e.g. in the stabiliser group where the sign makes a difference). A prominent way to model multi-qubit noise is stochastic Pauli noise.

Definition 3.2 *(Stochastic Pauli noise).* *Stochastic Pauli noise* on $n \in \mathbb{N}$ qubits is described by a probability distribution $\pi : \mathcal{P}_n \to [0,1]$. The n-qubit Pauli noise channel is given by

$$\mathcal{N}_\pi : \mathcal{D}\left((\mathbb{C}^2)^{\otimes n}\right) \to \mathcal{D}\left((\mathbb{C}^2)^{\otimes n}\right) \quad , \quad \rho \mapsto \sum_{E \in \{I,\sigma_x,\sigma_y,\sigma_z\}^{\otimes n}} \pi(E) E \rho E^\dagger \ . \tag{3.4}$$

Note that we use $E \in \{I, \sigma_x, \sigma_y, \sigma_z\}^{\otimes n}$ instead of $E \in \mathcal{P}_n$ in Eq. (3.4) since we neglect complex phases of E. Here, a single Pauli error $E \in \mathcal{P}_n$ occurs with probability $\pi(E)$. For $n = 1$, Definition 3.2 defines a *single-qubit stochastic Pauli noise channel*: there is a probability distribution $\pi \coloneqq (p_I, p_X, p_Y, p_Z) : \mathcal{P}_1 \to [0,1]$ such that the noise channel $\mathcal{N}_\pi$ is of the form

$$\begin{aligned} \mathcal{N}_{(p_I,p_X,p_Y,p_Z)} : \mathcal{D}(\mathbb{C}^2) &\to \mathcal{D}(\mathbb{C}^2) \ , \\ \rho &\mapsto p_I \rho + p_X \sigma_x \rho \sigma_x^\dagger + p_Y \sigma_y \rho \sigma_y^\dagger + p_Z \sigma_z \rho \sigma_z^\dagger \ . \end{aligned} \tag{3.5}$$

A special case of n-qubit stochastic Pauli noise is the *independent channel*

$$\mathcal{N}_\pi = \bigotimes_{i=1}^{n} \mathcal{N}_{\pi_i} \ ,$$

where for every qubit $i = 1, 2, \ldots, n$, the probability distribution $\pi_i : \mathcal{P}_1 \to [0,1]$ is defined on the single-qubit Pauli group $\mathcal{P}_1$, i.e., every $\mathcal{N}_{\pi_i} : \mathcal{D}(\mathbb{C}^2) \to \mathcal{D}(\mathbb{C}^2)$ is a single-qubit stochastic Pauli noise channel. Single-qubit Pauli errors occur independently on the qubits.

It is sometimes reasonable to assume that all π_i are the same. In these cases, we speak of *independent and identically distributed* (i.i.d.) Pauli noise and the n-qubit noise channel is of the form

$$\mathcal{N}_\pi = \mathcal{N}_{(p_I,p_X,p_Y,p_Z)}^{\otimes n} \tag{3.6}$$

where $\mathcal{N}_{(p_I,p_X,p_Y,p_Z)}$ is a single-qubit Pauli channel from Eq. (3.5). Here, the probability of a single-qubit Pauli error is the same on every qubit.

Examples of single-qubit noise channels

Recall from Section 2.1.3 the single qubit and the Pauli operators σ_x, σ_y and σ_z from Eq. (3.5). In the context of Pauli noise (3.5), one calls σ_x and σ_z bit flip and phase flip errors, respectively, since they act on the basis vectors $|0\rangle$ and $|1\rangle$ as

$$\sigma_x : \begin{array}{ccc} |0\rangle & \mapsto & |1\rangle \\ |1\rangle & \mapsto & |0\rangle \end{array} \quad , \quad \sigma_z : \begin{array}{ccc} |0\rangle & \mapsto & |0\rangle \\ |1\rangle & \mapsto & -|1\rangle \end{array} \ .$$

Due to the relation $\sigma_y = i\sigma_x\sigma_z$, one analogously calls σ_y bit-and-phase-flip. These three Pauli errors are also labelled X, Y and Z. Let us give a few relevant examples of stochastic single-qubit noise channels from Eq. (3.5).

The *dephasing noise channel* corresponds to pure σ_z-noise, i.e., to the Pauli noise from Eq. (3.5) for $p = p_Z \in [0,1]$ and $p_I = 1 - p$, $p_X = p_Y = 0$ and $p_Z = p$ such that

$$\mathcal{N}_{\text{dephasing}}(\rho) = (1-p)\rho + p\sigma_z \rho \sigma_z^\dagger \ .$$

A phase flip occurs with probability p and no error happens with probability $1 - p$. Under dephasing noise, the Bloch sphere (cf. Fig. 2.1) shrinks asymmetrically, where it stays unchanged along the σ_z axes but it is shrunk along the other two axis. Similarly, *bit flip* noise

corresponds a Pauli channel (3.5) of pure σ_x-noise, i.e.,

$$\mathcal{N}_{\text{flip}}(\rho) = (1-p)\rho + p\sigma_x\rho\sigma_x^\dagger \ . \tag{3.7}$$

The *depolarising noise channel*

$$\mathcal{N}_{\text{depolarizing}}(\rho) = (1-p)\rho + \frac{p}{3}\sigma_x\rho\sigma_x^\dagger + \frac{p}{3}\sigma_y\rho\sigma_y^\dagger + \frac{p}{3}\sigma_z\rho\sigma_z^\dagger \ , \quad \text{for } p \in [0,1] \tag{3.8}$$

is a Pauli channel for $p_I = 1-p$, and $p_X = p_Y = p_Z = p/3$. Each non-trivial Pauli error X, Y and Z occurs with equal probability $p/3$.

The noise channel for *independent X and Z-noise* is given by a single qubit Pauli channel (3.5) where the probabilities are given by

$$\begin{array}{rcccl} p_I &=& (1-q_X) &\cdot& (1-q_Z) \\ p_X &=& q_X &\cdot& (1-q_Z) \\ p_Y &=& q_X &\cdot& q_Z \\ p_Z &=& (1-q_X) &\cdot& q_Z \end{array} \tag{3.9}$$

for $q_X, q_Z \in [0,1]$. One assumes that bit flips and phase flips happen independently and σ_Y-errors if both bit and phase flips occur. One may define independent X and Y or independent Y and Z-noise analogously.

3.1.3 Noise channels and error models for bosonic systems

Let us consider noise on a quantum system of $n \in \mathbb{N}$ bosonic modes with associated Hilbert space $\mathcal{H} \coloneqq L^2(\mathbb{R}^n)$. Since general (non-Gaussian) bosonic channels are mathematically difficult to study, we list several, practically relevant examples which are Gaussian (cf. Section 2.2.3).

Classical noise channel

Recall that the Pauli group can be used as a basis of error operators for qubit systems. In the context of bosonic systems, the *Heisenberg-Weyl group*

$$H_n \coloneqq \left\{ e^{i\alpha} D(\xi) \mid \alpha \in [0, 2\pi),\ \xi \in \mathbb{R}^{2n} \right\} \tag{3.10}$$

plays a similar role. When analysing GKP codes (cf. Section 5.1.1), we are especially interested in noise channels which can be expressed in terms of these group elements:

Definition 3.3 *(Displacement noise).* The *displacement noise channel* or *classical noise channel* is given by a CPTP map

$$\mathcal{N}_f : \mathcal{D}\left(L^2(\mathbb{R}^n)\right) \to \mathcal{D}\left(L^2(\mathbb{R}^n)\right) \quad , \quad \rho \mapsto \int_{\mathbb{R}^{2n}} f(\xi) D(\xi) \rho D(\xi)^\dagger d^{2n}\xi \ , \tag{3.11}$$

where $f : \mathbb{R}^{2n} \to \mathbb{R}$ is a probability density function on the phase space $\mathbb{R}^{2n}$.

This channel is the bosonic analogy of the (stochastic) Pauli noise channel (3.4). The operators in which one expands this channel (3.11), i.e., the displacement operators $D(\xi)$, are called *(displacement) errors*. We speak of independent displacement noise on the n bosonic

modes if there are single-mode probability distributions $g_k : \mathbb{R}^2 \to \mathbb{R}$ for $k = 1, \ldots, n$ such that

$$f(\xi) = \prod_{k=1}^{n} g_k(\xi_k, \xi_{n+k}) \quad \text{for all } \xi = (\xi_1, \xi_2, \ldots, \xi_{2n})^T \in \mathbb{R}^{2n} \ .$$

The bosonic analogue of i.i.d. Pauli noise is the *i.i.d. displacement noise* where all g_k are identical, i.e.,

$$\mathcal{N}_f = \mathcal{N}_g^{\otimes n} \ , \qquad \text{where} \qquad \mathcal{N}_g(\rho) = \int_{\mathbb{R}^2} g(\xi) D(\xi) \rho D(\xi)^\dagger d^2\xi \ ,$$

for a probability distribution $g : \mathbb{R}^2 \to \mathbb{R}$ on the single-mode phase space.

Note that the classical noise channel from Definition 3.3 is Gaussian if and only if f is a Gaussian function. Recall the characterisation of Gaussian channels in terms of the triple (X, Y, v) from Section 2.2.3 (cf. Eqs. (2.28) and (2.29)). In the case of f being a Gaussian function of variance σ^2 in all $2n$ phase space dimensions, the classical noise channel corresponds to $X = I_{2n}$, $Y = \sigma^2 I_{2n}$, and $d = 0$.

Thermal noise and pure loss channels

Moreover, one may consider dilations of Gaussian channels, especially those which can be written as *dilations of Gaussian unitary channels* with a Gaussian environment. Such channels can be written as

$$\mathcal{N} : \mathcal{D}\left(L^2(\mathbb{R}^n)\right) \to \mathcal{D}\left(L^2(\mathbb{R}^n)\right) \quad , \quad \rho \mapsto \mathcal{N}(\rho) = \mathrm{tr}_E \left[U(\rho \otimes \sigma) U^\dagger \right] \ ,$$

for a Gaussian unitary $U \in \mathsf{U}(L^2(\mathbb{R}^{n+n_E}))$ on n system and n_E environment modes and a Gaussian environment state $\sigma \in \mathcal{D}(L^2(\mathbb{R}^{n_E}))$.

The *thermal noise channel* or *thermal attenuator* is given by

$$\mathcal{N}_{\eta,\nu}(\rho) = \mathrm{tr}_E \left[U_\lambda (\rho \otimes \rho_{\mathsf{th},\nu}) U_\lambda^\dagger \right]$$

where the environment state $\rho_{\mathsf{th},\nu} \in \mathcal{D}(L^2(\mathbb{R}^n))$ is a thermal state (cf. Eq. (2.26) from Section 2.2.3) of mean photon number ν and the unitary $U_\lambda \in \mathsf{U}(L^2(\mathbb{R}^{n+n}))$ is that of a beam splitter (cf. Eq. (2.47) in Section 2.2.4) between pairs of modes of the system and the environment $\mathcal{H}_E = L^2(\mathbb{R}^n)$ such that

$$U_\lambda \coloneqq U_{\mathrm{bs}}(\arccos(\lambda))^{\otimes n} \coloneqq e^{\arccos(\lambda) \sum_{j=1}^{n} (a_{j+n}^\dagger a_j - a_j^\dagger a_{j+n})} \ .$$

Then, the two matrices (X, Y) characterising the Gaussian channel $\mathcal{N}_{\eta,\nu}$ are $X = \sqrt{\eta} I$ and $Y = (1 - \eta)(2\overline{n} + 1) I$. If $\rho_{\mathsf{th},0}$ is the vacuum state, we call $\mathcal{N}_{\eta,0}$ the *pure loss channel* (other names for this channel are *amplitude damping channel* or attenuation channel). For a single bosonic mode $n = 1$, this channel can be written as [37]

$$\mathcal{N}_\chi : \mathcal{D}(L^2(\mathbb{R})) \to \mathcal{D}(L^2(\mathbb{R})) \quad , \quad \text{where} \quad \mathcal{N}_\chi \coloneqq e^{\chi \mathcal{D}}$$

in terms of the super operator $\mathcal{D}(\rho) = a \rho a^\dagger - \frac{1}{2} \left\{ a^\dagger a, \rho \right\}$ (also called Liouvillian) and the dimensionless damping parameter χ, or sometimes more conveniently in terms of the loss rate $\gamma \coloneqq 1 - e^{-\chi}$ that quantifies the amount of loss.

3.2 Quantum Control Theory

In the previous section we introduced how to describe noise and errors. Naturally, we now want to model fault-tolerance protocols that combat these error processes, e.g, by detection and correction. One of such frameworks is *quantum control theory*. Here, we briefly introduce how quantum error correcting codes and pulse sequences may fall into this framework. For further information, the reader is referred to reviews on surveys on quantum control theory such as [51].

One usually considers a quantum system with Hilbert space $\mathcal{H}_S$ and noise introduced to this system by a so-called drift or original Hamiltonian $H^{\text{orig}}(t)$. This Hamiltonian can be either acting on the system only, i.e., $H^{\text{orig}}(t) \in \mathcal{L}_{sa}(\mathcal{H}_S)$ if one considers a closed system or it is of the form of Eq. (3.3), i.e., acting on the system and its environment $\mathcal{H}_E$ if one considers an open system. Control on the system is exerted by a so-called control scheme.

Definition 3.4 *(Control scheme).* A *control scheme* on a quantum system $\mathcal{H}_S$ is given by a time $T > 0$ and a control Hamiltonian

$$H^{\text{control}}(t) = \sum_{j=1}^{L} u_j(t) H_j \qquad \text{for } t \in [0, T] \tag{3.12}$$

where $H_1, \ldots, H_L \in \mathcal{L}_{sa}(\mathcal{H}_S)$ are a collection of Hamiltonians and $u_1, \ldots, u_L : \mathbb{R} \to \mathbb{C}$ are control functions such that the propagator $U^{\text{control}}(T)$ from time 0 to T generated by $H^{\text{control}}(t)$ satisfies $U^{\text{control}}(T) = I_S$.

The effective dynamics are generated by the resulting Hamiltonian $H^{\text{res}}(t) = H^{\text{orig}}(t) + H^{\text{control}}(t)$ which is is the sum of the drift/original or decoherence Hamiltonian $H^{\text{orig}}(t)$ and the control Hamiltonian (3.12). The latter is split into several control Hamiltonians $H_1, H_2, \ldots, H_L$ which can be time-dependently modified – e.g. turned on or off – via the control amplitudes or control functions $u_1, u_2, \ldots, u_L$. The requirement $U^{\text{control}}(T) = I_S$ in Definition 3.4 ensures that after time $T > 0$, no additional decoherence is introduced to the system via the controls.

Bilinear control system and controllability of a closed system

Let us consider a closed quantum system, i.e., where both the original and the control Hamiltonian act on the system only. A *bilinear quantum control system* is given by the Schrödinger equation for the resulting Hamiltonian $H^{\text{res}}(t) = H^{\text{orig}}(t) + H^{\text{control}}(t)$, where H^{control} defines the control Hamiltonian of a control system from Definition 3.4, i.e., by

$$\begin{aligned} \frac{d}{dt}|\psi(t)\rangle &= -i\Big(H^{\text{orig}} + \sum_{j=1}^{L} u_j(t) H_j\Big)|\psi(t)\rangle \ , \\ \frac{d}{dt}U^{\text{res}}(t) &= -i\Big(H^{\text{orig}} + \sum_{j=1}^{L} u_j(t) H_j\Big)U^{\text{res}}(t) \ , \quad U^{\text{res}}(0) = I_S \ , \end{aligned} \tag{3.13}$$

for pure quantum states and unitaries, respectively. A typical task is to find a time $T > 0$ and admissible controls $\{u_j\}_{j=1}^{L}$ which drive the system (3.13) from a given initial state $|\psi_0\rangle$ into a predefined final state $|\psi_{\text{fin}}\rangle = |\psi(T)\rangle$. Using this strategy, one may be able to avoid

errors that are introduced over time by $H^{\text{orig}}(t)$, i.e., for the task of state-preservation where the final state $|\psi(T)\rangle = |\psi_0\rangle$ is equal to the initial state. Slightly more generally, one might be interested in engineering a predefined evolution: for a given unitary U, a drift H^{orig} and a set of potential control terms $\{H_j\}_{j=1}^{L}$, one searches for a finite time $T > 0$ and a set of admissible control functions $\{u_j\}_{j=1}^{L}$ such that $U^{\text{res}}(T) = U$.

Let us consider the standard theoretic notion of *controllability*. A bilinear system (3.13) is *pure-state controllable* if for every pair of states $|\psi_0\rangle$ and $|\psi_{\mathsf{fin}}\rangle$ there exist control functions that drive the system from $|\psi_0\rangle$ to $|\psi_{\mathsf{fin}}\rangle$. Similarly, the system is *operator controllable* if for every unitary U there exist control functions such that $U = U^{\text{res}}(T)$. Although most controllability proofs are not constructive, this notion is of practical relevance. For example, operator controllability is connected to universality in the context of quantum computing (cf. Section 2.3) where one wants to model every unitary on the system using a restricted set of controls. More precisely, one may be interested which unitaries $U \in \mathsf{U}(\mathcal{H}_S)$ can be engineered by a given control scheme 3.4.

For finite-dimensional systems, there exist well-established criteria on the controllability of a quantum system in terms of the Lie algebras generated by the control Hamiltonians [5]. This is more intricate in the infinite-dimensional case. Huang et al. [85] laid foundations on their controllability and we refer the interested reader to the review article by Dong and Peterson [51].

Control scheme in open systems

Quantum control theory can be extended to open quantum system. There are different approaches to do this, e.g. using so-called Markovian master equations (the 'open' analogy of the Schrödinger equation) or stochastic master equations (cf. e.g. [107, Chapter 1] for these noise models and [51] for control theory in this context). In this thesis, we focus on another direction: We assume that noise is introduced to the system by a decoherence Hamiltonian $H^{\text{orig}}(t)$ from Eq. (3.3) (as described in Section 3.1.1) which acts on the system and its environment $\mathcal{H}_E$. Note that in a control scheme, the control Hamiltonian still acts non-trivially on the system only, reflecting the practical restriction that one may only influence the system part. As a consequence, $H^{\text{control}}(t) = \sum_j u_j(t) H_j \otimes I_E$ in Definition 3.4.

The total or *resulting Hamiltonian* on the system and its environment is given by

$$H^{\text{res}}(t) = H^{\text{orig}}(t) + H^{\text{control}}(t) \qquad \text{for } t \in [0,T] \,. \tag{3.14}$$

One cannot influence its original part $H^{\text{orig}}(t) : \mathbb{R} \to \mathcal{L}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$ – which may include noise or decoherence (sometimes called drift) – but only the part $H^{\text{control}}(t)$ from Eq. (3.12). After time T, the resulting evolution $U^{\text{res}}(T)$ of system and environment is defined as the propagator generated by the resulting Hamiltonian (3.14) from time t to T, i.e., it is given by the relation

$$U^{\text{res}}(T) \coloneqq \mathcal{T}_+ \left[e^{-i \int_0^T (H^{\text{orig}} + H^{\text{control}}(t)) dt} \right] \,. \tag{3.15}$$

We will focus on open systems for the remainder of this section.

Bang-bang control and pulse sequences

Sometimes, one imposes certain restrictions on the control functions for practical or theoretical reasons, e.g. one restricts to stepwise constant or continuous functions. In $H^{\text{control}}(t)$ from

Eq. (3.12), the pulse shapes are characterised by the form of the functions $\{t \mapsto u_j(t)\}_{j=1}^{L}$. One example is to consider finite (constant) pulse width and strength which corresponds to piecewise constant control functions $u_j(t)$.

A so-called bang-bang control is exerted by the application of instantaneous and infinitely strong control pulses. More precisely, one assumes that the control Hamiltonian is of the form

$$H^{\text{control}} = \sum_{j=1}^{L} \delta(t - t_j) H_j \otimes I_E \ , \tag{3.16}$$

where $t_1, \ldots, t_L$ are called the control times and $U_1 := e^{iH_1}, \ldots, U_L := e^{iH_L}$ are called the control pulses.

Definition 3.5 *(Pulse sequence).* A *(bang-bang control) scheme* or simply a *pulse sequence* is given by of a pair of finite families $\{t_j\}_{j=1}^{L} \subset [0, T]$ and $\{U_j\}_{j=1}^{L} \subset \mathsf{U}(\mathcal{H}_S)$ for $L \in \mathbb{N}$ such that $\prod_{j=1}^{L} U_j = I_S$.

This is consistent with Definition 3.4 of a control system since a control Hamiltonian (3.16) can be fully characterised by a pulse sequence. If not stated otherwise, we consider control Hamiltonians in the sense of Definition 3.5 throughout this thesis. We note that this is an idealised setup, but mathematically, it can be treated reasonably well and realistic pulse shapes with finite strength and width can be seen as an approximation of these idealised delta peaks. We remark that there exist many other pulse shapes which have been studied in the literature [89].

Open-loop and closed-loop control

One distinguishes *open-loop* and *closed-loop* control problems: In the former case, one has to choose the control functions and Hamiltonians a-priori, i.e., without any feedback obtained during the evolution. Dynamical decoupling (DD) – cf. Section 3.3.2 – is such a control technique where a pulse sequence of the form of Definition 3.5 is applied to the system in order to approximately average out the effect of decoherence H^{orig}. In the latter case, in closed-loop control, the control functions might be adapted during the control process and they depend on feedback obtained from measurements. Typically, such measurements (which are called syndrome measurements) do not alter the system's state, but their outcomes (which are called syndromes) reveal information about potential errors and are performed using an additional system which is destructively measured. In this case, the control functions at time t depend on the outcomes of syndrome measurements performed before time t. In this sense quantum error correcting codes (QECC) – cf. Section 3.4 – can be viewed as a closed-loop control technique.

In principle the two techniques of DD and QECC can combined in order to use the advantages of both approaches: In a QECC, an additional DD protocol reduces the necessary overhead and resource-costs of the employed QECC. Using DD protocols to first reduce the effective error rate can allow QECC protocols with thresholds to be applied. For multi-qubit systems, there exist studies for the combination, e.g., designing DD-protected gates [120] and identifying optimal DD pulses for subsystem codes [129].

3.3 Decoherence-free subspaces and dynamical decoupling of qubit systems

Let us now consider open-loop quantum control protocols, starting with the passive error protection using decoherence-free subspaces in Section 3.3.1 and then with the active protocol of dynamical decoupling in Section 3.3.2. Throughout this chapter we only consider finite-dimensional quantum systems.

3.3.1 Decoherence-free subspaces and series expansions

Decoherence-free subspaces

In contrast to the active error correcting strategies of QECC, the concept of decoherence free subspaces offers a passive error preservation scheme. The term *passive* stands for the fact that no interaction with the system, i.e., no measurement, no feedback and no control pulse, is involved in these protocols. The main strategy is to cleverly encode information into a subspace of the physical system space which is not affected by decoherence.

Definition 3.6 *(Decoherence-free subspaces).* Consider a quantum system Hilbert space $\mathcal{H}_S$ and a linear subspace $\mathcal{H}_D \subset \mathcal{H}_S$ where P_D denotes the projection onto $\mathcal{H}_D$. Let $\Phi : \mathcal{D}(\mathcal{H}_S) \to \mathcal{D}(\mathcal{H}_S)$ be a quantum channel and let $H^{\text{orig}} \in \mathcal{B}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$ be a Hamiltonian between the system and an environment $\mathcal{H}_E$.

(i) The subspace $\mathcal{H}_D \subset \mathcal{H}_S$ is a *decoherence-free subspace* (DFS) in the quantum channel formulation if for any $\rho \in \mathcal{D}(\mathcal{H}_S)$ which satisfies $\rho = P_D \rho P_D$, one has $\Phi(\rho) = U\rho U^\dagger$ for some unitary $U \in \mathsf{U}(\mathcal{H}_S)$.

(ii) The subspace $\mathcal{H}_D \subset \mathcal{H}_S$ is a *decoherence-free subspace* (DFS) in the Hamiltonian formulation if for any $\rho \in \mathcal{D}(\mathcal{H}_S)$ which satisfies $\rho = P_D \rho P_D$ one has

$$\mathrm{tr}_E \left(e^{-itH^{\text{orig}}} (\rho \otimes \sigma) e^{itH^{\text{orig}}} \right) = U\rho U^\dagger$$

for some unitary $U \in \mathsf{U}(\mathcal{H}_S)$ and $\sigma \in \mathcal{D}(\mathcal{H}_E)$.

If $\mathcal{H}_D$ is a DFS then the initial and the final state are unitarily related. There is no leakage of information out of $\mathcal{H}_D$ by the decoherence map, i.e., any initially 'decoupled state' which lies in $\mathcal{H}_D$ behaves under the decoherence evolution as if it was part of a closed quantum system under a reversible (unitary) evolution. Note that Definition 3.6 captures both cases, when decoherence is introduced by noise channels and when by the dilation of a unitary time-dependent evolution in the Hamiltonian formulation. The main task in using decoherence-free subspaces for error reduction is to *identify* suitable subspaces for a given, specific decoherence channel Φ or Hamiltonian H^{orig}. The advantage of DFS is that this strategy is highly resource efficient. In contrast to QECC, no syndrome measurement and no correction operation is necessary.

Decoherence-free subspaces can be interpreted as completely degenerate quantum error correcting codes, i.e, if all errors act as the identity on the codespace, then no active error correction is necessary. The existence of states which do not decohere was first discovered

for specific models such as the spin-boson model. Lidar, Chuang and Whaley introduced the terminology of decoherence-free subspaces in 1998 [108], where they derived the existence of decoherence free states in the context of Markovian master equations, a result independently found in the same year by Zanardi [186]. We refer to the literature for a detailed analysis of DFS, e.g. [107, Chapter 3] by Lidar and Brun.

Dyson expansion and Magnus expansion

Let us now consider the case when the decoherence Hamiltonian is time dependent, i.e., $t \mapsto H(t) \in \mathcal{L}_{sa}(\mathcal{H})$ from Eq. (3.2). As discussed in Section 2.1.5, the unitary propagator from time t_0 to t is given by $U(t, t_0)$ from (2.15). Let us investigate two different approaches to expand $U(t) \coloneqq U(t, t_0 = 0)$ in a series, the *Dyson series* [54] and the *Magnus series* [111].

Definition 3.7 *(Dyson and Magnus series).* Let $n \in \mathbb{N}$. For $t > 0$, let $H(t) \in \mathbb{C}^{n\times n}$ be a time-dependent Hamiltonian and consider the initial value problem (2.15) defined by $H(t)$.

(i) The *Dyson expansion* or *Dyson series* of $U(t)$ is defined as

$$\begin{aligned} U^{\mathsf{Dy}}(t) \coloneqq& I + \sum_{k=1}^{\infty} A_k(t) \ , \qquad \text{where} \\ A_k(t) \coloneqq& (-i)^k \int_0^t \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} H(\tau_1)H(\tau_2)\cdots H(\tau_k) d\tau_k \cdots d\tau_2 d\tau_1. \end{aligned} \tag{3.17}$$

(ii) The *Magnus expansion* of the unitary $U(t)$ is given by

$$U^{\mathsf{Mag}}(t) \coloneqq e^{\Omega(t)} \ , \qquad \text{where} \qquad \frac{d}{dt}\Omega = -i \frac{\mathsf{ad}_\Omega}{\exp(\mathsf{ad}_\Omega) - 1} H \ . \tag{3.18}$$

The assumption $H(t) \in \mathbb{C}^{n\times n}$ implies the boundedness of $H(t)$ for every t. Let us briefly show that this is sufficient to guarantee that the Dyson series (3.17) is well-defined: If $H(t)$ is bounded in a unitarily invariant norm[2] $\|\cdot\|$, i.e., if

$$\lambda \coloneqq \max_{s\in[0,t]} \|H(s)\| < \infty \tag{3.19}$$

then every term in the Dyson series is bounded by

$$\|A_k(t)\| \leq \int_0^t \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} \|H(\tau_1)\|\|H(\tau_2)\| \cdots \|H(\tau_k)\| d\tau_k \cdots d\tau_2 d\tau_1 \leq \frac{(\lambda t)^k}{k!} \ .$$

As a consequence, the infinite sum over these terms can be bounded by the Taylor series of an exponential function $\|U^{\mathsf{Dy}}(t)\| \leq \|I\| + e^{\lambda t}$ and the Dyson series converges absolutely. Note that one recovers the expression $U(t) = e^{-itH}$ from the Dyson series if H is not time-dependent. A disadvantage of the Dyson expansion is that the partial sums $I + \sum_{k=1}^{l} A_k(t)$ for finite $l < \infty$ are not necessarily unitary.

For the Magnus series, the Hamiltonian

$$\Omega(t) \coloneqq \sum_{k=0}^{\infty} \Omega_k(t)$$

[2] Unitary invariance of a norm $\|\cdot\|$ is defined as $\|VA\| = \|A\|$ for all $V \in \mathsf{U}(\mathcal{H})$ and $A \in \mathcal{B}(\mathcal{H})$.

in the exponent is expanded as a series. The terms $\Omega_k(t)$ are recursively defined by Eq. (3.18). The first three terms, cf. e.g. [22], are

$$\Omega_0(t) = -i \int_0^t H(\tau_1) d\tau_1 \ ,$$
$$\Omega_1(t) = -\frac{1}{2} \int_0^t \int_0^{\tau_1} [H(\tau_1), H(\tau_2)] d\tau_2 d\tau_1 \ ,$$
$$\Omega_2(t) = \frac{i}{6} \int_0^t \int_0^{\tau_1} \int_0^{\tau_2} \left([H(\tau_1), [H(\tau_2), H(\tau_3)] + [H(\tau_3), [H(\tau_2), H(\tau_1)] \right) d\tau_3 d\tau_2 d\tau_1 .$$

In general, every $\Omega_k(t)$ is the integral over $k-1$ nested commutators of $H(t)$ at different times, i.e., it depends on $H(t)$ in nth order.

We note again that the Magnus expansion does not converge in general. This may not even be the case if the Hamiltonian is bounded. But if

$$\int_0^t \|H(\tau)\| d\tau < \pi$$

in a specific norm $\|\cdot\|$, then $\|\Omega(t)\| < \infty$ and the Magnus expansion converges. In contrast to the Dyson expansion, the considered time plays a role, too, i.e., the larger the norm of the Hamiltonian, the smaller we must choose the time t such that the Magnus expansion converges. An advantage of the Magnus series is that the partial sums $\sum_{k=0}^{l} \Omega_k(t)$ are all self-adjoint for every $l \in \mathbb{N}$, i.e., all orders of the Magnus expansion are unitary.

The Dyson and the Magnus series are useful tools for dynamical decoupling schemes discussed in the next section.

3.3.2 Dynamical decoupling

Dynamical decoupling (DD) is a control technique where unitary control pulses are applied to the system. Their choice is fixed a priori and the protocol does not involve any kind of feedback such as measurements. In this sense it might be considered as an example of an open-loop control technique and it is similar to DFS, cf. Section 3.3.1. But in contrast to DFS they do involve *active* control operations. Such protocols aim to reduce the effective error rates at the physical level.

Dynamical decoupling is among the most successful strategies for error reduction. It was originally developed in the context of nuclear magnetic resonance (NMR) as coherent averaging of interactions. In pioneering work, Haeberlen and Waugh [175, 73] designed pulse sequences that enhance the resolution of NMR spectroscopy by rapid periodic application of pulses which render the NMR spins to behave under the evolution of a time-dependent average Hamiltonian. They were inspired by previous paradigmatic protocols: the Hahn spin echo [74], a technique to refocus a decaying spin in a so-called spin echo and the Carr-Purcell-Meiboom-Gill (CPMG) sequences [34, 112], a popular method to measure the NMR spin-spin relaxation time T_2 using spin echoes.

Viola, Lloyd and Knill [167, 166, 165] generalised the above proposals and laid the theoretical foundation for the framework of dynamical decoupling. Practically, DD protocols are well-established and have been successfully applied to a plethora of different physical systems; besides NMR for which there exists an extensive amount of applications and relevant experiments (see [41] for a survey on NMR quantum information processing), this includes

electron spins in solids [52, 173], defect centres [45, 182], quantum dots [17], ion traps [20, 21], superconducting qubits [31, 132] and photon polarisation qubits [42]. Besides the obvious application of suppressing decoherence, there is a second interesting one: When the decoherence processes are complex and partly unknown, very simple DD schemes can be used as a diagnostic tool to analyse the decoherence Hamiltonian, cf. e.g. [155, 31].

In both applications, the basic idea of DD is to interleave the free and uncontrolled system-environment evolution responsible for decoherence by rapidly applied and suitably chosen control pulses. When one aims at reducing decoherence, the goal is to approximately average out its effect on the system such that after the application of the pulse sequence, the system and the environment are decoupled and the resulting evolution acts trivially on the system.

Definition of Dynamical Decoupling (DD)

Since achieving dynamical decoupling can be viewed as an open-loop control problem over a finite (and sufficiently small) time scale T, recall the setup of quantum control theory from Section 3.2. More precisely, consider a control scheme as discussed in Definition 3.4 where $H^{\text{control}}(t)$ determines a control Hamiltonian acting non-trivially on the system $\mathcal{H}_S$ only. Let furthermore $\mathcal{H}_E$ be the environment and let decoherence be given by an original Hamiltonian $H^{\text{orig}} \in \mathcal{L}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$ of the form of Eq. (3.2) that acts on the system and the environment. As a consequence, the resulting Hamiltonian acting on system and environment is given by the sum of the uncontrollable original Hamiltonian and the control Hamiltonian, i.e.,

$$H^{\text{res}}(t) = H^{\text{orig}} + H^{\text{control}}(t) \ .$$

This Hamiltonian generates the resulting unitary $U^{\text{res}}(t)$.

Let us fix a (typically small) time $T > 0$ which is sometimes referred to as *decoupling time* or *cycle time*.

Definition 3.8 *(Dynamical decoupling).* A control scheme in the form of Definition 3.4 is said to achieve *universal dynamical decoupling* after time $T > 0$ if and only if for all $H^{\text{orig}} \in \mathcal{B}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$, the effective evolution $U^{\text{res}}(T)$ from Eq. (3.15)

(i) involves no interaction between system and environment i.e., for all $H^{\text{orig}} \in \mathcal{B}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$ there exist $U_S \in \mathsf{U}(\mathcal{H}_S)$ and $U_E \in \mathsf{U}(\mathcal{H}_E)$ such that $U^{\text{res}}(T) = U_S \otimes U_E$, and

(ii) acts trivially on the system, i.e., $U_S = I_S$.

The distinction between the two conditions (i) and (ii) will become clearer in Chapter 4.6 since in the bosonic case, the second one (ii) is no longer achievable. We note that the first condition (i) of Definition 3.8 resembles DFSs (cf. Definition 3.6): the system Hilbert space is actively turned into a DFS which evolves (after time T) independent of the environment. Hence, we refer to pulse sequences which (only) achieve property (ii) as *decoherence suppression* schemes. Sometimes, it might be even desirable to not make the evolution $U^{\text{res}}(T)$ act trivially on the system, but to actively engineer a specific non-trivial system Hamiltonian $\overline{H}$ such that $U^{\text{res}}(T) = e^{-iT\overline{H}}$.

One may also formulate dynamical decoupling in terms of the evolved system state

$$\rho_S^{\text{res}}(T) = \text{tr}_E \left[U^{\text{res}}(t) \left(\rho_S(0) \otimes \rho_E(0) \right) U^{\text{res}}(t)^\dagger \right] \ .$$

If a control scheme achieves DD, then – assuming that initially, system and environment are decoupled such that $\rho_{SE}(0) = \rho_S(0) \otimes \rho_E(0)$ – the resulting system state should be the same as the initial state, i.e., $\rho_S(T) = \rho_S(0)$.

Achieving dynamical decoupling can be viewed as an open-loop control problem over the finite (and sufficiently small) control time scale T. The task of decoherence suppression amounts to suitably choosing control functions such that $U^{\text{res}}(T)$ from Eq. (3.24) acts trivially on the system. One aims at achieving DD for a large class of decoherence possible Hamiltonians. In this context, one distinguishes between *universal* and *non-universal* dynamical decoupling schemes. In the former case, an effective evolution of the form (3.22) is guaranteed irrespective of the Hamiltonian, i.e., for all $H^{\text{orig}} \in \mathcal{B}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$ as stated in Definition 3.8. In contrast, non-universal schemes achieve dynamical decoupling only for a subset of Hamiltonians $\mathfrak{H} \subset \mathcal{L}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$, where Eq. (3.22) is satisfied for every $H^{\text{orig}} \in \mathfrak{H}$. Such non-universal schemes can still be highly beneficial for quantum information processing tasks if the dominant decoherence terms originate from a restricted set of Hamiltonians.

Typically, property (i) and (ii) from Definition 3.8 are achieved *approximately*.

Definition 3.9 *(Nth order DD).* A control scheme in the form of Definition 3.4 achieves *universal Nth order dynamical decoupling* (in a suitably chosen unitarily invariant norm $\|\cdot\|$ on $\mathcal{B}(\mathcal{H}_S \otimes \mathcal{H}_E)$) if and only if for all $H^{\text{orig}} \in \mathcal{B}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$, the effective evolution $U^{\text{res}}(T)$ from Eq. (3.15)

(i) involves no interaction between system and environment up to order N in T, i.e., there exist $U_S \in \mathsf{U}(\mathcal{H}_S)$ and $U_E \in \mathsf{U}(\mathcal{H}_E)$ such that

$$\|U^{\text{res}}(T) - U_S \otimes U_E\| = O(T^{N+1}) \ , \tag{3.20}$$

(ii) and acts trivially on the system up to order N in T, i.e.,

$$\|U_S - I_S\| = O(T^{N+1}) \ . \tag{3.21}$$

The above definition involves a suitably chosen unitarily invariant norm, e.g., the Frobenius norm.

To investigate Nth order DD, one may expand the resulting unitary $U^{\text{res}}(T)$ in either a Dyson or a Magnus series , cf. Definition 3.7. Eqs. (3.20) and (3.21) are satisfied if the first N terms in the Dyson series – that act non-trivially on the system – vanish in a suitably chosen norm, i.e.,

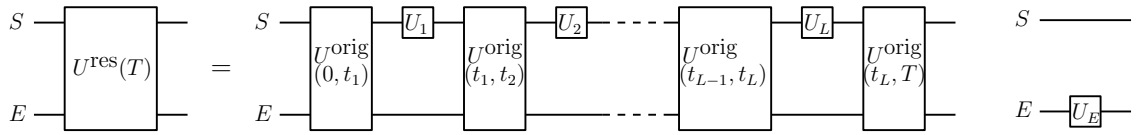
$$U^{\text{res}}(T) = I_S \otimes U_E + O(T^{N+1}) \ .$$

Similarly, if one considers the Magnus series of $U^{\text{res}}(T)$, then Nth order DD is satisfied if

$$U^{\text{res}}(T) = e^{iTH^{\text{eff}}} \qquad \text{where} \qquad H^{\text{eff}} = I_S \otimes H_E + O(T^{N+1}) \ . \tag{3.22}$$

Here, N is called the order of the DD scheme.

In Definition 3.9, the constants C and c usually depend on the norms of the original Hamiltonian $\|H^{\text{orig}}\|$ as well as the control Hamiltonian $\|H^{\text{control}}(t)\|$. If the environment Hilbert space is infinite-dimensional, one has to pay attention to the (un)boundedness of H^{orig}. We will neglect this issue here since in the newly introduced bosonic DD sequences in Chapter 4, we consider the series expansion of (finite) symplectic matrices and instead refer the interested reader to the literature (e.g. [11]).



(a) Resulting evolution $U^{\mathrm{res}}(T)$ (b) DD

Figure 3.1: Circuit representation of a DD sequence. The upper and the lower horizontal lines represent the system $\mathcal{H}_S$ and the environment $\mathcal{H}_E$, respectively, and may both consist of more than one qubit. (a) The resulting evolution $U^{\mathrm{res}}(T)$ from Eq. (3.24) corresponds to the uncontrolled system-environment evolution $U^{\mathrm{orig}}(\tau, s)$ interleaved with a pulse sequence of the form of Definition 3.5. (b) (Approximate) dynamical decoupling (DD) is achieved if $U^{\mathrm{res}}(T) = I_S \otimes U_E$ (approximately) for some $U_E \in \mathsf{U}(\mathcal{H}_E)$.

When considering DD sequences, one usually assumes that the control is exerted by applying unitary pulses. Recall the Definition 3.5 of a pulse sequence (or equivalently bang-bang control); throughout this thesis, we always assume this bang-bang control. The resulting evolution after time T under the Hamiltonian (3.14) and application of a pulse sequence from Definition is given by

$$U^{\mathrm{res}}(T) = e^{-i(T-t_L)H^{\mathrm{orig}}}(U_L \otimes I_E) \cdots e^{-i(t_2-t_1)H^{\mathrm{orig}}}(U_1 \otimes I_E)e^{-it_1 H^{\mathrm{orig}}} \ . \tag{3.23}$$

The free evolution of system and environment under the decoherence or original Hamiltonian H^{orig} is interleaved with the control pulses U_j instantaneously applied at time t_j for every $j = 1, \ldots, L$.

If the original Hamiltonian H^{orig} is time-dependent, every term of the form $e^{-i(t-s)H^{\mathrm{orig}}}$ in Eq. (3.23) should be replaced by the propagator $U^{\mathrm{orig}}(t, s)$ generated by the original Hamiltonian $H^{\mathrm{orig}}(t)$ from time s to time t. Then the resulting evolution is

$$U^{\mathrm{res}}(T) = U^{\mathrm{orig}}(T, t_L)(U_L \otimes I_E) \cdots U^{\mathrm{orig}}(t_2, t_1)(U_1 \otimes I_E)U^{\mathrm{orig}}(t_1, 0) \ . \tag{3.24}$$

Fig. 3.1 shows the circuit representation of such a pulse sequence.

Analysis of DD schemes

Let $\mathcal{H}_S$ be a finite-dimensional system coupled to the environment $\mathcal{H}_E$ by the original Hamiltonian H^{orig}. Consider a pulse sequence from Definition 3.5 and let system and environment evolve under $U^{\mathrm{res}}(t)$ from Eq. (3.23).

The DD properties of a pulse sequence are usually investigated as follows. Since we are interested in pulse sequences which approximately average out the effect of the decoherence Hamiltonian on the system, it is convenient to consider a transformation that explicitly removes the action of the control evolution, the so-called *toggling frame*. The evolution in the toggling frame is given by

$$U^{\mathrm{tf}}(t) \coloneqq (U^{\mathrm{control}}(t))^{\dagger} U^{\mathrm{res}}(t) \ .$$

It is generated by the toggling frame Hamiltonian

$$H^{\mathrm{tf}}(t) \coloneqq (U^{\mathrm{control}}(t))^{\dagger} H^{\mathrm{orig}} U^{\mathrm{control}}(t) \ .$$

The *toggling frame* corresponds to the interaction picture described by $H^{\mathrm{control}}(t)$. One usually aims to compute the effective Hamiltonian H^{eff} that satisfies

$$U^{\mathrm{tf}}(T) = e^{iTH^{\mathsf{eff}}} \ , \tag{3.25}$$

for a given decoupling time T and an original Hamiltonian. Here, H^{eff} plays the role of a an error action operator, in the sense that the 'error' after applying the pulse sequence, i.e., the distance between the actual evolution (3.25) and the desired evolution, can be expressed in terms of the norm of this operator.

If we consider a pulse sequence in the sense of Definition 3.5, then the control evolution is given by the product of pulses U_j applied up to the considered time t

$$U^{\text{control}}(t) = (U_k \cdots U_2 U_1) \otimes I_E = \prod_{j=1}^{K_t} U_j \otimes I_E \ , \qquad \text{for } t \in [t_k, t_{k+1}) \ ,$$

where $K_t = \max\{j \in \{1, \ldots, L\} \,|\, t \leq t_j\}$. Therefore the toggling frame Hamiltonian takes the form

$$H^{\text{tf}}(t) = \left(\prod_{j=1}^{K_t} U_j \otimes I_E \right)^{\dagger} H^{\text{orig}} \left(\prod_{j=1}^{K_t} U_j \otimes I_E \right) \ , \qquad \text{for } t \in [t_k, t_{k+1}) \ . \tag{3.26}$$

By definition of the pulse sequence 3.5, the toggling frame evolution satisfies $U^{\text{tf}}(T) = U^{\text{res}}(T)$. Let us expand the toggling frame evolution after time T in a Magnus series (cf. Definition 3.7)

$$(U^{\text{tf}})^{\mathsf{Mag}}(T) = e^{\sum_{k=0}^{\infty} \Omega_k(T)} \ .$$

The first two terms in the exponent are

$$\Omega_0(T) \coloneqq -i \int_0^T H^{\text{tf}}(t) dt = -i \int_0^T \left(\prod_{j=1}^{K_t} U_j \otimes I_E \right)^{\dagger} H^{\text{orig}} \left(\prod_{j=1}^{K_t} U_j \otimes I_E \right) d\tau \ ,$$

$$\Omega_1(T) \coloneqq -\frac{1}{2} \int_0^T \int_0^{\tau_1} \left[H^{\text{tf}}(\tau_1), H^{\text{tf}}(\tau_2) \right] d\tau_2 d\tau_1 \ .$$

The higher order terms $\Omega_k(T)$ involve nested commutators of $H^{\text{tf}}(t)$ each with $k+1$ factors of H^{orig} such that $\|\Omega_k(T)\| = O(T^k \|H^{\text{orig}}\|^k)$. The DD scheme achieves first order DD if $\Omega_1(T) \propto I_S \otimes B_E$ for some $B_E \in \mathcal{B}(\mathcal{H}_E)$. More generally, Nth order dynamical decoupling is achieved if for all $k < N$ there are $B_E \in \mathcal{B}(\mathcal{H}_E)$ such that $\Omega_k(T) \propto I_S \otimes B_E$.

Alternatively, $U^{\text{tf}}(T)$ can be expanded in a Dyson series (cf. Definition 3.7). Using Eq. (3.26), one obtains that

$$(U^{\text{tf}})^{\mathsf{Dy}}(T) = I_S \otimes I_E + \sum_{k=1}^{\infty} A_k(T) \ ,$$

where for $k \in \mathbb{N}$, the terms A_k are defined in Eq. (3.17) for the toggling frame Hamiltonian $H^{\text{tf}}(t)$. The pulse sequence achieves Nth order DD if for all $k < N$ there are $B_E \in \mathcal{B}(\mathcal{H}_E)$ such that $A_k \propto I_S \otimes B_E$.

To further compute the Dyson or Magnus series of H^{tf} one may impose some additional assumptions on the system-environment interactions H_{SE} in $H^{\text{orig}} = H_S \otimes I_E + H_{SE} + I_S \otimes H_E$: more precisely, let

$$H_{SE} = \sum_{\alpha} S_\alpha \otimes B_\alpha \in \mathcal{L}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E) \ , \tag{3.27}$$

where $S_\alpha \in \mathcal{B}(\mathcal{H}_S)$ and $B_\alpha \in \mathcal{B}(\mathcal{H}_E)$. If the system Hilbert space is that of multiple qubits, the operators S_α can be chosen as the (generalised) Pauli matrices.

Let us summarise the strategy to analyse whether a given pulse sequence achieves Nth order dynamical decoupling: consider the toggling frame evolution, expand it into either a Magnus series or a Dyson series 3.7 and compute the series expansions using some available assumptions on the decoherence Hamiltonian and the pulse sequence.

A simple example

Consider a single qubit with Hilbert space $\mathcal{H}_S = \mathbb{C}^2$ that undergoes pure dephasing noise, i.e., the decoherence Hamiltonian

$$H^{\text{orig}} = I_S \otimes B_0 + \sigma_z \otimes B_1 \ , \tag{3.28}$$

couples the qubit to a (not further specified) environment E with the two operators $B_0, B_1 \in \mathcal{L}_{sa}(\mathcal{H}_E)$. Consider furthermore the pulse sequence (Definition 3.5) where

$$L = 2 \quad \text{and} \quad U_1 = U_2 = \sigma_x, \ t_1 = T/2 \quad \text{and} \quad t_2 = T \ . \tag{3.29}$$

To analyse the DD properties of this pulse sequence, compute the toggling frame Hamiltonian

$$H^{\text{tf}}(t) \coloneqq U^{\text{control}}(t) H^{\text{orig}} \left(U^{\text{control}}(t)\right)^\dagger = \begin{cases} I_S \otimes B_0 + \sigma_z \otimes B_1 & \text{if } 0 \leq t < T/2 \ , \\ I_S \otimes B_0 - \sigma_z \otimes B_1 & \text{if } T/2 \leq t \leq T \ . \end{cases}$$

The first term in the Dyson as well as in the Magnus series of the toggling frame evolution involves

$$\int_0^T H^{\text{tf}}(t) dt = T I_S \otimes B_0 + \frac{T}{2}(\sigma_z \otimes B_1 - \sigma_z \otimes B_1) = T I_S \otimes B_0 \ .$$

Hence, the first order terms act trivially on the system. Thereby, we find

$$U^{\text{res}}(T) = U^{\text{tf}}(T) = I_S \otimes I_E + \int_0^T H^{\text{tf}}(t) dt + O(T^2) = I_S \otimes B_0 + O(T^2) \ .$$

As a consequence, the pulse sequence (3.29) achieves first order dynamical decoupling for pure dephasing noise as described by a Hamiltonian of the form (3.28). This simple example illustrates how one usually analyses dynamical decoupling orders of pulse sequences.

3.3.3 Periodic dynamical decoupling

In order to achieve an effective error suppression by dynamical decoupling, the time scale T should be sufficiently small such that the Magnus or Dyson series converges. In other words, the decoupling pulses must be applied fast enough.

If one aims at achieving decoupling after a *given* total time $T_{\mathsf{PDD}} \in [0, \infty)$, one may use *periodic dynamical decoupling* (PDD) sequences. More precisely, one picks a natural number $m \in \mathbb{N}$ and periodically applies m times a DD sequence with DD time T, i.e., one sets

$$T = \frac{T_{\mathsf{PDD}}}{m}$$

such that the cyclicity assumption

$$U^{\text{control}}(t + kT) = U^{\text{control}}(t) \qquad \text{for all } t \in [0, T_{\mathsf{PDD}}), \ k = 1, \ldots, m \ ,$$

on the control propagator is satisfied. This also gives meaning to the name *cycle time* for T. By choosing m large enough, one can make the cycle time T sufficiently small such that there is justified hope that the analysed Dyson or Magnus series converges.

Let a system interact with an environment via a decoherence Hamiltonian H^{orig} and consider a pulse sequence of times $\{t_j\}_{j=1}^L \subset [0, T]$ and pulses $\{U_j\}_{j=1}^L$. Then after m applications

of this decoupling cycle, one can use the theory of averaged Hamiltonians to compute the resulting evolution

$$U^{\mathrm{res}}(T_{\mathsf{PDD}}) = U^{\mathrm{res}}(mT) = \left(e^{-iH^{\mathsf{eff}}T}\right)^m \ .$$

Here, H^{eff} is the averaged Hamiltonian over a single cycle. Expanded in a Magnus series $H^{\mathsf{eff}} = \frac{1}{T}\sum_{k=0}^{\infty}\Omega_k$ (cf. Definition 3.7), its lowest order contribution is given by

$$\frac{1}{T}\Omega_0 \coloneqq \frac{1}{T}\int_0^T U^{\mathrm{control}}(t)^\dagger H^{\mathrm{orig}} U^{\mathrm{control}}(t)dt \ , \tag{3.30}$$

i.e., by the cycle average of the toggling frame Hamiltonian. The Magnus series converges if the kth order terms are of order $O(T^k) = O(1/m^2)$. In the limit $m \to \infty$, while keeping the total time $t = mT$ fixed, the lower order term Ω_0 dominates the higher order ones.

If a pulse sequence satisfies

$$\lim_{m\to\infty} U^{\mathrm{res}}(T)^m = I_S \otimes B \quad \text{for some } B \in \mathcal{B}(\mathcal{H}_E) \ ,$$

then it is said to achieve *DD in the limit of infinitely fast pulse application*. Note that a first order DD sequence achieves this for sufficiently small time T.

Let us consider the special case of a pulse sequence $\{jT/L\}_{j=1}^L$, $\{U_j\}_{j=1}^L$ of *equidistant pulses*. The toggling frame evolution after time T is

$$\begin{aligned} U^{\mathrm{res}}(T) &= (U_L \otimes I_E)e^{-i\frac{T}{L}H^{\mathrm{orig}}} \cdots (U_1 \otimes I_E)e^{-i\frac{T}{L}H^{\mathrm{orig}}} \\ &= (\widetilde{U}_L\widetilde{U}_{L-1}^\dagger \otimes I_E)e^{-i\frac{T}{L}H^{\mathrm{orig}}} \cdots (\widetilde{U}_1\widetilde{U}_0 \otimes I_E)e^{-i\frac{T}{L}H^{\mathrm{orig}}} \\ &= (\widetilde{U}_L \otimes I_E)\prod_{j=0}^{L-1}(\widetilde{U}_j^\dagger \otimes I_E)e^{-i\frac{T}{L}H^{\mathrm{orig}}}(\widetilde{U}_j \otimes I_E) \\ &= \prod_{j=0}^{L-1} e^{-i\frac{T}{L}(\widetilde{U}_j^\dagger \otimes I_E)H^{\mathrm{orig}}(\widetilde{U}_j \otimes I_E)} \end{aligned} \tag{3.31}$$

where we recursively define $\widetilde{U}_k$ via $\widetilde{U}_0 \coloneqq I_S$ and $\widetilde{U}_k \coloneqq U_k\widetilde{U}_{k-1}$ for $k = 1, \ldots, L$. Note that this implies that

$$\widetilde{U}_k \otimes I_E = U_kU_{k-1}\cdots U_1 \otimes I_E = U^{\mathrm{control}}(T_{\mathsf{PDD}}) \qquad \text{for } t \in [kT/L, (k+1)T/L)$$

and the relation $U^{\mathrm{control}}(T) = I_S$ implies $\widetilde{U}_L = I_S$.

Definition 3.10 *(DD set)***.** A *DD set* is a family of unitaries $\{V_j\}_{j=0}^L \subset \mathsf{U}(\mathcal{H}_S)$ such that $V_0 = I_S = V_L$ and which satisfies that for every $H^{\mathrm{orig}} \in \mathcal{B}_{sa}(\mathcal{H}_S \otimes \mathcal{H}_E)$ there is a $B \in \mathcal{B}_{sa}(\mathcal{H}_E)$ such that

$$\frac{1}{L}\sum_{j=0}^{L-1}(V_j^\dagger \otimes I_E)H^{\mathrm{orig}}(V_j \otimes I_E) \propto I_S \otimes B \ . \tag{3.32}$$

Consider $U^{\mathrm{res}}(T)$ from Eq. (3.31). Then after m periodic applications of the above pulse cycle, the effective evolution $[U^{\mathrm{res}}(T = T_{\mathsf{PDD}}/m)]^m$ converges by the generalised Trotter formula [154] in the limit of infinitely fast pulse application to

$$\lim_{m\to\infty}[U^{\mathrm{res}}(T_{\mathsf{PDD}}/m)]^m = e^{-iT_{\mathsf{PDD}}\frac{1}{L}\sum_{j=0}^{L-1}(\widetilde{U}_j^\dagger \otimes I_E)H^{\mathrm{orig}}(\widetilde{U}_j \otimes I_E)} \ . \tag{3.33}$$

Note that by the above considerations, a pulse sequence achieves DD in the limit of infinitely fast pulse application if the lowest order Ω_0/T of the Magnus series – which is given by the exponent in Eq. (3.33) – acts trivially on the system. Such a pulse sequence forms a first order DD scheme. As a consequence, we have just proven the following result.

Lemma 3.11. Let $\{V_j\}_{j=0}^{L}$ be a DD set in the sense of Definition 3.10. Then the pulse sequence

$$t_j \coloneqq jT/L \qquad \text{and} \qquad U_j \coloneqq V_j V_{j-1}^\dagger \qquad \text{for } j = 1, \ldots, L$$

achieves universal first order dynamical decoupling. □

We have just proven that a DD set satisfies first order DD. PDD develops its full potential when combining it with a group-theoretical averaging procedure as described in the next paragraph.

DD with unitary 1-designs and groups

In the above Definition 3.10 of DD sets, one searches for a finite subset of unitary operators that satisfy Eq. (3.32), more precisely that simulate the Haar measure of the unitary group on the quantum system. Unitary designs are finite collections of unitary matrices that reproduce expectation values or averages over the unitary group. Consequently it is useful to identify subsets of the unitary group that can adequately simulate the Haar measure for a given class of operational tasks.

Definition 3.12 *(Unitary design [43])***.** Let $n \in \mathbb{N}$ and $\mathsf{U}(n)$ be the unitary group. A *unitary t-design* of $\mathsf{U}(n)$ is a finite set $X = \{x_k\}_{k=1}^{|X|} \subset \mathsf{U}(n)$ of unitary operators on $\mathbb{C}^n$ such that for every polynomial $P_{(t,t)}(U)$ of degree at most t in the matrix elements of the unitaries U and at most t in the complex conjugate of those matrix elements satisfies

$$\frac{1}{|X|}\sum_{k=1}^{|X|} P_{(t,t)}(x_k) = \int_{\mathsf{U}(n)} P_{(t,t)}(x)dx$$

where on the right hand side one integrates with respect to the uniform Haar measure on $\mathsf{U}(n)$.

The theory of unitary t-designs was developed in 2006 by Dankert et al. [43] in the framework of randomised benchmarking [58], a technique to estimate the error rates on a system by application of sequences of random unitaries. In this context one only considers unitary 2-designs, but this concept was later extended to arbitrary $t \in \mathbb{R}$. An alternative definition of t-designs using concepts of representation theory is given in the work of Dankert et al. [43] as well.

Here, we are interested in 1-designs. Following Definition 3.12 for $t = 1$, a unitary 1-design is a finite subset $X = \{x_1, \ldots, x_{|X|}\} \subset \mathsf{U}(n)$ such that

$$\frac{1}{|X|}\sum_{k=1}^{|X|} x_k^\dagger H x_k = \frac{\mathrm{tr}(H)}{n} \tag{3.34}$$

holds for all $H \in \mathbb{C}^{n\times n}$.

Lemma 3.13 *(First order DD scheme).* Let $X = \{x_1, \ldots, x_L\}$ be a unitary 1-design of $\mathsf{U}(n)$ such that $I_S \in X$. Then the pulse sequence (cf. Definition 3.5)

$$t_j \coloneqq jT/L \qquad \text{and} \qquad U_j \coloneqq x_j x_{j-1}^\dagger \qquad \text{for } j = 1, \ldots, L \tag{3.35}$$

where one sets $x_0 = I_S = x_L$ achieves first order dynamical decoupling. □

This can be shown using the property of unitary 1-designs and works similarly to the proof of Lemma 3.11.

Proof. Assume that $H^{\text{orig}} = \sum_\alpha S_\alpha \otimes B_\alpha$ as in Eq. (3.27). By the property (3.34) of unitary 1-designs, one finds that for H^{orig}

$$\frac{1}{L}\sum_{j=1}^{L}(x_j^\dagger \otimes I_E)H^{\text{orig}}(x_j \otimes I_E) = \sum_\alpha \frac{\operatorname{tr}(S_\alpha)}{n} I_S \otimes B_\alpha = I_S \otimes B$$

where $B = \sum_\alpha \operatorname{tr}(S_\alpha)/nB_\alpha$. By Lemma 3.11 this implies that the pulse sequence (3.35) achieves first order DD – or equivalently DD in the limit of infinitely fast pulse application. ■

PDD has been considered before the introduction of unitary t-designs using a similar group-theoretical averaging procedure [166]. Consider a finite group $\mathcal{G} = \{g_1, \ldots, g_{|\mathcal{G}|}\}$ that acts on $\mathcal{H}_S$ with a faithful unitary representation $\{U_{g_1}, \ldots, U_{g_{|\mathcal{G}|}}\} \subset \mathsf{U}(\mathcal{H}_S)$ such that $U_{g_{|\mathcal{G}|}} = I_S$ and define $U_{g_0} = I_S$. Such a group is called a DD group if $\Pi_{\mathcal{G}}(A) = \lambda_A I_S$ for all $A \in \mathcal{B}(\mathcal{H}_S)$ where λ_A depends on A and where the average over $\mathcal{G}$ is defined as

$$\Pi_{\mathcal{G}}(A) \coloneqq \frac{1}{|\mathcal{G}|}\sum_{j=1}^{|\mathcal{G}|} U_{g_j}^\dagger A U_{g_j} \qquad \text{for } A \in \mathcal{B}(\mathcal{H}) \ .$$

Formally, $\Pi_{\mathcal{G}}(A) \in Z(\mathcal{G})$ for all $A \in \mathcal{B}(\mathcal{H}_S)$ where $Z(\mathcal{G})$ defines the centraliser of $\mathcal{G}$ (i.e., the commutant of the group algebra). The sequence of pulses

$$U_k \coloneqq U_{g_k} U_{g_{k-1}}^\dagger \qquad \text{applied at times} \qquad t_k \coloneqq kT/|\mathcal{G}|$$

for $k = 1, \ldots, |\mathcal{G}|$ satisfies the following: The first order term in the Magnus expansion of $U^{\text{res}}(T)$ which is a time average of the toggling frame Hamiltonian (cf. the calculation (3.31) in the previous paragraph) from Eq. (3.30) becomes the average over $\mathcal{G}$

$$(\Pi_{\mathcal{G}} \otimes I_E)(H^{\text{orig}}) \coloneqq \frac{1}{|\mathcal{G}|}\sum_{j=1}^{|\mathcal{G}|}\left(U_{g_j}^\dagger \otimes I_E\right) H^{\text{orig}} \left(U_{g_j} \otimes I_E\right) \ .$$

By Schurs Lemma for the case that $\mathcal{G}$ acts irreducibly on $\mathcal{H}_S$ the centraliser $Z(\mathcal{G})$ is trivial and hence

$$\Pi_{\mathcal{G},\mathsf{irred}}(A) = \frac{\operatorname{tr}(A)}{\dim(\mathcal{H}_S)} I_S \qquad \text{for all} \qquad A \in \mathcal{B}(\mathcal{H}_S) \ .$$

Applied to the case where the decoherence term H_{SE} in the original Hamiltonian is of the form of Eq. (3.27), we find that if S_α are traceless, then $\Pi_{\mathcal{G},\mathsf{irred}}(S_\alpha) = 0$. As a consequence, a DD group achieves first order DD.

More generally, this analysis implies that for $\dim(\mathcal{H}_S) < \infty$, one can always achieve universal first order DD: one chooses as a DD group a unitary error basis of $\mathcal{H}_S$ (such as the Pauli

group in the multi-qubit case), cf. e.g. [107, Chapter 4.4.2]. The only disadvantage of this result is that the size of such a DD group is exponentially large in the number of qubits, scaling like 4^n for $\mathcal{H}_S = (\mathbb{C}^2)^{\otimes n}$. As a consequence, such a DD sequence requires a large overhead since the number or necessary pulses grows exponentially with the system size even for first order DD or a single decoupling cycle. Note that the condition $\Pi_{\mathcal{G}}(A) = \lambda_A I_S$ for DD groups is similar to Eq. (3.34) for 1-designs. Comparing both, the latter definition may be more convenient, since a 1-design is smaller than a group. Hence a DD sequence originating from a 1-design requires less pulses and is in this sense more resource efficient than a DD group.

Another simple example

Unitary 1-designs on $\mathcal{H}_S = (\mathbb{C}^2)^{\otimes n}$ are called *unitary operator bases* of the unitary group $\mathsf{U}(2^n)$, cf. [93]; such sets are of size $L = (2^n)^2$, i.e., one needs $L = 2^{2n}$ pulses in order to achieve first order DD. For a single-qubit, i.e., for $n = 1$, such a unitary 1-design is given by

$$\{V_j\}_{j=0}^{4-1} = \{I_S, \sigma_x, \sigma_y, \sigma_z\}$$

which corresponds up to complex phases to the one-qubit Pauli group $\mathcal{P}_1$. To compute the corresponding pulse sequence according to Lemma 3.13, we use that $\sigma_z\sigma_y = -i\sigma_x$ and $\sigma_y\sigma_x = -i\sigma_z$. Hence

$$t_j = jT/L \quad \text{for } j = 1, \ldots, 4, \qquad \text{and} \qquad U_1 = U_3 = \sigma_x \quad , \quad U_2 = U_4 = \sigma_z \ , \tag{3.36}$$

achieves first order DD – or equivalently DD in the limit of infinitely fast pulse application. This pulse sequence is called the *universal decoupling* pulse sequence for a single qubit [166]. To shorten notation, it is commonly abbreviated as $\mathsf{f}X\mathsf{f}Z\mathsf{f}X\mathsf{f}Z = I\mathsf{f}IX\mathsf{f}XY\mathsf{f}YZ\mathsf{f}Z$ (order of pulse application from left to right) where f denotes the free (decoherence) evolution in between the pulses.

Random dynamical decoupling

Besides the deterministic DD sequences represented so far, there are also random DD schemes. This concept builds on DD groups. As introduced in 2005 by Viola and Knill [165] a specific realisation of a random DD scheme is given by a randomly chosen path of pulses over the DD group. The DD properties are independent of the realisation since Eq. (3.32) holds irrespective of the order of unitaries in the DD pulse sequence.

3.4 Quantum error correcting codes of qubit systems

Let us now consider a closed-loop control technique. Quantum error correcting codes were developed as a theoretical tool to protect quantum systems against noise. The core idea to protect information is to create redundancy and spread the information among many degrees of freedom. Given a system with many degrees of freedom, a quantum error correcting code is defined as a subspace (with fewer degrees of freedom) of this larger system. A crucial assumption for the procedure's success, i.e., it being able to protect the subspace from errors, is that errors usually happen only on a small number of these degrees of freedom. If the fraction of corrupted degrees of freedom is small enough, there is justified hope that a

measurement – called a syndrome measurement since it provides an error syndrome but no information on the actual encoded state – can reveal whether the encoded information has been corrupted and which correction operation should be applied in order to correct for the error. If the correction undoes the error, we say that the error has been successfully corrected, cf. Fig. 3.2 for a graphical representation of this idea.

3.4.1 Definition of quantum error correcting codes

Definition 3.14 *(Quantum code).* Let $\mathcal{H}_S$ be a quantum system. A *quantum code* or *quantum error correction code* (QECC) is defined as a subspace $\mathcal{C} \subset \mathcal{H}_S$.

We call a Hilbert space $\mathcal{H}_L$ which is isomorphic to the code space $\mathcal{C}$ the *logical Hilbert space*, i.e., if for example $\mathcal{C} \cong \mathbb{C}^d$, we say that the code $\mathcal{C}$ encodes a qudit into the physical Hilbert space $\mathcal{H}_S$. A partial isometry which satisfies

$$V : \mathcal{H}_L \to \mathcal{H}_S \qquad \text{such that} \qquad \mathcal{C} = V\mathcal{H}_S \coloneqq \text{Image}(V)$$

is called an encoding isometry of the code, but note that an encoding isometry is not unique. The subset $\mathcal{C}$ is called the *code space* an its (normalised) elements are the *code words* or encoded states. The code space $\mathcal{C} \subset \mathcal{H}_S$ is the subspace of the quantum system's Hilbert space $\mathcal{H}_S$ which 'encodes' a logical system into the (generally larger) physical system $\mathcal{H}_S$. To investigate the error correction property of a quantum code, let us specify the concept of recovery. A recovery may be able to correct some errors but fails to correct others.

Definition 3.15 *(Recovery of a QECC).* Let $\mathcal{C} \subset \mathcal{H}_S$ be a QECC with encoding isometry $V : \mathcal{H}_L \to \mathcal{H}_S$. Let furthermore $E : \mathcal{H}_S \to \mathcal{H}_S$ be a unitary error. A quantum channel $\mathcal{R} : \mathcal{D}(\mathcal{H}_S) \to \mathcal{D}(\mathcal{H}_S)$ is called a

(i) *valid recovery* for the error E if and only if for every $|\psi\rangle \in \mathcal{H}_L$

$$\text{supp}\Big(\mathcal{R}\big(EV|\psi\rangle\langle\psi|V^\dagger E^\dagger\big)\Big) \subseteq \mathcal{C} \ ; \tag{3.37}$$

(ii) *successful recovery* for the error E if and only if for all pure states $|\psi\rangle \in \mathcal{H}_L$ there is a complex number $c(E, |\psi\rangle)$ such that

$$\mathcal{R}\big(EV|\psi\rangle\langle\psi|V^\dagger E^\dagger\big) = c\big(E, |\psi\rangle\big)\, V|\psi\rangle\langle\psi|V^\dagger \ . \tag{3.38}$$

We say that the quantum code $\mathcal{C}$ is an *error correcting code for the error set* $\mathfrak{E} \subset \mathsf{U}(\mathcal{H}_S)$ if there is a (single) successful recovery $\mathcal{R}$ for all $E \in \mathfrak{E}$.

If there exists a successful recovery for an error set $\mathfrak{E}$, we also say that the quantum code (successfully) *corrects* the errors $E \in \mathfrak{E}$. Property (i) – validity of a recovery – ensures that every corrupted state is mapped back to a state inside the code space. Then a typical question is on which errors the valid recovery it is actually successful. Although the property of error correction only makes sense in the presence of errors and with respect to a recovery map, it is common to use the term quantum error correcting code and quantum code, interchangeably. One sometimes assumes that the recovery must be included in the definition of an error correcting code.

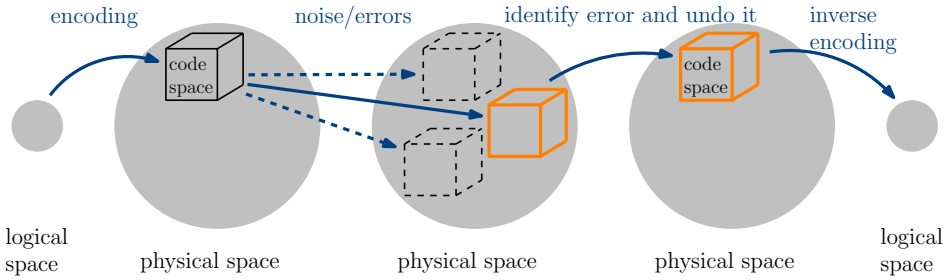


Figure 3.2: Schematic representation of a quantum error correcting code: A logical multi-qubit state is encoded into a state in the codespace – a subspace of a larger physical space. Ideally, the code is designed such that different errors map the codespace to different subspaces of the physical space which can be distinguished by a measurement. One may then identify which error has occurred (depicted in orange here) and then undo its effect by applying the inverse error as a correction. Generally, if the recovery maps the corrupted state back to the original logical state, then the error correction was successful.

Note that in Definition 3.15, validity and success of a recovery are formulated at the level of logical *pure* states $|\psi\rangle \in \mathcal{C}$. This is sufficient for most applications of QECC (such as those discussed within this chapter), but the generalisation to density matrices instead of pure states is straightforward: An encoding isometry $V : \mathcal{H}_L \to \mathcal{H}_S$ defines an *encoding channel*

$$\mathcal{E} : \mathcal{D}(\mathcal{H}_L) \to \mathcal{D}(\mathcal{H}_S) \qquad , \qquad \mathcal{E}(\rho) \coloneqq V\rho V^\dagger \ .$$

A single unitary error E defines the corresponding *noise channel* $\mathcal{N}_E(\rho) = E\rho E^\dagger$ and an error set $\mathfrak{E} \subset \mathsf{U}(\mathcal{H}_S)$ is related to a stochastic noise channel via the relations

$$\mathcal{N} : \mathcal{D}(\mathcal{H}_S) \to \mathcal{D}(\mathcal{H}_S) \qquad , \qquad \mathcal{N}(\rho) \coloneqq \sum_{E\in\mathfrak{E}} \pi(E) E\rho E^\dagger \ ,$$

where $\pi : \mathfrak{E} \to [0,1]$ is a probability distribution (cf. Section 3.1.1 for the definition of probabilistic noise channels). One calls a state $\rho \in \mathcal{D}(\mathcal{H}_S)$ which is supported on the code space $\mathcal{C}$ – it satisfies $P_\mathcal{C}\rho P_\mathcal{C} = \rho$ where $P_\mathcal{C} \coloneqq VV^\dagger$ is the projection onto $\mathcal{C}$ – an encoded state and $\mathcal{N}(\rho)$ a corrupted state. The recovery channel is defined as in Definition 3.15. It is *valid* for the noise channel $\mathcal{N}$ if and only if $(\mathcal{R}\circ\mathcal{N})(\rho)$ is supported on $\mathcal{C}$ for all $\rho \in \mathcal{D}(\mathcal{H}_S)$ supported on $\mathcal{C}$. Then the map $\mathcal{E}^\dagger \circ \mathcal{R} \circ \mathcal{N} \circ \mathcal{E}$ is well-defined on $\mathcal{D}(\mathcal{H}_L)$ – using that $\mathcal{E}^\dagger$ is defined on states ρ supported on the code space $\mathcal{C}$ as $\mathcal{E}^\dagger(\rho) = V^\dagger\rho V$ so that $\mathcal{E}\mathcal{E}^\dagger = I_\mathcal{C}$. One calls $\mathcal{E}^\dagger$ the *inverse encoding channel* and $\mathcal{E}^\dagger \circ \mathcal{R}$ the *decoder*. The recovery is successful for the noise channel $\mathcal{N}$ if and only if $(\mathcal{R}\circ\mathcal{N})(\rho) \propto \rho$ for all $\rho \in \mathcal{D}(\mathcal{H}_S)$ supported on $\mathcal{C}$. A (valid) recovery usually incorporates two steps: error detection and error correction. The former step is also called *syndrome extraction* and corresponds to measurement of the corrupted state $\mathcal{N}(\rho)$. But note that the encoded state must not be destroyed (in fact, not even altered) by such a measurement. The outcome of such a measurement – the syndrome – should only reveal information about the error that has happened and not about the actual encoded state. The second step – error correction – corresponds to an operation that depends on the syndrome which is applied to the corrupted state. The recovery is successful if it can correctly identify which type of error has happened and applies a correction which undoes the error, e.g. the inverse error.

A state on the logical Hilbert space undergoes the *logical noise channel*

$$\overline{\mathcal{N}} : \mathcal{D}(\mathcal{H}_S) \to \mathcal{D}(\mathcal{H}_S) \qquad , \qquad \overline{\mathcal{N}} \coloneqq \mathcal{E}^\dagger \circ \mathcal{R} \circ \mathcal{N} \circ \mathcal{E} \ , \tag{3.39}$$

which is the concatenation of the encoding channel $\mathcal{E} : \mathcal{D}(\mathcal{H}_L) \to \mathcal{D}(\mathcal{H}_S)$ – associated with the quantum error correcting code $\mathcal{C}$ – followed by the physical noise channel $\mathcal{N}$, the recovery channel $\mathcal{R}$ and the inverse encoding.

Remark on the terminology

We note that the terminology of QECC is used differently within the literature. What we call the inverse encoding map $\mathcal{E}^\dagger$ is sometimes referred to as the decoder and the combination $\mathcal{E}^\dagger \circ \mathcal{R}$ of the recovery and the inverse encoding channel as the recovery channel (cf. e.g. [107]). Some authors define a quantum error correcting code directly as a pair $(\mathcal{C}, \mathcal{R})$ of a code space and a valid recovery or as a pair $(\mathcal{C}, \mathfrak{E})$ of the code space and a correctable error set (an error set for which one can find a successful recovery).

3.4.2 Quantum error correcting condition

Let $\mathcal{H}_S = (\mathbb{C}^2)^{\otimes n}$ for $n \in \mathbb{N}$, i.e., the quantum codes are defined on an n-qubit system. We consider $(2^n, m)$-codes, i.e., codes which encode an m-dimensional subspace $\mathcal{C} \subset \mathcal{H}_S$ into the 2^n-dimensional Hilbert space of n qubits. Usually $m = 2^k$, i.e., k logical qubits are encoded into n physical qubits.

A central question of quantum error correcting codes (QECC) is to identify the correctable error sets for a QECC $\mathcal{C}$, i.e., to find a necessary and sufficient condition on an error set $\mathfrak{E}$ such that there exists a successful recovery $\mathcal{R}$ for every error $E \in \mathfrak{E}$.

Theorem 3.16 *(Knill-Laflamme quantum error correction condition).* Let $\mathcal{C} \subset \mathcal{H}_S$ be a quantum code with encoding isometry $V : \mathcal{H}_L \to \mathcal{H}_S$ and $P_\mathcal{C} \coloneqq VV^\dagger : \mathcal{H}_S \to \mathcal{H}_S$ the projection onto the code space $\mathcal{C}$. Let furthermore $\mathfrak{E}$ be a linear space of errors with basis $\{E_k\}_k$.

A necessary and sufficient condition for the existence of a successful recovery (cf. Definition 3.15) for all $E \in \mathfrak{E}$ is that for all k, l

$$P_\mathcal{C} E_k^\dagger E_l P_\mathcal{C} = A_{kl} P_\mathcal{C} \tag{3.40}$$

where $A \coloneqq (A_{kl})$ is a complex Hermitian matrix. □

For the proof of this famous result [19, 96], we refer to the literature. Eq. (3.40) is called the *Knill-Laflamme error correction condition*. If the matrix A in Eq. (3.40) has full rank, then the respective code is called *non-degenerate*, otherwise it is *degenerate*.

In practice, one is usually given a probabilistic noise channel

$$\mathcal{N} : \mathcal{D}(\mathcal{H}_S) \to \mathcal{D}(\mathcal{H}_S) \quad , \quad \mathcal{N}(\rho) = \sum_{E \in \mathfrak{E}} p_E E \rho E^\dagger$$

for a probability distribution $p_E : \mathfrak{E} \to [0, 1]$. For a given valid recovery $\mathcal{R}$, the error set $\mathfrak{E}$ has a decomposition with respect to error operators some of which satisfy Eq. (3.40) (i.e., they are correctable) and some of which are uncorrectable. The task is construct a recovery which maximises the probability that a randomly drawn error is successfully corrected.

For $n \in \mathbb{N}$ physical qubits, the Pauli group $\mathcal{P}_n$ from Definition 3.1 forms a set on which the error correction condition can be tested. One way to organise the errors is according to their weights.

Definition 3.17 *(Weight and distance).* The *weight* $\mathrm{wt}(E)$ of an element $E \in \mathcal{P}_n$ is the number of qubits on which E acts non-trivially. The *distance of a quantum code* $\mathcal{C}$ is the minimal weight of an error $E \in \mathcal{P}_n$ with the property that there are states $|\psi_1\rangle, |\psi_2\rangle \in \mathcal{C}$ such that $\langle\psi_1|E|\psi_2\rangle \neq c(E)\,\langle\psi_1|\psi_2\rangle$ where $c(E)$ is a constant that may depend on E.

The distance quantifies the amount of noise a QECC can tolerate in the following sense: Assume that a quantum code $\mathcal{C}$ can correct (in the sense of Theorem 3.16) all errors with weight smaller or equal than t but that this is no longer true for errors with weight $t+1$. Then $2t+1$ is the smallest weight of an operator $E \coloneqq E_k^\dagger E_l$ such that the error correction condition $P_\mathcal{C} E P_\mathcal{C} = A_{kl} P_\mathcal{C}$ is violated. Such a code has distance $d = 2t+1$. It takes $2t+1$ single-qubit operations (weight-1-Pauli operators) to transform one codeword into the other. A quantum code $\mathcal{C}$ that encodes k logical qubits into n physical qubits (i.e., $\mathcal{H}_L = (\mathbb{C}^2)^{\otimes k}$ and $\mathcal{H}_S = (\mathbb{C}^2)^{\otimes n}$) and has distance d is called an $[[n,k,d]]-$code. As discussed above, $d = 2t+1$ implies that the code can successfully correct any errors with weights up to t.

Example: the three qubit bit flip code

One of the simplest examples of a quantum error-correcting code is the *three-qubit bit-flip code* or simply called bit flip code [130].

It encodes one logical qubit $\mathcal{H}_L = \mathbb{C}^2$ into three physical qubits $\mathcal{H}_S = (\mathbb{C}^2)^{\otimes 3}$ in a way that is robust to bit-flip errors. More precisely, its code space

$$\mathcal{C} \coloneqq \{\alpha|000\rangle + \beta|111\rangle \mid \alpha, \beta \in \mathbb{C}\}\ .$$

is two dimensional and spanned by the two logical states $\overline{|0\rangle} \coloneqq |000\rangle$ and $\overline{|1\rangle} \coloneqq |111\rangle$. The canonical encoding isometry $V : \mathcal{H}_L \to \mathcal{H}_S$ acts as $\alpha|0\rangle + \beta|1\rangle \mapsto \alpha|000\rangle + \beta|111\rangle$ for $\alpha, \beta \in \mathbb{C}$ such that $|\alpha|^2 + |\beta|^2 = 1$. Let us consider the i.i.d. noise channel of bit flips (cf. Eq. (3.7) for the definition of $\mathcal{N}_{\text{flip}}$)

$$\mathcal{N} = \mathcal{N}_{\text{flip}}^{\otimes 3}\ . \tag{3.41}$$

On every single qubit a bit flip σ_x-error occurs with probability $p \in [0,1]$.

Table 3.1 shows the possible errors in the first line, their respective probability according to the noise channel (3.41) in the second line and the corrupted states in the next two lines. The corrupted states of no or a single bit flip error (rows 2-5) are all orthogonal to each other. Hence they uniquely determine the tuple $(E, |\psi\rangle)$, i.e., knowing the corrupted state, one knows from which original state it was produced by which error.

Let us define the recovery map of this code. It will be designed to successfully correct errors that are bit flips on one of the three qubits. This corresponds to the errors in the first to fourth column (none or one bit flip) in Table 3.1. The recovery procedure consists of two parts, the syndrome measurement and the correction. The former corresponds to the measurement defined by the four projectors P_0, P_1, P_3, P_4 where P_0 projects on the subspace of uncorrupted states and P_i on the subspace of $\mathcal{H}_S$ which differs from the encoded states by a single bit flip on qubit $i = 1,2,3$. This projective measurement corresponds to measuring the two observables

$$S_1 \coloneqq Z_1 Z_2 \qquad , \qquad S_2 \coloneqq Z_2 Z_3$$

with outcomes s_1 and s_2, respectively. As both operators S_1 and S_2 have eigenvalues ± 1, the measurement produces one of four possible outcomes $(s_1, s_2) \in \{-1,1\}^2$ (just as measuring $P_0, \ldots, P_3$ before). Measuring S_1 corresponds to comparing the σ_z-eigenvalues of the

error E	I	X_1	X_2	X_3	X_1X_2	X_1X_3	X_2X_3	$X_1X_2X_3$
prob. $\pi(E)$	$(1-p)^3$	$p(1-p)^2$			$p^2(1-p)$			p^3
corrupted state	$\|000\rangle$	$\|100\rangle$	$\|010\rangle$	$\|001\rangle$	$\|110\rangle$	$\|101\rangle$	$\|011\rangle$	$\|111\rangle$
	$\|111\rangle$	$\|011\rangle$	$\|101\rangle$	$\|110\rangle$	$\|001\rangle$	$\|010\rangle$	$\|100\rangle$	$\|000\rangle$
syndrome s	1,1	-1,-1	-1,1	1,-1	1,-1	-1,1	-1,-1	1,1
$C(s)$	I	X_1	X_2	X_3	X_3	X_2	X_1	I
corrected state	$\|000\rangle$	$\|000\rangle$	$\|000\rangle$	$\|000\rangle$	$\|111\rangle$	$\|111\rangle$	$\|111\rangle$	$\|111\rangle$
	$\|111\rangle$	$\|111\rangle$	$\|111\rangle$	$\|111\rangle$	$\|000\rangle$	$\|000\rangle$	$\|000\rangle$	$\|000\rangle$

Table 3.1: Error detection and projection process of the three-qubit bit-flip code under the noise channel $\mathcal{N}$ from Eq. (3.41) of single or multi-qubit bit-flip errors: Error operators E (first line) and the probabilities $\pi(E)$ with which they occur (second line). The next two lines show the corrupted basis states $E|000\rangle$ and $E|111\rangle$, respectively. The syndrome $s = (s_1, s_2)$ associated with measurement of the two observables Z_1Z_2 and Z_2Z_3 (line 5) is equal for both encoded states $|000\rangle$ and $|111\rangle$, i.e., it does not reveal any information about the encoded state. The correction operation is given by $C(s)$. The last two lines show the post-correction basis states. The recovery is successful if it coincides with the entry in the second row, i.e., the three-qubit bit-flip code can successfully correct single qubit bit flips (row 3-5) but two- or three-qubit bit-flips result in a logical σ_x-error.

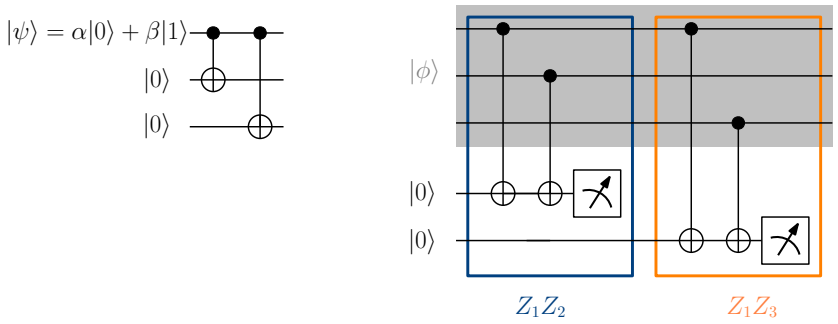
first and second physical qubit, returning outcome $+1$ if they are equal and -1 if they differ. Similarly, S_2 measures whether the bit values of qubits one and three coincide. Hence measuring S_1 and S_2 reveals which single qubit is in minority. Fig. 3.3(b) shows a circuit implementing this syndrome measurement: for each of the two measurements of S_1 and S_2, an additional auxiliary qubit is initialised in the state $|0\rangle$, the 'error' is transferred to this auxiliary qubit and it is successively measured in the computational basis.

Depending on the syndrome $s = (s_1, s_2)$, one applies a correction operation $C(s)$ as specified in Table 3.1. Since all errors E are tensor products of Pauli matrices, they are self-inverse, i.e., the error itself applied a second time gives a successful correction. Part of the correction is a majority vote of the σ_z eigenvalues of all three qubits. Successive application of the syndrome measurement and the correction defines the recovery map $\mathcal{R}$. This recovery is valid – satisfying (i) of Definition 3.15 for all errors $E \in \mathfrak{E}$ (cf. the last two lines of Table 3.1 which only contain elements of $\mathcal{C}$ for all errors). If at most one qubit is bit flipped, then this recovery is successful, satisfying (ii) of Definition 3.15 (cf. the last two lines of Table 3.1) and if two or more bit flips occur then the channel $\mathcal{R} \circ \mathcal{N}$ acts as a bit flip on the logical qubit.

Let us now quantify the performance of the single-qubit bit-flip code under the noise channel from Eq. (3.41). The success probability of the code P_{success} is given by the sum of successfully correctable errors. In terms of the probability p of a bit-flip on a single qubit, successfully correctable and uncorrectable errors occur with probability $(1-p)^3 + 3p(1-p)^2$ and $3(1-p)p^2 + p^3 = 3p^2 - 2p^3$, respectively, and hence $P_{\mathsf{success}} = (1-p)^3 + 3p(1-p)^2$. We will compare this success probability with the success probability $1-p$ when no encoding is used. If $p < 1/2$ then $(1-p)^3 + 3p(1-p)^2 > 1-p$. Hence the probability of a failure reduces from p (no error protection) to $3p^2 - 2p^3 \leq 3p^2$ (using the three-qubit bit-flip code).

More advanced quantum error correcting codes

Bit flip errors are not the only kind of errors: There exists a simple variant of the three-qubit bit-flip code which protect against single-qubit phase flips (cf. dephasing errors from



(a) Encoding of the logical state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ into $\alpha|000\rangle + \beta|111\rangle \in \mathcal{H}_S$.

(b) Stabiliser measurement of $S_1 = Z_1Z_2$ (blue) and $S_2 = Z_1Z_3$ (orange).

Figure 3.3: Quantum circuits realising the three qubit bit flip encoding map (a) and the syndrome measurement (b). For the latter, the upper three qubits in state $|\phi\rangle \in \mathcal{H}_S$ represent the physical qubits and the lower two are auxiliary qubits used for the syndrome measurements.

Eq. (3.28)) instead of bit flips, the three-qubit phase-flip code. The concatenation (cf. Section 3.4.4) of both codes – bit flip code and phase flip code – defines the nine qubit Shor code [143] which was one of the first quantum error correcting codes. It corrects one single qubit Pauli error, may it be bit flip, phase flip or both.

Other fundamental QECC include the Steane code [147] as well as subsystem codes [102, 133] (also called operator quantum error correction) where information is encoded in subsystems instead of single states, incorporating passive techniques such as *decoherence free subspaces* with active QECC. Directly relevant for this thesis are the topological codes and qubit-into-CV encodings (more precisely, the GKP code) which we discuss in more detail in Sections 3.4.5 and 3.5, respectively.

3.4.3 Stabiliser formalism

A natural next question is how to construct quantum error correcting codes, i.e., how to identify 'good' encoding and recovery maps for a given error set. In this context, the *stabiliser formalism* [70] offers a useful framework which allows to formulate and analyse quantum error correcting codes in an elegant way. Many codes, such as the three qubit bit flip code from Section 3.4.2, can be formulated within this framework. Stabiliser codes are a class of quantum error correcting codes that were originally developed as qubit error correcting codes and later extended to more general systems.

Definition 3.18 *(Stabiliser group and code).* Let $\mathcal{H}_n = (\mathbb{C}^2)^{\otimes n}$ be the Hilbert space of $n \in \mathbb{N}$ physical qubits and consider the Pauli group $\mathcal{P}_n$ from Definition 3.1.

- A subgroup $\mathcal{S} \subset \mathcal{P}_n$ is a *stabiliser group* if and only if it is abelian and it does not contain the operator $-I^{\otimes n}$. Its elements are called *stabilisers* and the elements of a generating set are called *stabiliser generators*. A set $\mathcal{C}_S \subset \mathcal{H}_n$ is said to be *stabilised* by $\mathcal{S}$ if and only if it consists of elements of $\mathcal{H}_n$ which are invariant under action of elements of $\mathcal{S}$.
- A *stabiliser code* on n qubits is the quantum error correcting code $\mathcal{C}_S \subset \mathcal{H}_n$ being stabilised by the stabiliser group $\mathcal{S}$.

A group $\mathcal{G}$ is said to be generated by $g_1, \ldots, g_m \in \mathcal{G}$ if every element $g \in \mathcal{G}$ can be written as $g = \prod_{j \in J} g_j$. A generating set $\{g_1, \ldots, g_m\}$ is called minimal if removing one of the g_i would make the generated group smaller. One writes $\mathcal{G} \coloneqq \langle g_1, \ldots, g_m \rangle$ and calls $g_1, \ldots, g_m \in \mathcal{G}$ the generators. Note that a set of generators determines the group but that the set of generators of a group is usually not unique.

If $\mathcal{S} \coloneqq \langle g_1, \ldots, g_{n-k} \rangle \subset \mathcal{P}_n$ is a stabiliser group, then the stabiliser code space $\mathcal{C}_S$ is defined as the joint $+1$ eigenspace of the stabiliser generators $g_1, \ldots g_{n-k}$. If the set of generators is chosen to be minimal then $\dim \mathcal{C}_S = 2^k$. The code space is isomorphic to $\mathcal{H}_L \coloneqq (\mathbb{C}^2)^{\otimes k}$ and the stabiliser code encodes k logical qubits into n physical qubits. In the following, we assume that the generators $g_1, \ldots g_{n-k}$ are tensor products of single-qubit Pauli operators omitting the pre-factors $-1, \pm i$. Since different Pauli matrices anti-commute, elements of $\mathcal{P}_n$ (and hence all stabiliser elements) either commute or anti-commute.

The logical operators are those Pauli operators that leave the codespace $\mathcal{C}_S$ invariant, i.e., such operators $\overline{L} \in \mathcal{P}_n$ should satisfy

$$\overline{L}|\psi\rangle \in \mathcal{C}_S \text{ for all } |\psi\rangle \in \mathcal{C}_S \ .$$

They are elements of the centraliser of $\mathcal{S}$ inside the Pauli group $\mathcal{P}_n$

$$Z(\mathcal{S}) \coloneqq \{A \in \mathcal{P}_n \mid AS = SA \text{ for all } S \in \mathcal{S}\} \ . \tag{3.42}$$

Since stabilisers act trivially on the codespace, non-trivial logical action comes from operators in $Z(\mathcal{S}) \setminus \mathcal{S}$. These are the kind of errors which the stabiliser code cannot detect. However, note that there is still redundancy in this definition of logical operators since elements of $Z(\mathcal{S}) \setminus \mathcal{S}$ can be written as a product of a stabiliser and an element which is not a stabiliser. Since $\mathcal{S}$ is abelian, the centraliser of $\mathcal{S}$ is equal to the normaliser of $\mathcal{S}$

$$N(\mathcal{S}) \coloneqq \{A \in \mathcal{P}_n \mid A\mathcal{S} = \mathcal{S}A\}$$

and it can be shown that the quotient group $Z(\mathcal{S})/\mathcal{S}$ contains 4^k logical operators $\overline{L} \in Z(\mathcal{S})/\mathcal{S}$ up to phases (cf. e.g. [6, Section 1.3.2]). Let us call these 4^k elements $\overline{L}_1, \ldots, \overline{L}_{4^k}$. Given a Pauli operator $E \in \mathcal{P}_n$, the set $E\mathcal{S} = \{ES | S \in \mathcal{S}\}$ is called a coset of $\mathcal{S}$. For an analysis of the logical effect of an error $E \in \mathcal{P}_n$ on the code space, it is useful to consider the cosets

$$EZ(\mathcal{S}) = \{EA \mid A \in \mathcal{P}_n \text{ such that } AS = SA \text{ for all } S \in \mathcal{S}\} = \bigcup_{i=1}^{4^k} E\ \overline{L_i}\mathcal{S} \tag{3.43}$$

which are the disjoint union (right hand side) of cosets $E\ \overline{L_i}\mathcal{S}$ of $\mathcal{S}$ for $\overline{L_i} \in Z(\mathcal{S})/\mathcal{S}$.

Recall from Definition 3.17 that the distance of a stabiliser code is defined as the number of single-qubit changes it takes to get from one code word to the other, i.e., in a stabiliser code it is the minimal weight of non-trivial logical operators

$$\min_{\overline{L} \in Z(\mathcal{S})/\mathcal{S}} \mathrm{wt}(\overline{L}) \ .$$

We consider the standard stabiliser code error correction procedure: An encoded state ρ undergoes noise, i.e., it is mapped by an n-qubit noise channel $\mathcal{N} : \mathcal{D}((\mathbb{C}^2)^{\otimes n}) \to \mathcal{D}((\mathbb{C}^2)^{\otimes n})$ to a corrupted state $\mathcal{N}(\rho)$. The recovery procedure in the stabiliser framework is divided into two steps: syndrome measurement and correction operation. The syndrome measurement comprises measurements of a set of stabiliser generators $g_1, \ldots, g_{n-k}$. We call the outcome of such measurements the *error syndrome* $s = (s_1, \ldots, s_{n-k})$. Since all stabilisers are elements

of the Pauli group $\mathcal{P}_n$, they have eigenvalues $+1$ or -1. We associate $s = 0$ with the eigenvalue $+1$ and $s = 1$ with the eigenvalue -1 so that we have $s_i \in \{0, 1\}$ for all $i = 1, \ldots, n-k$. In more detail, for an encoded state $\psi \in \mathcal{C}$ and an error $E \in \mathcal{P}_n$, the corrupted state $E\psi$ is an eigenstate of g_i with eigenvalue $(-1)^{s_i}$. More generally,

$$s_i = \begin{cases} 0 & \text{if} \quad [E, g_i] = 0 \\ 1 & \text{if} \quad \{E, g_i\} = 0 \end{cases}$$

for $i = 1, \ldots, n-k$ where $[\cdot, \cdot]$ is the commutator and $\{\cdot, \cdot\}$ is the anti-commutator. A syndrome $s \in \{0, 1\}^{n-k}$ occurs with probability

$$p(s) \coloneqq \operatorname{tr}\left(\Pi(s)\mathcal{N}(\rho)\right) \quad \text{where} \quad \Pi(s) \coloneqq \prod_{i=1}^{n-k} \frac{1}{2}\left(I + (-1)^{s_i} g_i\right) \ ,$$

for the encoded state ρ supported on $\mathcal{C}_\mathcal{S}$. In this case, the post-measurement state is given by $p(s)^{-1}\Pi(s)\mathcal{N}(\rho)\Pi(s)$. Note that since the code space is the $+1$-eigenspace of the stabilisers, a measurement of their eigenvalues reveals whether a corrupted state is still inside the code space; however, it cannot reveal which encoded state it is exactly.

Finally, a unitary correction operation $C(s) \in \mathcal{P}_n$ which depends on the syndrome s is applied to the post-measurement state. The *recovery channel* $\mathcal{R}$ of a stabiliser code $\mathcal{C}_\mathcal{S} \subset \mathcal{H}_n$ is uniquely determined by the choice of a *correction* map

$$C : \{0, 1\}^{n-k} \to \mathcal{P}_n \quad , \quad s \mapsto C(s) \ ,$$

i.e., to every syndrome s one associates a Pauli correction $C(s) \in \mathcal{P}_n$. More precisely, the recovery map $\mathcal{R}$ incorporates both syndrome measurement and correction such that with probability $p(s)$ a corrupted state $\rho \in \mathcal{D}(\mathcal{H}_n)$ is transformed as

$$\rho \mapsto p(s)^{-1} C(s)\Pi(s)\rho\Pi(s)C(s)^\dagger$$

for the syndrome outcome s. The recovery map hence is

$$\mathcal{R}(\rho) = \sum_{s \in \{0,1\}^{n-k}} C(s)\Pi(s)\rho\Pi(s)C(s)^\dagger \tag{3.44}$$

for $\rho \in \mathcal{D}(\mathcal{H}_n)$. Since the error syndrome s is known after the recovery, one may write the post-recovery state (3.44) as $\sum_s C(s)\Pi(s)\rho\Pi(s)C(s)^\dagger \otimes |s\rangle\langle s|$ where the orthogonal projection $|s\rangle\langle s| \in \mathcal{B}(\mathbb{C}^2)$ incorporates the classical information (in contrast to quantum information) about the measurement outcome.

For each syndrome $s \in \{0, 1\}^{n-k}$, one may choose a representative error $E_s \in \mathcal{P}_n$ which causes this syndrome. Let us partition the coset $E_s Z(\mathcal{S})$ into

$$E_s Z(\mathcal{S}) = \bigcup_{i=1}^{4^k} E_s \overline{L_i} \mathcal{S} \tag{3.45}$$

using Eq. (3.43).

Lemma 3.19. Let $\mathcal{C}_\mathcal{S} \subset \mathcal{H}_n$ be a stabiliser code for a stabiliser group $\mathcal{S} \subset \mathcal{P}_n$ with a minimal set of generators $\{g_1, \ldots, g_{n-k}\} \subset \mathcal{S}$. Then, the recovery map $\mathcal{R}$ determined by these generators and the correction map $s \mapsto C(s)$ is

(i) valid for an error $E \in \mathcal{P}_n$, i.e., Eq. (3.37) is satisfied, if and only if $C(s)Z(\mathcal{S}) = EZ(\mathcal{S})$ where s is the syndrome caused by the error E;

(ii) successful for an error $E \in \mathcal{P}_n$, i.e., Eq. (3.38) is satisfied, if and only if E and $C(s)$ belong to the same coset on the right hand side of Eq. (3.45) where s is the syndrome of the error E and E_s a representative error for s.

□

Proof. A recovery is defined as valid for an error E if it maps back a corrupted state to the code space. This is the case if $C(s)$ causes the same syndrome as E. But this is exactly the condition (i).

The recovery is successful if furthermore the recovered state is proportional to the encoded state. This is the case if the actual error E and the correction operation $C(s)$ have the same logical effect on the code space (such that the correction undoes the effect of the error): E and $C(s)$ belong to the same coset on the right hand side of Eq. (3.45). ■

In slight abuse of terminology, we use the expression *decoding strategy or decoder* in the context of a stabiliser codes, both for the recovery channel $\mathcal{R}$ and the correction function $s \mapsto C(s)$.

Different decoding strategies

Assume that $n-k$ stabiliser generators $g_1, \ldots, g_{n-k}$ are fixed. Let $s \mapsto C(s)$ be a valid decoding strategy for a stabiliser code $\mathcal{C}_\mathcal{S}$ (as mentioned before, the choice $C(s)$ is sufficient to define a decoder). Recall the partition of the coset $E_s Z(\mathcal{S})$ from Eq. (3.45) into a disjoint union of subsets $E_s \overline{L}_i \mathcal{S}$. For every syndrome s define the quantity $\mathcal{C}_{\text{dec}}(s)$ as the subset $\mathcal{C}_{\text{dec}}(s) \in \{E_s\mathcal{S} \cup E_s\overline{L}_1\mathcal{S} \cup \cdots \cup E_s\overline{L}_{4^k-1}\mathcal{S}\}$ from Eq. (3.45) such that $C(s) \in \mathcal{C}_{\text{dec}}(s)$. It actually does not matter which of the elements $C(s)$ of a coset $\mathcal{C}_{\text{dec}}(s)$ one chooses since they all have equivalent logical action on the code space.

If we consider the performance of a decoder with correction $C(s) \in \mathcal{C}_{\text{dec}}(s)$ under a probabilistic error model

$$\mathcal{N}_\pi(\rho) = \sum_{E \in \mathcal{P}_n} \pi(E) E \rho E^\dagger$$

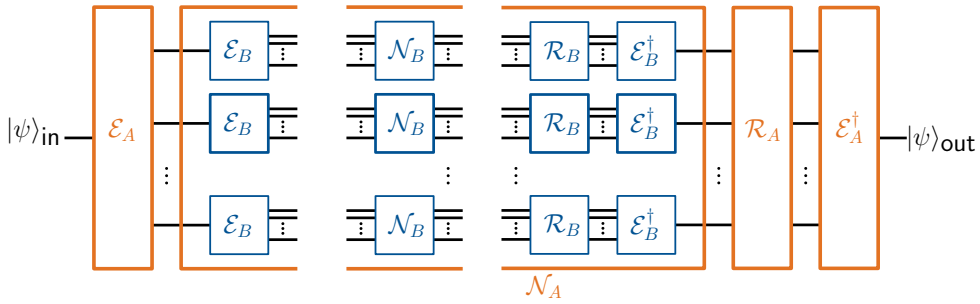
with error distribution $\pi : \mathcal{P}_n \to [0,1]$, then the average success probability is a measure of its error tolerance. The *average success probability* is defined as the probability that a Pauli error – randomly chosen with respect to the probability distribution π – is successfully corrected by the decoder. Using the decomposition of cosets from Eq. (3.45) and Lemma 3.19, it is given by the sum of the probability that $E \in \mathcal{C}_{\text{dec}}(s)$ over all E, i.e.,

$$P_{\mathsf{success}} = \sum_E \text{Prob}[E \in \mathcal{C}_{\text{dec}}(s)] = \sum_{E \in \mathcal{C}_{\text{dec}}(s)} \pi(E) = \sum_{s \in \{0,1\}^m} \pi(\mathcal{C}_{\text{dec}}(s)) \ . \tag{3.46}$$

One calls $P_{\mathsf{err}} = 1 - P_{\mathsf{success}}$ the logical failure probability.

The map $s \mapsto C_{\text{ML}}(s) \in \mathcal{C}_{\text{dec}}(s)$ which maximises P_{success} is called the *maximum likelihood decoder*. Here, for every syndrome s one chooses the coset from the right hand side of Eq. (3.45) which is most probable given a syndrome s, i.e.,

$$\mathcal{C}_{\text{ML}}(s) \coloneqq \text{argmax}_{\mathcal{A} \in \{E_s\mathcal{S}, E_s\overline{L}_1\mathcal{S}, \ldots, E_s\overline{L}_{4^k-1}\mathcal{S}\}} \pi(\mathcal{A}) \ .$$



(a) Encoding of the logical state $|\psi\rangle_{\text{in}}$

(b) Physical Noise

(c) Example of decoding using the recoveries of $\mathcal{C}_A$ and $\mathcal{C}_B$

Figure 3.4: Circuit presentation of the concatenation of code $\mathcal{C}_A$ on the outer (top) level and code $\mathcal{C}_B$ on the inner (base) level. The recovery of errors from the noise channel $\mathcal{N}_B^{\otimes n}$ is successful if $|\psi\rangle_{\text{in}} = |\psi\rangle_{\text{out}}$.

3.4.4 Code concatenation

One possibility to further enhance the noise tolerance is to concatenate several layers of quantum error correcting codes. One may concatenate one code with itself to improve its noise tolerance against the one error model it is designed for (cf. Section 3.4.6) or one may combine different codes on different layers to protect the encoded system against different noise models. Indeed, the first threshold theorems for quantum fault-tolerance were based on concatenated codes [95, 2, 97, 99, 91]. A difficulty with this approach is the increasing complexity of corresponding decoders.

Let us briefly define the concatenation of two codes $\mathcal{C}_A \subset \mathcal{H}_S^A$ and $\mathcal{C}_B \subset \mathcal{H}_S^B$.

Definition 3.20 *(Concatenated code).* The concatenated code $\mathcal{C}_{AB}$ of an inner code $\mathcal{C}_B \subset \mathcal{H}_S^B$ and an outer code that satisfies $\mathcal{C}_A \subset \mathcal{H}_S^A \cong \left(\mathcal{H}_L^B\right)^{\otimes n}$ for some $n \in \mathbb{N}$ is defined by the combined encoding map

$$\mathcal{E}_{AB} \coloneqq \mathcal{E}_B^{\otimes n} \circ \mathcal{E}_A : \mathcal{D}\left(\mathcal{H}_L^A\right) \xrightarrow{\mathcal{E}_A} \mathcal{D}\left(\mathcal{H}_S^A\right) \xrightarrow{\mathcal{E}_B^{\otimes n}} \mathcal{D}\Big(\left(\mathcal{H}_S^B\right)^{\otimes n}\Big)$$

where $\mathcal{E}_B : \mathcal{D}(\mathcal{H}_L^B) \to \mathcal{D}(\mathcal{H}_S^B)$ and $\mathcal{E}_A : \mathcal{D}(\mathcal{H}_L^A) \to \mathcal{D}(\mathcal{H}_S^A)$ are the two encoding maps of $\mathcal{C}_A$ and $\mathcal{C}_B$, respectively.

If for example $\mathcal{C}_A$ is the three-qubit bit-flip code and $\mathcal{C}_B$ is the three-qubit phase-flip code, then the concatenation with encoding $\mathcal{E}_B^{\otimes n} \circ \mathcal{E}_A$ encodes one logical qubit into nine qubits and is designed to protect from both bit flips and phase flips.

Let us describe *one example of a recovery* of the above defined concatenated code $\mathcal{C}_{AB}$ in more detail, assuming that the two constituent codes have recoveries $\mathcal{R}_A$ and $\mathcal{R}_B$. Fig. 3.4 shows the circuit associated with this error correction procedure of a concatenated code.

It is usually reasonable to assume that the concatenated code $\mathcal{C}_{AB}$ undergoes the physical noise channel

$$\mathcal{N}_{AB} \coloneqq \mathcal{N}_B^{\otimes n} : \mathcal{D}\Big(\left(\mathcal{H}_S^B\right)^{\otimes n}\Big) \to \mathcal{D}\Big(\left(\mathcal{H}_S^B\right)^{\otimes n}\Big)$$

which corresponds to the n-fold tensor product of a physical noise channel $\mathcal{N}_B$ of the inner code $\mathcal{C}_B$. We will start by considering the n-fold tensor product of the inner code $\mathcal{C}_B$. If $\mathcal{R}_B$ is a valid recovery for the noise $\mathcal{N}_B$, then $\left(\mathcal{E}_B^\dagger\right)^{\otimes n} \circ \mathcal{R}_B^{\otimes n}$ is well-defined on corrupted states $\mathcal{N}_B^{\otimes n}(\rho)$

for ρ supported on $\mathcal{C}_B^{\otimes n}$. Hence, the $\mathcal{R}_B$-recovered states are supported on $\mathcal{C}_B^{\otimes n}$ and one can apply the n-fold inverse encoding map $\left(\mathcal{E}_B^\dagger\right)^{\otimes n}$ to them. Define a decoder of the concatenated code $\mathcal{C}_{AB}$ as

$$\mathcal{E}_{AB}^\dagger \circ \mathcal{R}_{AB} \coloneqq \mathcal{E}_A^\dagger \circ \mathcal{R}_A \circ \left(\mathcal{E}_B^\dagger\right)^{\otimes n} \circ \mathcal{R}_B^{\otimes n} : \mathcal{D}\left(\mathcal{H}_L^A\right) \to \mathcal{D}\left(\left(\mathcal{H}_S^B\right)^{\otimes n}\right) .$$

To investigate under which conditions this defines a valid recovery (in the sense of Definition 3.15), the recovered state must be supported on $\mathcal{C}_A$. This is the case if the recovery $\mathcal{R}_A$ is valid for noise channel

$$\mathcal{N}_A = (\mathcal{E}_B^\dagger)^{\otimes n} \circ \mathcal{R}_B^{\otimes n} \circ \mathcal{N}_B^{\otimes n} \circ \mathcal{E}_B^{\otimes n}$$

which is the n-fold tensor product of the logical noise channel $\overline{\mathcal{N}}_B \coloneqq \mathcal{E}_B^\dagger \circ \mathcal{R}_B \circ \mathcal{N}_B \circ \mathcal{E}_B$ (cf. Eq. (3.39)) of the inner code $\mathcal{C}_B$. Therefore the recovery channel $\mathcal{R}_{AB}$ of the concatenated code is defined as

$$\mathcal{R}_{AB} \coloneqq \mathcal{E}_B^{\otimes n} \circ \mathcal{R}_A \circ \left(\mathcal{E}_B^\dagger\right)^{\otimes n} \circ \mathcal{R}_B^{\otimes n} : \mathcal{D}\left(\left(\mathcal{H}_S^B\right)^{\otimes n}\right) \to \mathcal{D}\left(\left(\mathcal{H}_S^B\right)^{\otimes n}\right) .$$

By definition, it is successful if $\mathcal{R}_{AB} \circ \mathcal{N}_{AB}(\rho) \propto \rho$ for all states ρ supported on $\mathcal{C}_{AB}$.

3.4.5 Topological codes

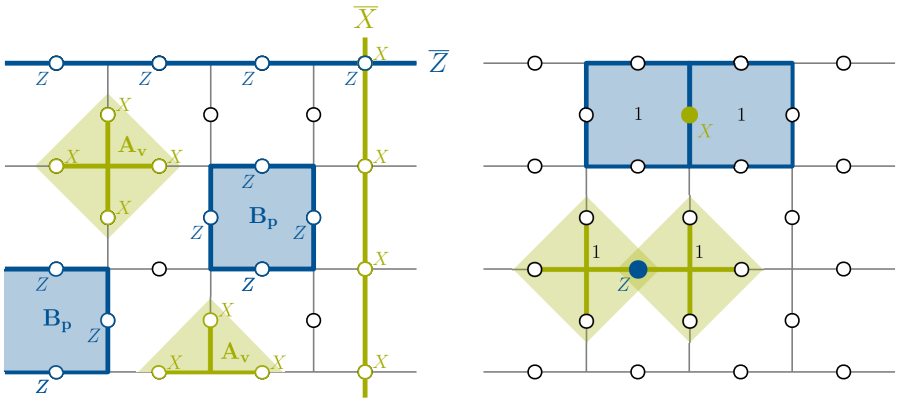
Let us consider an example for a family of stabiliser codes which is of wide interest, so called topological codes [30, 92]. They encode $k \in \mathbb{N}$ logical qubits into $n \in \mathbb{N}$ physical qubits where the latter are arranged on a lattice on some N-dimensional surface. The stabiliser generators are chosen such that different logical operations correspond to topologically distinct closed loops of lattice edges (more precisely, different non-trivial so-called homology cycles) and hence the topology of this surface determines the quantum error correction properties of the associated code. Probably the most famous topological code, the *toric code*, was introduced in 1997 by Kitaev [91]. Here, the physical qubits are arranged on a torus (i.e., a two-dimensional lattice with periodic boundary conditions) constituting a non-trivial topology in two dimensions. The generalisation to planar lattices with different boundary conditions gives rise to a new family of quantum error correcting codes, the surface codes [30, 67, 47, 92].

Surface codes are among the most promising architectures for a realistic quantum computer; see [16, 40, 87, 156] for experiments in this direction.

The square surface code

In the following, we exemplarily discuss a specific variant of surface codes, namely the square surface code with so-called 'rough' left/right boundary conditions and 'smooth' top/bottom boundary conditions. We choose this example of a surface code since we use this code as part of the surface-GKP code in Chapter 5.

For its construction, consider a square lattice in two dimensions of $d \times d$ vertices for some $d \in \mathbb{N}$. Then identify the left and right boundaries of the lattice, i.e., fold the lattice on a cylinder. Cutting open the cylinder vertically (such that the cut intersects horizontal edges but does not touch vertical edges or vertices) and unfolding results in a lattice that has d^2 horizontal edges and $(d-1)^2$ vertical edges. One associates a physical qubit with every edge of this lattice, resulting in $d^2 + (d-1)^2$ physical qubits. Let us denote by $v \in \mathsf{V}$ the vertices of the lattice, by $e \in \mathsf{E}$ its edges and by $p \in \mathsf{P}$ its regions, which we call plaquettes. Sometimes,



(a) Stabilisers and logical operators (b) Examples of single-qubit errors

Figure 3.5: Distance $d = 4$ square surface code with 25 physical qubits (each represented by a circle) on the edges of a lattice. This lattice has horizontal edges both at the top/bottom boundaries as well as at the left/right boundaries. (a) Examples of stabiliser generators which are either vertex stabilisers A_v (in green) or plaquette stabilisers B_p (blue). Logical $\overline{Z}$ ($\overline{X}$) operators are defined as horizontal (vertical) strings at the top (right) boundary along the (dual) lattice. (b) A single-qubit Z-error (blue circle) is detected by the two adjacent vertex stabiliser generators as they give syndrome 1 whereas all other stabiliser generators give syndrome 0. Similarly, a single-qubit X-error (green circle) causes syndromes 1 on the two neighbouring plaquette stabilisers.

it might be useful to distinguish the horizontal edges (denoting them as h-edges) and the vertical edges (denoted as v-edges). Furthermore, let δv and δp denote the set of edges adjacent to a given vertex v and a plaquette p, respectively.

Generally, the term rough or plaquette-type boundary is used when the left/right boundary edges are horizontal, and the top/bottom edges are vertical, whereas we speak of smooth or vertex-type boundaries for vertical edges at the left/right boundary and horizontal edges on the top/bottom boundary.

The so-called *vertex and plaquette operators* are defined as

$$A_v := \prod_{e \in \delta v} X_e \ , \qquad \text{and} \qquad B_p := \prod_{e \in \delta p} Z_e \ , \tag{3.47}$$

for $v \in \mathsf{V}$, $p \in \mathsf{P}$, cf. Fig. 3.5. One can easily check that they all commute. Hence,

$$\mathcal{S} := \langle A_v, B_p \mid v \in \mathsf{V}, p \in \mathsf{P} \rangle \tag{3.48}$$

forms a stabiliser group. The surface code is defined as the stabiliser code $\mathcal{C}_\mathcal{S}$ of (3.48). With $|\mathsf{V}| + |\mathsf{P}|$ independent stabiliser generators, the number of encoded logical qubits is computed to be $n - |\mathsf{V}| - |\mathsf{P}| = d^2 + (d-1)^2 - d(d-1) - d(d-1) = 1$.

The logical operations are elements of the centraliser $Z(\mathcal{S})$ of $\mathcal{S}$ inside the Pauli group, cf. (3.42). More precisely, the quotient group $Z(\mathcal{S})/\mathcal{S}$ is equal to the Pauli group $\mathcal{P}_1$ of the logical encoded qubit. One defines the logical X-operator, denoted by $\overline{X}$, and the logical Z-operator as the tensor products of X (Z) operators along the right line (top line) of horizontal edges of the square lattice, respectively. We note that they anticommute and that they commute with the stabiliser generators (3.47). The logical Y is defined as $\overline{Y} := i\overline{X} \cdot \overline{Z}$. The weight of both $\overline{X}$ and $\overline{Z}$ is d. As these are minimal-weight logical operators, the code distance is d.

Let us compute the computational code space basis $|\overline{0}\rangle, |\overline{1}\rangle$. Its two elements are in the joint $+1$ eigenspace of the stabilisers A_v and B_p, the logical 0-state is furthermore also invariant under $\overline{Z}$. It can be computed by first preparing all physical qubits in the 0-state and then subsequently applying all stabilisers, i.e., it is

$$|\overline{0}\rangle_{\text{surface}} = \frac{1}{\sqrt{2^{d(d-1)}}} \prod_{v\in\mathsf{V}} \left(I^{\otimes(d^2+(d-1)^2)} + A_v\right) |0\rangle^{\otimes(d^2+(d-1)^2)} \ .$$

The logical one $|\overline{1}\rangle_{\text{surface}}$ is defined similarly.

The syndrome measurement

We consider the standard stabiliser code error correction procedure: The encoded ideal state $\rho \in \mathcal{D}(\mathcal{H}_n)$ supported on the code space $\mathcal{C}$ undergoes a noise channel $\mathcal{N} : \mathcal{D}(\mathcal{H}_n) \to \mathcal{D}(\mathcal{H}_n)$. Subsequently, the stabiliser generators (3.47) are measured. The measurement outcome is a syndrome

$$s = (\{s_v\}_{v\in\mathsf{V}}, \{s_p\}_{p\in\mathsf{P}}) \in \{0,1\}^{|\mathsf{V}|} \times \{0,1\}^{|\mathsf{P}|}$$

occurring with probability $p(s) \coloneqq \mathrm{tr}(\Pi(s)\mathcal{N}(\rho))$ where

$$\Pi(s) \coloneqq \prod_{p\in\mathsf{P}} \frac{1}{2}\left(I + (-1)^{s_p} B_p\right) \prod_{v\in\mathsf{V}} \frac{1}{2}\left(I + (-1)^{s_v} A_v\right) \ .$$

The syndromes detect errors in the following sense. In the tuple s one associates a binary value $\{0,1\}$ with every stabiliser generator, where for an X-type stabiliser A_v (or a Z-type stabiliser B_p) the value 1 symbolises that a Y- or Z-type error (or an X- or Y-type error, respectively) has corrupted one of its neighbouring (physical) qubits. Two examples of this error detection are given in Fig. 3.5(b).

Surface code decoders

As for all stabiliser codes, a *decoding strategy* amounts to the choice of a function $s \mapsto C(s) \in \mathcal{P}_n$ that associates a correction operation with every syndrome. The choice of the optimal decoder (i.e., the one giving the highest success probability) depends on the noise model.

The noise we consider is stochastic Pauli noise on the n physical qubits (cf. Definition 3.2). Recall the decomposition of the coset

$$E_s Z(\mathcal{S}) \coloneqq E_s\mathcal{S} \cup E_s\overline{X}\mathcal{S} \cup E_s\overline{Y}\mathcal{S} \cup E_s\overline{Z}\mathcal{S} \tag{3.49}$$

from Eq. (3.43) where for every syndrome s we fixed a representative error E_s causing s. Let $\mathcal{C}_{\text{dec}}(s)$ be the subset of the right hand side of (3.49) to which $C(s)$ belongs. The average success probability of a decoder $s \mapsto C(s)$ is given by Eq. (3.46), i.e., it is

$$P_{\mathsf{success}} = \sum_E \mathrm{Prob}[E \in \mathcal{C}_{\text{dec}}(s)] = \sum_{E\in\mathcal{C}_{\text{dec}}(s)} \pi(E) \ ,$$

where for a syndrome s, $\mathcal{C}_{\text{dec}}(s)$ is the subset of the right hand side of (3.49) to which $C(s)$ belongs and $\pi : \mathcal{P}_n \to [0,1]$ is the probability distribution of the errors (cf. Definition 3.2).

For the *maximum likelihood* (ML) decoder $s \mapsto C_{\mathrm{ML}}(s)$, the correction is chosen as

$$\mathcal{C}_{\mathrm{ML}}(s) = \arg \max_{\mathcal{A} \in \{E_s\mathcal{S}, E_s\overline{X}\mathcal{S}, E_s\overline{Y}\mathcal{S}, E_s\overline{Z}\mathcal{S}\}} \pi(\mathcal{A}) \ , \tag{3.50}$$

where ties are broken in an arbitrary manner. This decoder maximises the average success probability over all decoders. In practice, it is usually very costly to compute the coset probabilities

$$\pi(E_s\mathcal{S}) \quad , \quad \pi(E_s\overline{X}\mathcal{S}) \quad , \quad \pi(E_s\overline{Y}\mathcal{S}) \quad , \quad \pi(E_s\overline{Z}\mathcal{S}) \tag{3.51}$$

from Eq. (3.50). More precisely, the computation of the above coset probabilities usually involves sums over sets of exponential size. One may search for a decoder which is an efficiently computable approximation of the ML decoder (cf. Section 5.3.3 on the BSV decoder).

We note that there are also other prominent decoders such as the minimum weight perfect matching decoder, cf. e.g. the review by Fowler et al. [66] for a detailed description of the use of a minimum weight perfect matching [55] algorithm for surface code error correction. Moreover, there are also decoders which are tailored towards specific physically realistic types of noise.

Error threshold in topological codes

A remarkable feature of toric and surface codes is that by increasing their distance d – i.e., the size of the surface/toric code lattice and consequently the number of physical qubits – the success probability of the error correction procedure grows (up to one) as long as the physical error rate lies beyond a threshold value.

Let us illustrate this threshold behaviour: To parametrise the noise model by a single parameter – the physical error rate p – one makes additional assumptions on the noise model, e.g. one restricts to i.i.d. depolarising noise on all physical qubits, i.e., noise is of the form of Eqs. (3.6) and (3.8). For now, one assumes that noise is only introduced as memory errors (characterised by the error rate p in the i.i.d. depolarsing noise) and that the error-correction process itself is noiseless (in contrast to the considerations given in the next section). Then the threshold theorem implies that $\lim_{d\to\infty} P_{\mathsf{success}}(p) \to 1$ if $p < p_{\mathrm{crit}}$. As long as $p < p_{\mathrm{crit}}$, increasing the code size improves its noise tolerance such that in the limit of an infinite code the resulting success probability of the code is equal to one. There exist different strategies to prove this result and to compute the actual threshold value p_{crit} for different versions of the toric or the surface code.

For the toric code, simple combinatorial considerations imply an upper bound on the logical failure probability of the code and thereby one obtains a lower bound for the error threshold of approximately $p_{\mathrm{crit}} \geq 0.027$ (cf. Preskill's lecture notes [134, handwritten notes on "Toric code recovery, fault-tolerant recovery, fault-tolerant gates"]) or $p_{\mathrm{crit}} \geq 0.037$ [47]. This method uses the fact that every stabiliser generator of the surface code is a product of either X or Z operators (on up to four qubits) and therefore the code belongs to the class of so-called Calderbank-Shor-Steane (CSS) codes [147, 32, 148]. We note that this method involves rough estimates and thereby only gives a lower bound on the actual error threshold. More precisely, numerical simulations indicate that the error threshold is in fact much higher. By simulating the toric code error correction procedure, Wang et al. [171] numerically estimated $p_{\mathrm{crit}} \approx 0.155 \pm 0.05$ for depolarising noise, cf. Eq. (3.8) which corresponds to the error threshold $\frac{2}{3}0.155 = 0.103$ for pure X-errors. Similarly for the surface code, Bravyi, Suchara and Vargo [29] numerically found $0.17 \leq p_{\mathrm{crit}} \leq 0.185$ for depolarising noise

and $0.109 \leq p_{\text{crit}} \leq 0.11$ for pure X- or pure Z-noise. The next section explains that this behaviour extends to the case when the syndrome measurements are noisy as well.

Note that these above threshold results assume *memory-only noise*. For error thresholds of *circuit-based noise*, we refer to the next paragraph.

3.4.6 Fault tolerance and threshold theorem

The QECC protocols discussed so far in this chapter only consider the corruption of the encoded states. But in a realistic quantum computer, every part of the computation might be noisy: errors may also effect the state preparation, the gates used for encoding or decoding as well as the auxiliary qubits and gates used for the syndrome measurements, the gates of the computation itself and the measurements to read out the result of the computation: in summary, all gates and all qubits can be noisy. Of course, this poses additional obstacles on QECC. To reliably use a quantum computer, such errors must be prevented from propagating uncontrollably. Protocols that take such errors into account and are able to combat them (in some asymptotic way) are called *fault-tolerant*.

In principle, it is possible that the error correction might not be successful when errors accumulate, the gates and auxiliary qubits are noisy as well, i.e., quantum computers with imperfect gates and qubits might necessarily fail after a constant number of gate operations due to noise. But the *quantum threshold theorem* tells us that – under certain additional, but reasonable assumptions on the underlying hardware and noise – the opposite is the case. If the physical error rate per gate and per time step is low enough – i.e., it lies below a constant threshold value – reliable quantum computation is possible in the following approximate sense: the actual noisy computation becomes arbitrarily close to the desired (perfect) one with overhead which is polylogarithmic in the length of the computation. To do this, one has to encode the whole computation: logical qubits are replaced by encoded qubits, logical gates by encoded gates and logical measurements by encoded measurements. Historically, Shor pioneered the study of a recursive re-encoding for fault-tolerant quantum computation [144]. Subsequently, the threshold theorem was shown to hold for a series of different quantum codes and noise models, independently proven by several authors [2, 3, 97, 99, 98, 91, 8].

There are basically two approaches to construct families of codes that achieve a fault tolerance error threshold. First one can use concatenations (cf. Section 3.4.4) of QECC, where the underlying idea is as follows: the underlying idea is the following: If the encoded computation is less noisy than the unprotected one, then recursively re-encoding the encoded circuit into itself yields the desired error threshold [8, 38]; see also the review article by Terhal [157]. The value of the error threshold depends highly on the assumptions, e.g. Knill [94] argued that the threshold lies above $p_{\text{crit}} \approx 0.03$ compared to the estimate of $\approx 10^{-5}$ by Stephens and Evans [151].

The second approach uses topological quantum codes [47, 170, 137]. More precisely, the threshold behaviour for storage errors and with perfect syndrome measurements (as described in the above Section 3.4.5) extends to the case of imperfect syndromes. The key insight to prove this result is to establish a connection between the surface code and the random bond Ising model [47]. The latter statistical physics model is parametrised by two parameters in thermal equilibrium: the inverse temperature β and the probability p of individual spins being antiferromagnetic. In the (β, p)-plane, this model admits a phase transition between an ordered phase for low temperatures and low p and a disordered phase at high temperatures and high p. Furthermore, the order can also be destroyed – even at zero temperature – by

increasing p above a critical value p_{crit}. This critical values lies where the line separating the ordered from the disordered phase in the (β, p)-plane (the phase boundary) crosses the so-called Nishimori line [124] $e^{-2\beta} = p(1-p)^{-1}$. By suitably mapping the Ising model to the lattice of a toric code [47] one finds that the accuracy threshold is determined by the critical value of p on the Nishimori line. For example for perfect syndrome measurements, the error threshold corresponds to the zero temperature crossing point – the latter value has been numerically computed to be $p_{\text{crit}} \approx 0.1094 \pm 0.0002$ by Honecker et al. [84]. This principle (as well we as the proof) extends to the case when one allows the syndrome measurement to be noisy as well. For noisy syndromes, this fault-tolerance threshold value has been estimated to vary between ≈ 0.0078 [171] and ≈ 0.0075 [137] under different assumptions on the decoder for pure dephasing noise. Such results are obtained numerically by simulating the error correction process of larger and larger (in terms of distance) surface codes on a classical computer; a review article computes the value $p_{\text{crit}} \approx 0.0057$ [66]. In both the noisy as well as the non-noisy syndrome case, there exists a threshold value for the single-qubit errors in the following sense: As long as $p < p_{\text{crit}}$ increasing the code size improves the success probability, and if $p > p_{\text{crit}}$, increasing the code size decreases the success probability of error correction. As a consequence, information processing in such codes can be implemented fault-tolerantly.

3.5 Continuous variable quantum error correcting codes

Let us consider quantum codes in the bosonic setting. Here, the physical system is infinite-dimensional, more precisely $\mathcal{H}_S = L^2(\mathbb{R}^n)$ for $n \in \mathbb{N}$. Generalising Definition 3.14 of a QECC to the CV setting, a quantum code $\mathcal{C}$ is defined by a subspace of this n-mode Hilbert space $\mathcal{H}_S$. One can then distinguish two fundamentally distinct cases:

(i) The code space $\mathcal{C}$ is infinite-dimensional, i.e., we encode an infinite-dimensional system. This is called a *CV-into-CV encoding*.

(ii) The code space $\mathcal{C}$ is finite-dimensional, i.e., we encode a finite-dimensional or discrete variable (DV) system into the bosonic system. Such codes are called *DV-into-CV encodings*.

In both cases, the physical Hilbert space $\mathcal{H}_S$ is infinite-dimensional. Whereas in the finite-dimensional multi-qubit case, the Pauli group $\mathcal{P}_n$ forms a finite basis of the error set on $(\mathbb{C}^2)^{\otimes n}$, this is no longer true in the CV setting. Recall for example that in the CV setup, the Heisenberg-Weyl group H_n from Eq. (3.10) which forms an analogous error basis contains uncountably many displacement operators (characterised by continuous variables).

In the first case, one harmonic oscillator mode is encoded into many harmonic oscillator modes. Such a code is potentially well-suited for CV quantum information processing tasks since the encoded system is itself a CV system associated with an infinite-dimensional Hilbert space. Proposals for such CV-into-CV encodings include bosonic analogues of the three, five and nine qubit codes [27, 110, 28] as well as more advanced schemes [10, 80, 79, 60, 179, 125]. The difficulty of such schemes is that there are fundamental reasons which prohibit their success when using only Gaussian encodings: even Gaussian errors cannot be corrected by Gaussian resources only. Different formulations of these no-go results include [169, 123, 69, 56].

The second approach may be practically more advantageous since it is generally easier to protect a finite number of degrees of freedom than an infinite one. Although the encoded

DV system is not suited for CV quantum information processing tasks – after all, it is finite-dimensional – this approach might be more powerful than a standard DV-into-DV encoding and be applied to physical systems which are inherently continuous. Usually, such DV-into-CV encodings are designed to protect the encoded information against a typical noise model. Let us mention the most prominent examples of DV-into-CV encodings that are promising candidates fo experimental realisations. Gottesman, Kitaev and Preskill designed a family of CV stabiliser codes [71], which are named *GKP codes* after their inventors. These codes protect encoded qubit or qudit states – which are superpositions of position and momentum eigenstates and hence involve an infinite amount of squeezing – against small Weyl displacement errors, i.e., against Gaussian noise of the form of Eq. (3.11). A different proposal considers so-called *cat code states* [39, 105, 115] which are superpositions of coherent states, more precisely, the equally weighted superpositions of peaked distributions at equidistant points on a circle in phase space. Depending on a parameter, they protect against small loss errors and backaction errors. *Binomial codes* [114] by Michael et al. were designed to protect against the error set $\mathfrak{E} = \{a, a^2, \ldots, a^L, a^\dagger, \ldots, (a^\dagger)^G, (a^\dagger a), \ldots, (a^\dagger a)^N\}$ for $L, G, N \in \mathbb{N}$, i.e., against photon gain and losses up to a fixed number. Here, a basis of the code space is given by superpositions of Fock states with binomial coefficients. A numerical comparison between these three proposals – where a single-qubit is encoded into a CV system – is given in [4], showing that under most scenarios the GKP code performs best. Experimentally, GKP states are difficult to prepare: a preparation proposal [164] uses an atomic ensemble in a spin-coherent state that is coupled to a squeezed state of light, a more recent one is build on circuit QED [142]. Approximate GKP states have recently been realised using superconducting qubits (circuit QED) [33] with a squeezing of 9.5dB and ion traps [65, 46]. In contrast, cat code error correction has already been performed [126] using superconducting qubits.

4 Higher order dynamical decoupling of continuous variable systems

This chapter includes results which have been published in the article *"Universal Uhrig dynamical decoupling for bosonic systems"* [82] by the author of this thesis and Robert König.

More precisely, this concerns the results in Sections 4.5, 4.6, 4.7, 4.9 and parts of 4.8. The (proof) ideas of this work were developed in joint discussions between the two authors (Robert König and Margret Heinze). Margret Heinze mainly conducted the computations and proofs, i.e., those for Sections 4.5, 4.6.3 4.7, 4.8 and 4.9.1. The two proofs of higher order decoherence suppression in 4.6.2 and for the linear terms 4.9.2 were developed and computed jointly between the two co-authors. We remark that Sections 4.2, 4.8.1 and 4.8.3 contain new concepts, examples, and results that have not been published.

The idea

Although dynamical decoupling (DD) schemes in finite-dimensional systems are established tools to noise reduction and have been theoretically extensively studied, there is only little literature on DD for infinite-dimensional systems. A first publication in this direction analysed a specific system-environment model [168]. Inspired by this work, Arenz, Burgarth and Hillier pioneered in 2017 the systematic study of DD of infinite-dimensional systems [12]. They show that in the CV setting, universal dynamical decoupling is not possible in the same strong sense as in the finite-dimensional case. The reason for this is essentially the non-compactness of the associated unitary groups in the infinite-dimensional case. More precisely, no pulse sequence of Gaussian unitary pulses can render the resulting evolution – i.e., the evolution generated by the decoherence Hamiltonian between the system and the environment interleaved with the control pulses – act trivially on the system for an arbitrary decoherence Hamiltonian. However, one may be able to achieve a slightly more modest goal: for all such quadratic decoherence Hamiltonians it is possible to approximately render the resulting evolution to be in a tensor product form between system and environment, a process referred to as *decoherence suppression*. And moreover, the system part of this effective evolution – although not trivial – may be approximately reduced to that of free harmonic oscillators rotating at the *same* frequency, a process called *homogenisation*.

In this sense, there is a crucial difference between DD in the finite-dimensional case and DD of continuous variable systems. However, in this chapter, we show that both settings also share several useful properties. More precisely, we investigate and exploit these similarities to construct novel and highly efficient pulse sequences in the bosonic setting from multi-qubit DD schemes. We note that the decoherence suppression and homogenisation pulse sequences constructed in [12] use equidistant pulse timings and homogenise the system evolution at the

level of symplectic matrices up to first order in the total sequence time. Our schemes use non-equidistant pulses and are able to achieve any desired suppression order. As the main result of this chapter, we show that the most efficient higher order finite-dimensional DD schemes, based on the so-called Uhrig DD [161] scheme, can be translated to the CV setting. The constructed CV Uhrig decoherence suppression scheme, discussed in Section 4.6, and the CV nested Uhrig homogenisation scheme, cf. Section 4.8, are remarkably simple, involving only passive Gaussian unitaries as pulses. Yet, they are highly efficient, using only a polynomial number (in the suppression order) of passive Gaussian unitary pulses.

Organisation of this chapter

- Section 4.1 recaps previous work on DD for CV systems by Arenz, Burgarth and Hillier [12]: their impossibility result on universal first order DD in the CV setting and the concepts of first order decoherence suppression and homogenisation.
- In Section 4.2, we show how to construct first order CV homogenisation schemes from unitary 1-designs, similarly to the construction of 1-design-DD sequences in Lemma 3.13 for the finite-dimensional setting.
- Section 4.3 introduces the definitions necessary to prove the two main results of this chapter: we present the noise model and define *higher order* decoherence suppression and homogenisation sequences. Here, we extend the definitions of Arenz, Burgarth and Hillier to capture more general pulse timings and higher suppression orders.
- Section 4.4 is an excursion to the finite-dimensional case. We summarise prior results on higher order DD on multi-qubit systems, especially the two schemes which are directly relevant to our constructions: the Uhrig DD and nested Uhrig DD schemes. These are the most efficient (multi-)qubit DD schemes up to date and hence serve as a starting point for the novel CV schemes introduced in this thesis.
- Section 4.5 presents a unifying framework for DD which captures both multi-qubit DD as well as bosonic homogenisation. This idea builds on previous results by Jiang and Imambekov [86] on the qubit case and extends these results to the CV setting by analysing the DD properties at the levels of the associated Lie groups and algebra.
- Section 4.6 presents the **first central result** of this chapter: We introduce an efficient higher order bosonic decoherence suppression scheme which is deduced from UDD for a single qubit. We furthermore analyse optimality of the pulse timings in the CV setting, establishing similar optimality results of the Uhrig DD times as in the qubit setting.
- Section 4.7 relates the Lie algebra and Lie group of $m+1$-qubit systems to those associated with 2^m bosonic modes. This construction is not limited to bosonic DD but it is the key ingredient of the bosonic homogenisation schemes proposed subsequently.
- Section 4.8 presents the **second central result** of this chapter: the CV nested Uhrig homogenisation schemes. More precisely, we first introduce a translation of multi-qubit DD schemes into bosonic homogenisation schemes and then apply this to the multi-qubit NUDD scheme.
- Section 4.9 discusses two further aspects of the CV decoherence suppression and homogenisation. We characterise sufficiently low decoupling rates and analyse the performance of the proposed schemes in the presence of decoherence Hamiltonians with terms that are *linear* in the mode operators.

4.1 Prior work on bosonic DD by Arenz, Burgarth and Hillier

Since we later work on the basis of their definitions and extend their findings, let us summarise the prior work by Arenz, Burgarth and Hillier [12] on CV DD in this section.

4.1.1 Impossibility of universal DD in infinite dimensions

The authors of [12] establish a no-go result for DD in the infinite-dimensional setting. Using a simple rank argument they show that for an infinite-dimensional Hilbert space $\dim \mathcal{H}_S = \infty$, there is no DD set (or DD group) in the sense of Definition 3.10, i.e., there is no finite group $\mathcal{G} = \{g_j\}_{j=1}^{|\mathcal{G}|}$ of unitaries on $\mathcal{H}_S$ such that

$$\frac{1}{|\mathcal{G}|}\sum_{j=1}^{|\mathcal{G}|} U_{g_j}^\dagger M U_{g_j} = \lambda_M I_S \qquad \text{for all } M \in \mathcal{B}(\mathcal{H}_S), \tag{4.1}$$

where $\lambda_M \in \mathbb{C}$ depends on M.

Let me add two remarks on this result: First, condition (4.1) is a statement on *bounded* operators on the system $M \in \mathcal{B}(\mathcal{H}_S)$. Many Hamiltonians on CV systems contain unbounded operators such as the mode operators Q and P. Therefore, the no-go-result implies that even in the restricted setting of (decoupling of) bounded operators, DD is not possible in the same (strong) sense as for finite-dimensional systems. Second, this result does not imply that a group of unitaries may not average out a specific decoherence Hamiltonian. It is still be possible that there exist *subsets* $\mathfrak{H} \subset \mathcal{B}(\mathcal{H}_S)$ such that Eq. (4.1) is satisfied for every $M \in \mathfrak{H}$. The no-go result only implies that there is no *universal* DD set for infinite-dimensional systems. Note that in later work by the same authors together with Facchi [11] this result is made mathematically more precise by analysing the domain of the (unbounded) Hamiltonian.

This is why the remainder of the article [12] makes two relaxations for defining DD in the infinite-dimensional setting: First, the right hand side of Eq. (4.1) is relaxed to a weaker condition referred to as homogenisation (cf. Definition 4.3). Second, the set of considered decoherence Hamiltonians is restricted to those which are quadratic in the quadrature operators.

To make the second assumption precise, let decoherence on a system of $n_S \in \mathbb{N}$ modes, i.e., $\mathcal{H}_S = L^2(\mathbb{R}^{n_S})$, be introduced by a Hamiltonian of the form

$$H^{\text{orig}} = \frac{1}{2}\sum_{j,k=1}^{2n_S+2n_E} A_{jk} R_j R_k \ . \tag{4.2}$$

Here, E denotes an environment of n_E modes, i.e., $\mathcal{H}_E = L^2(\mathbb{R}^{n_E})$, R the vector of quadrature operators and $A \in \mathbb{R}^{(2n_S+2n_E)\times(2n_S+2n_E)}$ is a symmetric matrix. It is furthermore assumed that the DD pulses are Gaussian unitaries. In this restricted setup[1] – decoherence and pulses are Gaussian unitaries – the whole analysis can be conducted at the level of symplectic matrices $S \in \mathsf{Sp}(2n_S+2n_E, \mathbb{R})$ on phase space $\mathbb{R}^{2n_S+2n_E}$. This simplifies the mathematics massively

[1] Recall from Section 2.2.4 that a Gaussian unitary of the form $U = e^{-\frac{i}{2}R^T A R}$ is in one-to-one correspondence with the symplectic matrix $S = e^{JA}$ via the relation $U^\dagger R_j U = \sum_k S_{jk} R_k$.

since one deals with (symplectic) matrices instead of potentially unbounded operators. As a consequence,

- instead of the original Hamiltonian (4.2), one considers the matrix $JA \in \mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$,
- instead of the (Gaussian) unitary control pulses $U_1, U_2, \ldots, U_{|\mathcal{G}|} \in \mathsf{U}(\mathcal{H}_S)$ applied at equidistant times $T/|\mathcal{G}|, 2T/|\mathcal{G}|, \ldots, T$ one considers the associated (according to the metaplectic representation, cf. Section 2.2.4) symplectic control pulses $S_1, \ldots, S_{|\mathcal{G}|} \in \mathsf{Sp}(2n_S, \mathbb{R})$ applied at the same times,
- instead of the resulting unitary evolution $U^{\mathrm{res}}(T)$ one analyses the resulting symplectic evolution

$$S^{\mathrm{res}}(T) = \left(S_{|\mathcal{G}|} \oplus I_{2n_E}\right) e^{\frac{T}{|\mathcal{G}|}JA} \cdots \left(S_2 \oplus I_{2n_E}\right) e^{\frac{T}{|\mathcal{G}|}JA} \left(S_1 \oplus I_{2n_E}\right) e^{\frac{T}{|\mathcal{G}|}JA} \ .$$

Note that we write

$$B \oplus C = \begin{pmatrix} B & 0 \\ 0 & C \end{pmatrix}$$

for two matrices $B \in \mathbb{R}^{2n_S \otimes 2n_S}$ and $C \in \mathbb{R}^{2n_E \otimes 2n_E}$. Furthermore, in the right basis when ordering the quadratures as

$$R^T = (Q_1 \ldots Q_{n_S} \ P_1 \ldots P_{n_S} \ Q_{n_S+1} \ldots Q_{n_S+n_E} \ P_{n_S+1} \ldots P_{n_S+n_E}) \ , \tag{4.3}$$

a tensor product of unitary operators corresponds on the symplectic level to a direct sum of associated symplectic matrices, e.g. $U_k \otimes I_E$ corresponds to $S_k \oplus I_{2n_E}$.

The considered framework is similar to that of *DD sets* (cf. Definition 3.10 from Section 3.3.3 in the finite-dimensional case): For a finite set $\mathcal{G} = \{g_j\}_{j=0}^{|\mathcal{G}|-1} \subset \mathsf{Sp}(2n_S, \mathbb{R})$ of symplectic matrices one analyses – in the limit of infinitely fast pulse application – the resulting evolution after application of the pulses $S_j = g_j g_{j-1}^{-1}$ to the system at equidistant times $jT/|\mathcal{G}|$ for $j = 1, \ldots, |\mathcal{G}|$. The authors show that under these assumptions it is possible to suppress decoherence and to homogenise the system evolution, two processes that we describe in the following.

4.1.2 Decoherence suppression

Definition 4.1 *(Decoherence suppression).* A bosonic pulse sequence achieves *decoherence suppression of system-environment interactions* after time t in the limit of infinitely fast pulse application if there are two symplectic matrices $S_S \in \mathsf{Sp}(2n_S, \mathbb{R})$ and $S_E \in \mathsf{Sp}(2n_E, \mathbb{R})$ such that

$$\lim_{m\to\infty} \left(S^{\mathrm{res}}(t/m)\right)^m = S_S \oplus S_E \ . \tag{4.4}$$

When ordering the quadratures in the vector R as in Eq. (4.3), a resulting symplectic evolution of the form $S^{\mathrm{res}}(t) = S_S \oplus S_E$ for some $S_S \in \mathsf{Sp}(2n_S, \mathbb{R})$, $S_E \in \mathsf{Sp}(2n_E, \mathbb{R})$ corresponds to a unitary evolution in tensor product form $U^{\mathrm{res}}(t) = U_S \otimes U_E$ for $U_S \in \mathsf{U}(L^2(\mathbb{R}^{n_S}))$ and $U_E \in \mathsf{U}(L^2(\mathbb{R}^{n_E}))$. The above definition, Eq. (4.4) does not imply that S_S acts trivially on the system (or respectively $U_S = I_S$). In this sense it is weaker condition than DD and corresponds to property (i) for finite-dimensional systems.

The authors derive a condition on a pulse sequence such that Eq. (4.4) holds. Let $\mathcal{G} = \{g_j\}_{j=0}^{|\mathcal{G}|-1} \subset \mathsf{Sp}(2n_S, \mathbb{R})$ be a finite subgroup and consider the pulse sequence of pulses

$$(g_j \oplus I_{2n_E})(g_{j-1} \oplus I_{2n_E})^{-1} \quad \text{at times} \quad Tj/|\mathcal{G}| \quad \text{for} \quad j = 1, \ldots, |\mathcal{G}| \; . \tag{4.5}$$

At time t after $m \in \mathbb{N}$ applications of the cycle one finds

$$(S^{\text{res}}(t/m))^m = \left(\prod_{k=1}^{|\mathcal{G}|} e^{\frac{T}{m|\mathcal{G}|}(g_k \oplus I_{2n_E})^{-1} J A (g_k \oplus I_{2n_E})} \right)^m ,$$

In the limit of infinite fast decoupling, this converges to

$$\lim_{m\to\infty} (S^{\text{res}}(t/m))^m = e^{t \sum_{g\in\mathcal{G}} (g \oplus I_{2n_E})^{-1} J A (g \oplus I_{2n_E})} \; .$$

We use the relation $(g \oplus I_{2n_E})^{-1} J = J (g \oplus I_{2n_E})^T$ which holds for all $g \in \mathsf{Sp}(2n_S, \mathbb{R})$ to conclude that

$$\lim_{m\to\infty} (S^{\text{res}}(t/m))^m = e^{tJ \sum_{g\in\mathcal{G}} (g \oplus I_{2n_E})^T A (g \oplus I_{2n_E})} \; . \tag{4.6}$$

The matrix A – when using the ordering of R from (4.3) – can be written in block form as

$$A = \begin{pmatrix} A_{SS} & A_{SE} \\ A_{ES} & A_{EE} \end{pmatrix}$$

where $A_{SS} \in \mathbb{R}^{2n_S \times 2n_S}$ and $A_{EE} \in \mathbb{R}^{2n_E \times 2n_E}$ are symmetric and $A_{SE} \in \mathbb{R}^{2n_S \times 2n_E}$, $A_{ES} \in \mathbb{R}^{2n_E \times 2n_S}$ are such that $A_{SE}^T = A_{ES}$. The exponential in Eq. (4.6) becomes

$$\sum_{g\in\mathcal{G}} (g \oplus I_{2n_E})^T A (g \oplus I_{2n_E}) = \begin{pmatrix} \sum_{g\in\mathcal{G}} g^T A_{SS}\, g & \sum_{g\in\mathcal{G}} g^T A_{SE} \\ \sum_{g\in\mathcal{G}} A_{SE}\, g & A_{EE} \end{pmatrix} \; . \tag{4.7}$$

Since $J = J_{n_S} \oplus J_{n_E}$ is block-diagonal, Eq. (3.20) holds if the off-diagonal terms in Eq. (4.7)

$$\frac{1}{|\mathcal{G}|} \sum_{g\in\mathcal{G}} A_{ES}\, g$$

vanish. This results in the following definition of a group achieving decoherence suppression in the limit.

Definition 4.2 *(First order decoherence suppression for bosonic systems).* A *decoherence suppression group* for $n_S \in \mathbb{N}$ system modes and $n_E \in \mathbb{N}$ environment modes is a finite subgroup $\mathcal{G} \subset \mathsf{Sp}(2n_S, \mathbb{R})$ such that

$$\sum_{g\in\mathcal{G}} g = 0_{2n_S} \; . \tag{4.8}$$

We have just shown that if $\mathcal{G}$ is a decoherence suppression group in the sense of Definition 4.2, then the deduced pulse sequence from Eq. (4.5) achieves decoherence suppression in the DD in the limit of infinitely fast pulse application, i.e., it satisfies Eq. (4.4): This is a simple consequence of inserting Eq. (4.8) into Eq. (4.7).

It is easy to see that the two-element group $\mathcal{G} = \{I_{2n_S} \oplus I_{2n_E}, -I_{2n_S} \oplus I_{2n_E}\} = \{\pm I_{2n_S+2n_E}\}$ satisfies condition (4.8) and hence achieves decoherence suppression such that in the limit of infinitely fast pulse application

$$\lim_{m\to\infty} (S^{\text{res}}(t/m))^m = e^{J \sum_{g\in\mathcal{G}} (g^T A_{SS} g) \oplus A_{EE}} = e^{J_{n_S} A_{SS}} \oplus e^{J_{n_E} A_{EE}} \; .$$

4.1.3 Homogenisation

The authors show that there is actually no group of symplectic matrices which achieves $S_S = I_S$ in Eq. (4.4) for all quadratic Hamiltonians. However, a weaker but nevertheless useful condition may hold: *homogenisation*. They assume that system and environment are originally decoupled, i.e., $A = A_{SS} \oplus A_{EE}$ where $A_{SS} \in \mathbb{R}^{2n_S \times 2n_S}$ in the original evolution. Since all considered matrices act non-trivial on the system only, it is convenient to consider the system only and drop the environment, i.e, $S^{\text{res}}(T) \in \mathbb{R}^{2n_S \times 2n_S}$.

Definition 4.3 *(Homogenisation).* A bosonic pulse sequence achieves *homogenisation* of the system with $n_S \in \mathbb{N}$ modes after time t in the limit infinitely fast pulse application if – under the assumption that system and environment are originally decoupled, i.e., $A \in \mathbb{R}^{2n_S \times 2n_S}$ – there are two constants $c_1, c_2 \in \mathbb{R}$ such that

$$\lim_{m\to\infty} (S^{\text{res}}(t/m))^m = c_1 I_{2n_S} + c_2 J \ . \tag{4.9}$$

Here, again, the authors derive a condition on the group of pulses such that Eq. (4.9) holds.

Consider a finite group $\mathcal{G} \subset \mathsf{Sp}(2n_S, \mathbb{R})$. Then the resulting symplectic evolution after a single cycle of application of the pulses $g_j g_{j-1}^{-1}$ at times $\frac{Tj}{|\mathcal{G}|}$ for $j = 1, \ldots, |\mathcal{G}|$ is given by

$$S^{\text{res}}(T) = \prod_{j=1}^{|\mathcal{G}|} g_j^{-1} e^{\frac{T}{|\mathcal{G}|} JA} g_j = \prod_{j=1}^{|\mathcal{G}|} e^{\frac{T}{|\mathcal{G}|} g_j^{-1} J A g_j} \ .$$

At time t after $m \in \mathbb{N}$ applications of this cycle it is

$$(S^{\text{res}}(t/m))^m = \left(\prod_{j=1}^{|\mathcal{G}|} e^{\frac{T}{m|\mathcal{G}|} g_j^{-1} J A g_j} \right)^m \ ,$$

which converges in the limit $m \to \infty$ to

$$\lim_{m\to\infty} (S^{\text{res}}(t/m))^m = e^{tJ \sum_{g\in\mathcal{G}} g^T A g} \ , \tag{4.10}$$

where we used again that $g^{-1} J = J g^T$.

Definition 4.4 *(Homogenisation group).* A *homogenisation group* for $n_S \in \mathbb{N}$ system modes is a finite subgroup $\mathcal{G} \subset \mathsf{Sp}(2n_S, \mathbb{R})$ such that

$$\frac{1}{|\mathcal{G}|} \sum_{g\in\mathcal{G}} g^T A g = c_A I_{2n_S} \ , \tag{4.11}$$

for some $c_A \in \mathbb{R}$ depending on A.

Let $\mathcal{G}$ be a homogenisation group. Then by insertion of Eq. (4.11) into Eq. (4.10)

$$\lim_{m\to\infty} (S^{\text{res}}(t/m))^m = e^{tc_A J} = \cos(tc_A) I_{2n_S} + \sin(tc_A) J_{n_S} \ ,$$

one achieves homogenisation in the limit of infinitely fast pulse application.

They show in Theorem 3 of their article that the group $\mathcal{G} = \langle I_2 \otimes O(n_S, \mathbb{Z}), J \rangle$ forms a homogenisation group for $n_S \in \mathbb{N}$ modes.

In the remainder of their work [12], Arenz, Burgarth and Hillier construct *random* bosonic pulse sequences – where elements of the decoherence suppression/homogenisation group are chosen at random with respect to the Haar measure – and numerically analyse their performance.

4.2 First order homogenisation from unitary 1-designs

In this section, we construct novel first order bosonic homogenisation schemes in the sense of Definition 4.4. More precisely, they are build from unitary 1-designs (cf. Definition 3.12) similarly to the construction of DD pulse sequences in the finite-dimensional setting from unitary 1-designs (cf. Lemma 3.13). In order to do so, the isomorphism

$$\eta : \mathsf{U}(n) \to \mathsf{Sp}(2n,\mathbb{R}) \cap \mathrm{O}(2n,\mathbb{R}) \quad , \quad \eta(U) \coloneqq \begin{pmatrix} \mathsf{Re}(U) & -\mathsf{Im}(U) \\ \mathsf{Im}(U) & \mathsf{Re}(U) \end{pmatrix} \tag{4.12}$$

between the unitary group and the orthogonal symplectic group proves to be a useful tool since it maps unitaries on $\mathbb{C}^n$ to symplectic orthogonal matrices on $\mathbb{R}^{2n}$. It may therefore be used to assign homogenisation pulses on $2n$ bosonic modes to unitary operators on $\mathbb{C}^n$: we apply the map (4.12) to elements of a unitary 1-design on $\log_2(n)$ qubits. To this end, consider a unitary 1-design $X = \{x_1, \ldots, x_L\}$ for $\mathsf{U}(n)$ which satisfies that $x_L = I$. Recall from Section 3.3.3 that on the finite-dimensional system $\mathbb{C}^n$ it defines a pulse sequence via

$$\{jT/L\}_{j=1}^{L} \quad , \quad \{x_j x_{j-1}^{\dagger}\}_{j=1}^{L} \, ,$$

where one sets $x_0 \coloneqq x_L$. By Lemma 3.13, this pulse sequence achieves first order dynamical decoupling which is guaranteed by the property

$$\frac{1}{L}\sum_{x\in X} x^{\dagger} H x = \frac{\mathrm{tr}(H)}{n} I_n \qquad \text{for every} \qquad H \in \mathbb{C}^{n\times n} \tag{4.13}$$

of elements of the unitary 1-design X. Here, we show that a similar argument can be made in the CV setting to deduce a homogenisation group on the phase space $\mathbb{R}^{2n}$ from a unitary 1-design.

Theorem 4.5 *(Homogenisation from 1-designs).* Let $X = \{x_1, \ldots, x_L\}$ be a unitary 1-design for $\mathsf{U}(n)$ and let η be the isomorphism from Eq. (4.12). Then the set

$$\mathcal{G} \coloneqq \{\eta(x) \mid x \in X\} \cup \{J\eta(x) \mid x \in X\}$$

satisfies

$$\frac{1}{|\mathcal{G}|}\sum_{g\in G} g^T A g = \frac{\mathrm{tr}(A)}{2n} \qquad \text{for all} \qquad A = A^T \in \mathbb{R}^{2n\times 2n} \, , \tag{4.14}$$

i.e., it is a homogenisation group in the sense of Definition 4.4. □

Proof. Let us check the homogenisation condition (4.14). Insertion of the set $\mathcal{G}$ into the left hand side of (4.14) gives

$$\frac{1}{|\mathcal{G}|}\sum_{g\in G} g^T A g = \frac{1}{2|X|}\sum_{x\in X} \left(\eta(x)^T A \eta(x) + \eta(x)^T J^T A J \eta(x)\right) \tag{4.15}$$

where $|\mathcal{G}| = 2|X|$. To further compute this quantity, let us write the matrix A as

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$$

where $A_{11}, A_{22} \in \mathbb{R}^{n\times n}$ are symmetric and $A_{12}^T = A_{21} \in \mathbb{R}^{n\times n}$. To shorten notation, we introduce the two quantities $\mathsf{R}_x \coloneqq \mathsf{Re}(x)$ and $\mathsf{I}_x \coloneqq \mathsf{Im}(x)$ for the real and imaginary part of the unitary 1-design element $x \in X$, respectively. The summands in Eq. (4.15) can be directly computed: one has

$$\begin{aligned} \eta(x)^T A \eta(x) &= \begin{pmatrix} \mathsf{R}_x^T & \mathsf{I}_x^T \\ -\mathsf{I}_x^T & \mathsf{R}_x^T \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} \mathsf{R}_x & -\mathsf{I}_x \\ \mathsf{I}_x & \mathsf{R}_x \end{pmatrix} \\ &= \begin{pmatrix} \big(\eta(x)^T A \eta(x)\big)_{11} & \big(\eta(x)^T A \eta(x)\big)_{12} \\ \big(\eta(x)^T A \eta(x)\big)_{21} & \big(\eta(x)^T A \eta(x)\big)_{22} \end{pmatrix} , \end{aligned}$$

where $\big(\eta(x)^T A \eta(x)\big)_{21} = \big(\eta(x)^T A \eta(x)\big)_{12}^T$ due to the symmetry of A and the remaining three submatrices are

$$\begin{aligned} \big(\eta(x)^T A \eta(x)\big)_{11} &= \mathsf{R}_x^T A_{11} \mathsf{R}_x + \mathsf{R}_x^T A_{12} \mathsf{I}_x + \mathsf{I}_x^T A_{21} \mathsf{R}_x + \mathsf{I}_x^T A_{22} \mathsf{I}_x \ , \\ \big(\eta(x)^T A \eta(x)\big)_{12} &= -\mathsf{R}_x^T A_{11} \mathsf{I}_x + \mathsf{R}_x^T A_{12} \mathsf{R}_x - \mathsf{I}_x^T A_{21} \mathsf{I}_x + \mathsf{I}_x^T A_{22} \mathsf{R}_x \ , \\ \big(\eta(x)^T A \eta(x)\big)_{22} &= \mathsf{I}_x^T A_{11} \mathsf{I}_x - \mathsf{I}_x^T A_{12} \mathsf{R}_x - \mathsf{R}_x^T A_{21} \mathsf{I}_x + \mathsf{R}_x^T A_{22} \mathsf{R}_x \ , \end{aligned} \tag{4.16}$$

and similarly

$$\begin{aligned} \eta(x)^T J^T A J \eta(x) &= \begin{pmatrix} \mathsf{I}_x^T & -\mathsf{R}_x^T \\ \mathsf{R}_x^T & \mathsf{I}_x^T \end{pmatrix} \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} \mathsf{I}_x & \mathsf{R}_x \\ -\mathsf{R}_x & \mathsf{I}_x \end{pmatrix} \\ &= \begin{pmatrix} \big(\eta(x)^T J^T A J \eta(x)\big)_{11} & \big(\eta(x)^T J^T A J \eta(x)\big)_{12} \\ \big(\eta(x)^T J^T A J \eta(x)\big)_{21} & \big(\eta(x)^T J^T A J \eta(x)\big)_{22} \end{pmatrix} , \end{aligned}$$

where again $\big(\eta(x)^T J^T A J \eta(x)\big)_{12}^T = \big(\eta(x)^T J^T A J \eta(x)\big)_{21}$ and where

$$\begin{aligned} \big(\eta(x)^T J^T A J \eta(x)\big)_{11} &= \mathsf{I}_x^T A_{11} \mathsf{I}_x - \mathsf{I}_x^T A_{12} \mathsf{R}_x - \mathsf{R}_x^T A_{21} \mathsf{I}_x + \mathsf{R}_x^T A_{22} \mathsf{R}_x \ , \\ \big(\eta(x)^T J^T A J \eta(x)\big)_{12} &= \mathsf{I}_x^T A_{11} \mathsf{R}_x + \mathsf{I}_x^T A_{12} \mathsf{I}_x - \mathsf{R}_x^T A_{21} \mathsf{R}_x - \mathsf{R}_x^T A_{22} \mathsf{I}_x \ , \\ \big(\eta(x)^T J^T A J \eta(x)\big)_{22} &= \mathsf{R}_x^T A_{11} \mathsf{R}_x + \mathsf{R}_x^T A_{12} \mathsf{I}_x + \mathsf{I}_x^T A_{21} \mathsf{R}_x + \mathsf{I}_x^T A_{22} \mathsf{I}_x \ . \end{aligned} \tag{4.17}$$

To further compute the quantity (4.15), one may use the 1-design property (4.13) of X. It particularly implies that the design average $\sum_x x^\dagger H x$ over real matrices $H \in \mathbb{R}^{n\times n}$ must be proportional to the identity with a real number constant (the trace of a real matrix is real). Hence, together with the decomposition

$$x^\dagger B x = (\mathsf{R}_x^T - i\mathsf{I}_x^T) B (\mathsf{R}_x + i\mathsf{I}_x) = \mathsf{R}_x^T B \mathsf{R}_x - i\mathsf{I}_x^T B \mathsf{R}_x + i\mathsf{R}_x^T B \mathsf{I}_x + \mathsf{I}_x^T B \mathsf{I}_x$$

one finds that for every real matrix $B \in \mathbb{R}^{n\times n}$

$$\begin{aligned} \frac{1}{|X|} \sum_{x\in X} \Big(\mathsf{R}_x^T B \mathsf{I}_x - \mathsf{I}_x^T B \mathsf{R}_x \Big) &= \mathsf{Im} \left(\sum_{x\in X} x^\dagger B x \right) = 0 \ , \\ \frac{1}{|X|} \sum_{x\in X} \Big(\mathsf{R}_x^T B \mathsf{R}_x + \mathsf{I}_x^T B \mathsf{I}_x \Big) &= \mathsf{Re} \left(\sum_{x\in X} x^\dagger B x \right) = \frac{\mathrm{tr}(B)}{n} I_n \ . \end{aligned} \tag{4.18}$$

The first line holds since the considered matrices B are real valued and hence have a real trace. In summary, inserting the entries from Eqs. (4.16) to (4.17) and Eqs. (4.18) into Eq. (4.15) becomes

$$\begin{aligned}\frac{1}{|\mathcal{G}|}\sum_{g\in G} g^T A g &= \frac{1}{2|X|}\begin{pmatrix}\sum_x (x^\dagger A_{11} x + x^\dagger A_{22} x) & 0\\ 0 & \sum_x (x^\dagger A_{11} x + x^\dagger A_{22} x)\end{pmatrix}\\ &= \frac{1}{2|X|}\operatorname{tr}(A) I_{2n} \ .\end{aligned}$$

■

With this result one may construct first order bosonic homogenisation schemes: any first order DD scheme on a qu*n*it with L pulses (forming a unitary 1-design $X \subset \mathsf{U}(n)$) is transformed into a first order homogenisation scheme with $2L$ Gaussian unitary pulses for n bosonic modes. Note that these bosonic pulses are passive since their associated symplectic matrices $\{\eta(x)\}_{x\in X}$ are orthogonal. For example, in the case of k qubits, i.e., if $n = 2^k$, a unitary 1-design consists of $n^2 = 2^{2k} = 4^k$ unitary pulses and can be associated to a first order homogenisation scheme on $n = 2^k$ modes where the latter uses $2n^2 = 2 \cdot 4^k$ passive Gaussian pulses.

4.3 Higher order decoherence suppression and homogenisation of CV systems

In this section, we define the decoherence model and the concepts of higher order decoherence suppression and homogenisation schemes.

4.3.1 Noise and pulse sequences on the symplectic level

Quadratic Hamiltonians and Gaussian unitary pulse sequences

We consider a bosonic system consisting of $n_S \in \mathbb{N}$ modes which interacts with an environment of $n_E \in \mathbb{N}$ modes with joint Hilbert space

$$\mathcal{H} = \mathcal{H}_S \otimes \mathcal{H}_E \quad \text{where} \quad \mathcal{H}_S = L^2(\mathbb{R}^{2n_S}),\ \mathcal{H}_E = L^2(\mathbb{R}^{2n_E}) \ .$$

More precisely, assume that this interaction is quadratic in the mode operators, i.e., described by the Hamiltonian

$$H^{\text{orig}}(t) = \frac{1}{2}\sum_{j,k=1}^{2n} A_{jk}(t) R_j R_k \tag{4.19}$$

where for every time $t \in [0,T]$, the matrix $A(t) \in \mathbb{R}^{2n\times 2n}$ is symmetric, $n \coloneqq n_S + n_E$ denotes the total number of bosonic modes and R is a vector of length $2n$ containing the mode operators Q_i and P_i in a suitable order. Let us mention that we will analyse the slightly more general case of *at most quadratic* terms, i.e., quadratic and linear terms, in Section 4.9.2.

The class of quadratic Hamiltonians (4.19) is ubiquitous in quantum optics. It is common to characterise CV Hamiltonians according to their powers of the field operators a and $a^\dagger$,

where those which are at most quadratic in the field operators describe *linear optics*. Such Hamiltonians generate unitaries which include single-mode phase rotations, beam splitters and squeezing operations (Gaussian unitaries, cf. Section 2.2.4). Since the terms with lower orders in the field operators a and $a^\dagger$ usually dominate the error sources, it is reasonable and practically relevant to focus on decoherence caused by Hamiltonians of the form of (4.19). A Hamiltonian of the form (4.19) generates dilations of Gaussian unitary channels on the system (cf. Section 2.2.5), i.e., such that a system state $\rho_S \in \mathcal{D}(L^2(\mathbb{R}^{n_S}))$ evolves as

$$\rho_S(t) = \mathrm{tr}_E \left[U^{\mathrm{orig}}(0,t)(\rho_S \otimes \rho_E) U^{\mathrm{orig}}(0,t)^\dagger \right] \qquad \text{for } t \in [0,T]$$

for a (typically Gaussian) environment state ρ_E. Here, $U^{\mathrm{orig}}(s,t)$ denotes the evolution generated by the original Hamiltonian (4.19) from time s to time t. Dilations of Gaussian unitaries include specifically the following noise channels on the system: the thermal noise channel, the attenuation channel and the amplification channel. Besides quantum optics, there are also other systems for which quadratic Hamiltonians (4.19) provide good approximations of typical decoherence processes in suitable parameter regimes. These systems include atomic ensembles [119], ion traps [106] and opto- and nanomechanical resonators [131, 180].

Let a pulse sequence on this system be given by the pair of times $\{t_j\}_{j=1}^L$ such that $0 \leq t_1 < \cdots < t_L \leq T$ and Gaussian unitary pulses $\{U_j\}_{j=1}^L \subset \mathsf{U}(L^2(\mathbb{R}^{n_S}))$ acting on the system. The goal is to analyse the resulting evolution

$$U^{\mathrm{res}}(T) \coloneqq U^{\mathrm{orig}}(t_L, T)(U_L \otimes I_E) \cdots U^{\mathrm{orig}}(t_1, t_2)(U_1 \otimes I_E) U^{\mathrm{orig}}(0, t_1)$$

after time T, where the evolution under the uncontrolled original Hamiltonian is interleaved with the Gaussian unitary pulses. We ask whether $U^{\mathrm{res}}(T)$ approximately achieves decoherence suppression, i.e., $U^{\mathrm{res}}(T) \approx U_S \otimes U_E$ for some $U_S \in \mathsf{U}(L^2(\mathbb{R}^{n_S}))$ and $U_E \in \mathsf{U}(L^2(\mathbb{R}^{n_E}))$, or homogenisation, i.e., $U_S \approx e^{i\omega H_0}$ for H_0 from Eq. (2.40) for some $\omega \in \mathbb{R}$.

Pulse sequences at the symplectic level

Since all unitaries considered are Gaussian, the above analysis may be conducted at the level of the symplectic group $\mathsf{Sp}(2n_S + 2n_E, \mathbb{R})$ and its associated Lie algebra $\mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$, respectively, (cf. Definition 2.29 for a definition of these sets). For convenience, we fix the ordering of the mode operator in the vector R to be given by

$$R^T = (Q_1^S, Q_2^S, \ldots, Q_{n_S}^S, P_1^S, P_2^S, \ldots, P_{n_S}^S, Q_1^E, Q_2^E, \ldots, Q_{n_E}^E, P_1^E, P_2^E, \ldots, P_{n_E}^E) \,, \tag{4.20}$$

where Q_j^S denotes the position quadrature of the jth system mode and Q_k^E the position quadrature of the kth environment mode for $j = 1, \ldots, n_S$ and $k = 1, \ldots, n_E$, respectively. The momentum operators P_j^S and P_k^E are defined similarly. In this convention the matrix J, which is defined by the relation $[R_j, R_k] = iJ_{jk} I_{2n_S+2n_E}$ for $j = 1, \ldots, n_S$ and $k = 1, \ldots, n_E$, is of the block-diagonal matrix form

$$J \coloneqq J_{n_S} \oplus J_{n_E} = \begin{pmatrix} 0 & I_{n_S} & 0 & 0 \\ -I_{n_S} & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{n_E} \\ 0 & 0 & -I_{n_E} & 0 \end{pmatrix} . \tag{4.21}$$

The original evolution $U^{\mathrm{orig}}(t,s)$ generated by $H^{\mathrm{orig}}(t)$ translates at the symplectic level to its associated symplectic evolution $S^{\mathrm{orig}}(t,s)$ generated by an *original generator* $X^{\mathrm{orig}}(t) \in$

$\mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$ for $t \in [0, T]$. Recall that by the relation between symplectic matrices and Gaussian unitaries (cf. Section 2.2.4), such a generator $X^{\text{orig}}(t)$ is in one-to-one correspondence with the Hamiltonian (4.19) via the equation

$$X^{\text{orig}}(t) = JA(t) \quad \text{for } t \in [0, T] \ , \tag{4.22}$$

where J is the matrix from Eq. (4.21) and $A(t)$ is the symmetric matrix from Eq. (4.19).

Throughout this chapter, we consider decoherence introduced by *quadratic Hamiltonians with analytic time-dependence*: Let us list the corresponding assumptions on the generator X^{orig}.

Definition 4.6 *(Noise model).* An *original generator* $X^{\text{orig}} : [0, T] \to \mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$ has *analytic time dependence* if for any $t \in [0, T]$ it is given by a matrix of the form

$$X^{\text{orig}}(t) = \sum_{r=0}^{\infty} X_r^{\text{orig}} t^r \quad \text{for } t \in [0, T] \ ,$$

where for every $r \in \mathbb{N}_0$ there exists $X_r^{\text{orig}} \in \mathbb{R}^{(2n_S+2n_E)\times(2n_S+2n_E)}$. The evolution S^{orig} generated by X^{orig} is called the *(symplectic) original evolution.*

In the case that X^{orig} is not time-dependent, the original evolution $S^{\text{orig}} : [0, T] \to \mathsf{Sp}(2n_S + 2n_E, \mathbb{R})$ is the one-parameter group of symplectic matrices generated by X^{orig}. However, if X^{orig} depends on time, then we write $S^{\text{orig}}(t, s)$ to denote the time evolution under X^{orig} between s and t.

One may define the notion of a pulse sequence at the symplectic level. A pulse sequence of Gaussian unitaries translates to a pulse sequence at the symplectic level as

$$\left(\{t_j\}_{j=1}^L, \{U_j\}_{j=1}^L\right) \quad \longleftrightarrow \quad \left(\{t_j\}_{j=1}^L, \{S_j\}_{j=1}^L\right) \ .$$

Here, one employs the same pulse times t_j and every Gaussian unitary U_j is replaced by its associated symplectic matrix $S_j \in \mathsf{Sp}(2n_S, \mathbb{R})$.

Definition 4.7 *(CV Pulse sequence).* A *(CV) pulse sequence* for a system with $n_S \in \mathbb{N}$ bosonic modes is given by a tuple

$$\left(\{t_\lambda\}_{\lambda\in\Lambda}, \{S_\lambda\}_{\lambda\in\Lambda}\right) \ ,$$

where $\Lambda := \{\lambda_1, \ldots, \lambda_L\}$ is a set of indices with an ordering induced by the pulse times t_λ such that $0 \leq t_{\lambda_1} < t_{\lambda_2} < \cdots < t_{\lambda_L} \leq T$ and where $S_\lambda \in \mathsf{Sp}(2n_S, \mathbb{R})$ for every $\lambda \in \Lambda$. The latter are called the *bosonic pulses* which satisfy that for every $j = 1, \ldots, L$ there is a matrix $X_{\lambda_j} \in \mathfrak{sp}(2n_S, \mathbb{R})$ such that $S_{\lambda_j} = e^{X_{\lambda_j}}$ and furthermore that

$$\prod_{j=1}^{L} S_{\lambda_j} = I_{2n_S} \ . \tag{4.23}$$

We note that the different notation for labelling the pulses, where we choose a general index set Λ instead of $\{1, \ldots, L\}$, will prove useful in the context of nested Uhrig DD and homogenisation.

The control generator associated to a CV pulse sequence from Definition 4.7 is defined as

$$X^{\mathrm{control}}(t) = \sum_{j=1}^{K_t} \delta(t - t_{\lambda_j})(X_{\lambda_j} \oplus 0_{2n_E}) \qquad \text{for } t \in [0,T] \ ,$$

where the matrix $X_{\lambda_j} \in \mathfrak{sp}(2n_S, \mathbb{R})$ exponentially generates the pulse $S_{\lambda_j} = e^{X_{\lambda_j}}$ for $j = 1, \ldots, L$, and where

$$K_t \coloneqq \max\{j = 1, \ldots, L \mid t_{\lambda_j} \leq t\} \ . \tag{4.24}$$

Furthermore, we write $B_S \oplus B_E = \mathrm{diag}(B_S, B_E)$ for the two matrices $B_S \in \mathbb{R}^{2n_S \times 2n_S}$ and $B_E \in \mathbb{R}^{2n_E \times 2n_E}$. As a consequence, the (symplectic) control evolution

$$S^{\mathrm{control}} : [0,T] \to \mathsf{Sp}(2n_S + 2n_E, \mathbb{R}) \quad , \quad S^{\mathrm{control}}(t) \coloneqq \prod_{j=1}^{K_t} S_{\lambda_j} \oplus I_{2n_E} \ ,$$

is given by the product of the symplectic pulses up to the considered time t.

Analysis of bosonic pulse sequences

Let us present the strategy how we analyse the decoherence suppression and homogenisation properties of such bosonic pulse sequences at the symplectic level. Let $X^{\mathrm{orig}}(t)$ as in Definition 4.6 be given. After application of a pulse sequence from Definition 4.7, the resulting symplectic evolution $S^{\mathrm{res}}(T)$, generated by $X^{\mathrm{orig}}(t) + X^{\mathrm{control}}(t)$, is defined as

$$S^{\mathrm{res}}(T) \coloneqq \left[\prod_{j=1}^{L} S^{\mathrm{orig}}(t_{\lambda_{j+1}}, t_{\lambda_j})(S_{\lambda_j} \oplus I_{2n_E}) \right] S^{\mathrm{orig}}(t_{\lambda_1}, 0) \ . \tag{4.25}$$

Here $t_{\lambda_{L+1}} \coloneqq T$. Throughout this chapter, we construct CV pulse sequences and analyse their effect on this resulting evolution $S^{\mathrm{res}}(T)$. In order to do so, it is convenient to write the matrices in block from, i.e., to separate them into a system and environment part. In this block form, the original generator is given by

$$X^{\mathrm{orig}}(t) = \begin{pmatrix} X_{SS}(t) & X_{SE}(t) \\ X_{ES}(t) & X_{EE}(t) \end{pmatrix} \ , \tag{4.26}$$

for $t \in [0,T]$. A matrix of the form (4.26) is a symplectic generator, i.e., an element of $\mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$, if and only if the above block matrices satisfy the equations

$$\begin{aligned} J_S^T X_{SS}(t) = X_{SS}(t)^T J_S \qquad &\Leftrightarrow \qquad X_{SS}(t) \in \mathfrak{sp}(2n_S, \mathbb{R}) \ , \\ J_S^T X_{SS}(t) = X_{SS}(t)^T J_S \qquad &\Leftrightarrow \qquad X_{EE}(t) \in \mathfrak{sp}(2n_E, \mathbb{R}) \ , \\ J_E^T X_{ES}(t) = X_{SE}(t)^T J_S \ . & \end{aligned}$$

This follows from the symmetry of the matrix A to which X^{orig} is related by Eq. (4.22).

Similar to finite-dimensional DD, we change into the interaction picture associated with the control evolution, the so-called toggling frame: the toggling frame generator is given by

$$X^{\mathrm{tf}}(t) \coloneqq S^{\mathrm{control}}(t)^{-1} X^{\mathrm{orig}}(t) S^{\mathrm{control}}(t) = \begin{pmatrix} P(t)^{-1} X_{SS}(t)\, P(t) & P(t)^{-1} X_{SE}(t) \\ X_{ES}(t) P(t) & X_{EE}(t) \end{pmatrix} \tag{4.27}$$

for the submatrices from Eq. (4.26) and where

$$P(t) = \prod_{j=1}^{K_t} S_{\lambda_j}$$

is the product of all pulses up to time t. It generates the toggling frame evolution

$$S^{\mathrm{tf}}(T) \coloneqq \sum_{k=1}^{\infty} \int_0^T \int_0^{s_1} \cdots \int_0^{s_{k-1}} X^{\mathrm{tf}}(s_k) \ldots X^{\mathrm{tf}}(s_2) X^{\mathrm{tf}}(s_1)\, ds_k \ldots ds_2 ds_1 \tag{4.28}$$

at time T. The analytic time dependence of $X^{\mathrm{orig}}(t)$ furthermore implies that for every $r \in \mathbb{N}_0$ there exist $X_{BC,r} \in \mathbb{R}^{2n_B \times 2n_C}$ for $B, C \in \{S, E\}$ such that

$$X_{BC}(t) = \sum_{r=0}^{\infty} X_{BC} t^r \ .$$

As a consequence, the toggling frame evolution (4.28) can be computed

$$\begin{aligned} S^{\mathrm{tf}}(T) = \sum_{k=1}^{\infty} \int_0^T \cdots \int_0^{s_{k-1}} & \begin{pmatrix} P(s_k)^{-1} \sum_{r=0}^{\infty} X_{SS,r}\, s_k^r P(s_k) & P(s_k)^{-1} \sum_{r=0}^{\infty} X_{SE,r}\, s_k^r \\ \sum_{r=0}^{\infty} X_{ES,r}\, s_k^r P(s_k) & \sum_{r=0}^{\infty} X_{EE,r}\, s_k^r \end{pmatrix} \\ & \cdots \begin{pmatrix} P(s_1)^{-1} \sum_{r=0}^{\infty} X_{SS,r} s_1^r\, P(s_1) & P(s_1)^{-1} \sum_{r=0}^{\infty} X_{SE,r} s_1^r \\ \sum_{r=0}^{\infty} X_{ES,r} s_1^r P(s_1) & \sum_{r=0}^{\infty} X_{EE,r} s_1^r \end{pmatrix} ds_k \ldots ds_1 . \end{aligned} \tag{4.29}$$

One may expand $S^{\mathrm{tf}}(T)$ in a Dyson power series, i.e., in orders of the total sequence time T. Since the relation (4.23) implies that $S^{\mathrm{tf}}(T) = S^{\mathrm{res}}(T)$, this Dyson expansion of the toggling frame evolution corresponds to the one of the resulting evolution $S^{\mathrm{res}}(T)$. We note that this strategy – changing to the toggling frame and compute the corresponding Dyson series – resembles the one that we apply in the context of multi-qubit DD.

Let us furthermore argue why the block-diagonal form of matrices, i.e., the ordering from Eq. (4.20), where we separated the system quadratures in the first $2n_S$ entries from the ones related to the environment in the last $2n_E$ entries, is a suitable choice. A block-diagonal $S_S \oplus S_E$ at the level of symplectic matrices corresponds a decoupled unitary evolution $U \otimes V$ for $U \in \mathsf{U}(\mathcal{H}_S)$ and $V \in \mathsf{U}(\mathcal{H}_E)$ involving no interaction of the system and the environment, where U and V are Gaussian unitaries that corresponds to the symplectic matrices S and S_E, respectively, according to the metaplectic representation (2.37) (cf. Definition 2.30).

4.3.2 Higher order decoherence suppression

As discussed by Arenz, Burgarth and Hillier [12], one may investigate the two concepts of decoherence suppression and homogenisation separately. Let us first consider *decoherence suppression* of system-environment interactions: This is achieved by a pulse sequence if system and environment (approximately) do not interact with each other.

At the symplectic level, exact decoherence suppression after time T is achieved if the resulting symplectic evolution if of the form

$$S^{\mathrm{res}}(T) = S^{\mathrm{res}}_{SS} \oplus S^{\mathrm{res}}_{EE} \ ,$$

for some $S^{\mathrm{res}}_{SS} \in \mathsf{Sp}(2n_S, \mathbb{R})$ and $S^{\mathrm{res}}_{EE} \in \mathsf{Sp}(2n_E, \mathbb{R})$. This is equivalent to the existence of an effective generator $X^{\mathsf{eff}} \in \mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$ of the form

$$X^{\mathsf{eff}} = X_{SS} \oplus X_{EE} \quad \text{such that} \quad S^{\mathrm{res}}(T) = e^{TX^{\mathsf{eff}}} \ , \tag{4.30}$$

for some $X_{SS} \in \mathfrak{sp}(2n_S, \mathbb{R})$ and $X_{EE} \in \mathfrak{sp}(2n_E, \mathbb{R})$. Using the ordering of the mode operators from (4.20), this implies that the generator involves no terms coupling the system mode operators with those of the environment. Let us define the notion of approximate decoherence suppression up to a certain order N in T as follows.

Definition 4.8 *(Higher order decoherence suppression).* Consider a bosonic system of $n_S \in \mathbb{N}$ modes and an environment of $n_E \in \mathbb{N}$ modes. Let $X^{\text{orig}} : [0,T] \to \mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$ have analytic time-dependence, cf. Definition 4.6.

A CV pulse sequence $(\{t_\lambda\}_{\lambda\in\Lambda}, \{S_\lambda\}_{\lambda\in\Lambda})$ (cf. Definition 4.7) achieves universal *Nth order decoherence suppression* after time $T > 0$ if and only if for every X^{orig} satisfying the above assumptions, there are $S_{SS} \in \mathsf{Sp}(2n_S, \mathbb{R})$ and $S_{EE} \in \mathsf{Sp}(2n_E, \mathbb{R})$ such that

$$\left\| S^{\text{res}}(T) - \begin{pmatrix} S_{SS} & 0 \\ 0 & S_{EE} \end{pmatrix} \right\| = O(T^{N+1}) \,, \tag{4.31}$$

where $S^{\text{res}}(T)$ is the resulting evolution from Eq. (4.25).

A pulse CV sequence may also be non-universal, i.e., we say that it achieves Nth order decoherence suppression of the original evolution generated by X^{orig} if there are $S_{SS} \in \mathsf{Sp}(2n_S, \mathbb{R})$ and $S_{EE} \in \mathsf{Sp}(2n_E, \mathbb{R})$ such that Eq. (4.31) is satisfied, where $S^{\text{res}}(T)$ is given by (4.25) and S^{orig} is generated by this X^{orig}. We note that in Theorem 4.8 and throughout this chapter, we use the Frobenius norm which is defined as $\|B\| := \sqrt{\text{tr}(B^\dagger B)}$ for a matrix B.

When analysing whether property (4.31) is satisfied for a given pulse sequence, we will proceed similarly as in the finite-dimensional case where we considered unitaries. At the symplectic level, we analyse whether the toggling frame evolution $S^{\text{tf}}(T)$ from Eq. (4.28) is approximately (up to order N in T) of a direct sum form, i.e., whether its off-diagonal blocks $S^{\text{tf}}_{SE}(T)$ and $S^{\text{tf}}_{ES}(T)$ vanish up to order N in T.

At the level of unitaries and Hamiltonians, exact decoherence suppression, i.e. Eq. (4.30), translates to the existence of an effective Hamiltonian H^{eff} of the form

$$H^{\text{eff}} = H_S \otimes I_E + I_S \otimes H_E \quad \text{such that} \quad U^{\text{res}}(T) = e^{-iTH^{\text{eff}}} = e^{-iTH_S} \otimes e^{-iTH_E} \,,$$

where $H_S \in \mathcal{L}_{sa}(\mathcal{H}_S)$ and $H_E \in \mathcal{L}_{sa}(\mathcal{H}_E)$ and where these quantities are related to the symplectic generator $X^{\text{eff}} = JA^{\text{eff}}$ via $H^{\text{eff}} = \frac{1}{2}\sum_{j,k=1}^{2n_S+2n_E} {A^{\text{eff}}}_{jk} R_j R_k$.

In Section 4.6.2, we construct pulse sequences which achieve Nth order universal decoherence suppression.

4.3.3 Higher order homogenisation

As previously discussed (cf. the work by Arenz, Burgarth and Hillier [12] and the discussion thereof in Section 4.3.2), homogenisation can be analysed in the context of an already decoupled system-environment evolution, i.e., when system and environment are assumed to not interact with each other. Conceptually, this simplifies the analysis of homogenisation and allows to study decoherence suppression and homogenisation separately. Practically, one may still achieve a homogenised system evolution of an originally non-decoupled one by combining decoherence suppression and homogenisation schemes, more precisely, by concatenating one into the other, what is explained in more detail after the definition.

But first recall that by homogenisation of an evolution, we mean that the resulting symplectic evolution is reduced to that of non-interacting harmonic oscillators, which all rotate at the same averaged frequency. More precisely, exact homogenisation is achieved if

$$S^{\mathrm{res}}(T) = e^{\omega J_{n_S}} \oplus S_{EE} \tag{4.32}$$

for some $\omega \in \mathbb{R}$ and $S_{EE} \in \mathsf{Sp}(2n_E, \mathbb{R})$. Approximate homogenisation of an already decoupled evolution is defined as follows.

Definition 4.9 *(Higher order homogenisation).* Consider a system of $n_S \in \mathbb{N}$ modes and an environment of $n_E \in \mathbb{N}$ modes. Assume that the system is already decoupled from the environment, i.e., the original generator $X^{\mathrm{orig}} : [0,T] \to \mathfrak{sp}(2n_S+2n_E, \mathbb{R})$ from Definition 4.6 satisfies

$$X_{SE}(t) = X_{ES}(t) = 0 \qquad \text{for } t \in [0,T] \ , \tag{4.33}$$

when written in the block form of Eq. (4.26).

A CV pulse sequence $(\{t_\lambda\}_{\lambda\in\Lambda}, \{S_\lambda\}_{\lambda\in\Lambda})$ (cf. Definition 4.7) achieves universal *Nth order homogenisation* after time $T > 0$ if and only if for every X^{orig} satisfying the above assumptions there are $c_1, c_2 \in \mathbb{R}$ such that

$$\|S^{\mathrm{res}}_{SS}(T) - c_1 I_{2n_S} - c_2 J_{2n_S}\| = O(T^{N+1}) \ , \tag{4.34}$$

where $S^{\mathrm{res}}_{SS} \in \mathsf{Sp}(2n_S, \mathbb{R})$ is the system part of $S^{\mathrm{res}}(T) = S^{\mathrm{res}}_{SS}(T) \oplus S^{\mathrm{res}}_{EE}(T)$ for the resulting evolution from Eq. (4.25) and where $S^{\mathrm{res}}_{EE} \in \mathsf{Sp}(2n_E, \mathbb{R})$ is the its environment part.

Similar to decoherence suppression, one may drop the notion of universality, i.e., a pulse sequence achieves homogenisation up to order N in T for a specific original generator X^{orig} if for this X^{orig} there are $c_1, c_2 \in \mathbb{R}$ such that Eq. (4.34) is satisfied. Furthermore, by the assumption (4.33) on the original generator X^{orig}, every CV pulse sequence achieves that the resulting evolution is of block form $S^{\mathrm{res}}(T) = S^{\mathrm{res}}_{SS}(T) \oplus S^{\mathrm{res}}_{EE}(T)$ since the pulses act on the system only.

Let us briefly argue that the resulting evolution of an Nth order homogenisation scheme is close to an evolution of the form (4.32). Let an Nth order homogenisation scheme be given. By construction $S^{\mathrm{res}}(T)$ is symplectic and using (4.34), this implies that there are $c_1, c_2 \in \mathbb{R}$ such that $\|(c_1^2 + c_2^2) J_{n_S} - J_{n_S}\| = O(T^{N+1})$ and hence these constants satisfy $c_1^2 + c_2^2 = 1 + \epsilon$ where $\epsilon = O(T^{N+1})$. Then we can choose ω such that

$$\cos(\omega T) = \frac{c_1}{\sqrt{1+\epsilon}} \quad \text{and} \quad \sin(\omega T) = \frac{c_2}{\sqrt{1+\epsilon}} \ .$$

Direct computation shows that $\|c_1 I_{2n_S} + c_2 J_{n_S}\| = \sqrt{2n_S}\sqrt{(c_1^2 + c_2^2)}$ which we use to compute

$$\begin{aligned}
\left\|(c_1 I_{2n_S} + c_2 J_{n_S}) - e^{\omega T J_{n_S}}\right\| &= \|c_1 I_{2n_S} + c_2 J_{n_S} - \cos(\omega T) I_{2n_S} - \sin(\omega T) J_{n_S}\| \\
&= \left|1 - \frac{1}{\sqrt{1+\epsilon}}\right| \|c_1 I_{2n_S} + c_2 J_{n_S}\| \\
&= O\left(|1 - \frac{1}{\sqrt{1+\epsilon}}| \cdot \sqrt{c_1^2 + c_2^2}\right) \\
&= O\left(|1 - \frac{1}{\sqrt{1+\epsilon}}| \cdot \sqrt{1+\epsilon}\right) = O(\epsilon) \ .
\end{aligned}$$

As a consequence, $S^{\mathrm{res}}(T)$ is equal to $e^{\omega J_{n_S}}$ for some $\omega \in \mathbb{R}$ up to order N in T.

Note that at the Hamiltonian level, the assumption (4.33) that system and environment do not interact with each other is equivalent to

$$H^{\text{orig}}(t) = H_S(t) \otimes I_E + I_S \otimes H_E(t) \qquad \text{for } t \in [0,T] \ ,$$

where $H_S(t)$ and $H_E(t)$ are quadratic system and environment Hamiltonians, respectively. Moreover, exact homogenisation in the form of Eq. (4.32) corresponds to

$$U^{\text{res}}(T) = e^{-i\omega H_0 T} \otimes U_E \tag{4.35}$$

for some $U_E \in \mathsf{U}(\mathcal{H}_E)$, for $\omega \in \mathbb{R}$ and where $H_0 = \frac{1}{2}\sum_{j,k=1}^{n_S}(Q_j^2 + P_j^2)$ is the free oscillator Hamiltonian on all system modes. It is obvious that a decoupled and homogenised evolution of the form of Eq. (4.35) does not couple system and environment, and furthermore neither the different system modes nor the different eigenspaces of the free oscillator Hamiltonian H_0; more precisely, such an evolution provides decoherence-free subspaces (cf. Definition 3.6) where the eigenspaces of H_0 can be used to encode logical qubits. A state $|\psi\rangle$ which is fully supported at the level $k \in \mathbb{N}$ eigenspace of H_0 of a specific mode, can only be altered by as the unitary $|\psi\rangle \mapsto e^{-i\omega k T}|\psi\rangle$ and the recovery can be chosen as $e^{i\omega k T} I$.

An Nth order decoherence suppression scheme and a homogenisation scheme of the same order can be combined by concatenation: In between every pulse application of the latter scheme, one does not let the system and the environment evolve freely under the decoherence Hamiltonian but instead, one applies a decoherence suppression scheme. Then, one achieves a decoupled and homogenised evolution (4.32) up to order N in the total sequence time of the outer homogenisation scheme T for any original generator in the sense of Definition 4.6.

In Section 4.8.3, we construct higher order homogenisation schemes from multi-qubit DD schemes.

4.4 Prior work on higher order DD in finite dimensions

Let us return to DD in the finite-dimensional case for this section. Here, we recap prior work on pulse sequences which achieve higher order DD for qubit and multi-qubit systems (in the sense of Definition 3.9), starting with concatenated DD in Section 4.4.1 and then presenting Uhrig and Nested Uhrig DD in Sections 4.4.2 and 4.4.3, respectively. These two latter (multi)-qubit pulse sequences are the ones which are used as a starting point of the CV schemes proposed later in Sections 4.6 and 4.8.

4.4.1 Concatenated dynamical decoupling

Consider a single qubit system with Hilbert space $\mathcal{H}_S = \mathbb{C}^2$. Recall the first order *universal decoupling* pulse sequence $\mathsf{f}X\mathsf{f}Z\mathsf{f}X\mathsf{f}Z$ as described in Section 3.3.3, Eq. (3.36). Starting from this scheme, Khodjasteh and Lidar [88] introduced a technique to deduce an Nth order DD scheme from this for an arbitrary higher decoupling order $N \in \mathbb{N}$; It is called *Concatenated Dynamical Decoupling* (CDD).

The central idea is to recursively embed the first order pulse sequence into itself. With each embedding, one higher DD order is achieved. Define the first order scheme as $\mathsf{p}_1 := \mathsf{f}_0 X \mathsf{f}_0 Z \mathsf{f}_0 X \mathsf{f}_0 Z$ where $\mathsf{f}_0 = U^{\text{orig}}(T/4) = e^{-iH^{\text{orig}}T/4}$ is the system's uncontrolled evolution

between pulses. The N-time concatenated pulse sequence is p_N where the kth nesting level is recursively defined as

$$\mathsf{p}_k \coloneqq \mathsf{p}_{k-1} X \mathsf{p}_{k-1} Z \mathsf{p}_{k-1} X \mathsf{p}_{k-1} Z \qquad \text{where } \mathsf{p}_0 \coloneqq \mathsf{f}_0 \ .$$

The number of pulses in p_N is 4^N .

In their article from 2005, Khodjasteh and Lidar [88] showed that (for ideal pulses) the pulse sequence p_N makes the Nth order of the Magnus series vanish. In [88, 89], Khodjasteh and Lidar compared CDD to PDD for the universal first order DD scheme $\mathsf{f}X\mathsf{f}Z\mathsf{f}X\mathsf{f}Z$ (cf. Section. 3.3.3) showing superior performance of the CDD sequence for large parameter regimes of specific system-environment couplings.

CDD sequences can be constructed for other pulse sequences as well. Let a pulse sequence with pulse times $t_1, \ldots, t_L$ and unitary pulses $U_1, \ldots, U_L$ be given. For the first nesting level of the concatenated sequence, one chooses the above pulse times and pulses. For the second nesting level – instead of letting system and environment evolve freely between the pulses of the first nesting level – the same original sequence is applied between every pair of original pulse times. This procedure of recursively repeated, applying the pulse sequence in between the pulses of the next lower concatenation level. Starting with $U^{[0]}(t) \coloneqq U^{\text{orig}}(t)$, the Nth order DD sequence is recursively defined via

$$U^{[k]}(T) = U^{[k-1]}(T - t_L)(U_L \otimes I_E)U^{[k-1]}(t_L - t_{L-1}) \cdots (U_1 \otimes I_E)U^{[k-1]}(t_1)$$

for $k = 1, \ldots, N$. One calls the pulses associated with the above evolution $U^{[k]}(T)$ the kth nesting level.

Efficiency in DD protocols should be formulated in terms of the resource costs – the number and simplicity of the applied control pulses – and the benefit – the error reduction compared to the case where no decoupling is applied. One useful figure of merit is the relation between the decoupling order and the number of pulses necessary to achieve this DD order.

4.4.2 Uhrig dynamical decoupling for qubits

In CDD, the number of required pulses scales exponentially with the DD order. In the case of a single qubit and pure dephasing noise one would need 4^N pulses to achieve Nth order DD. In seminal work, Uhrig [161] considered *optimising the pulse times* to achieve higher order DD (extending the Carr-Purcell-Meiboom-Gill cycle [34, 112]) outperforming the number of required pulses in CDD. He studied the spin-boson model, which couples a single qubit to a bosonic bath via pure dephasing noise.

Let us recall the first example from Section 3.3.2, decoherence is caused by the Hamiltonian $H^{\text{orig}} = I_S \otimes B_0 + \sigma_z \otimes B_1$ where $B_0, B_1 \in \mathcal{L}_{sa}(\mathcal{H}_E)$ for some environment $\mathcal{H}_E$. The application of two equidistant σ_x-pulses at times $T/2$ and T achieves first order DD (or equivalently DD in the limit of infinitely fast application). Let us now consider Uhrig's question: Can different pulse timings achieve a higher DD order? It is enlightening to look at the lowest order term in the Magnus expansion for arbitrary pulse timings $t_1, t_2 \in [0, T]$ which is

$$\int_0^T H^{\text{tf}}(\tau)d\tau = TI_S \otimes B_0 + \int_0^{t_1} \sigma_z \otimes B_1 d\tau - \int_{t_1}^{t_2} \sigma_z \otimes B_1 d\tau + \int_{t_2}^{T} \sigma_z \otimes B_1 d\tau \ .$$

One can directly read off that this term acts trivially on the system – first order DD is achieved – if the pulse times t_1, t_2 are chosen such that $t_1 + (T - t_2) = t_2 - t_1$. The first order

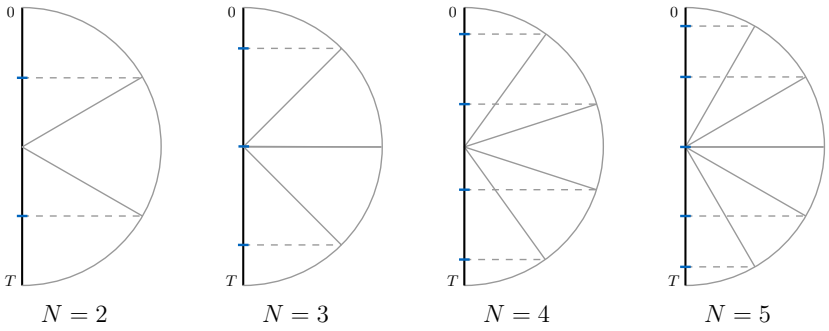


Figure 4.1: Construction of the Uhrig DD times from Eq. (4.36) for the cases $N \in \{2, 3, 4, 5\}$. First, one constructs a semicircle around the centre of vertical time interval $[0, T]$ and divides this semicircle into $N + 1$ segments of equal size (angle). Then, the orthogonal projections (dotted lines) of the segments onto the interval $[0, T]$ return the N Uhrig times (blue). Note that for odd N, the Uhrig scheme includes an additional pulse at time T.

DD condition does not fully determine the pulse times. In particular, there is some freedom in the choice of pulse times. A natural next question is whether one can use this freedom to achieve higher order DD.

For the spin-boson model, Uhrig found that applying σ_x-pulses at times $t_j = \Delta_j T$, where one defines

$$\Delta_j \coloneqq \sin^2\left(\frac{j\pi}{2(N+1)}\right) \qquad \text{for } j = 1, 2, \ldots, N, \tag{4.36}$$

eliminates the dephasing up to order N in T. This pulse sequence is called the *Uhrig Dynamical Decoupling* (UDD) sequence, sometimes denoted with the index N as UDD_N. Yang and Liu proved that the UDD_N sequence achieves universal Nth order DD [184]; for a proof of the universality see also [172]. We note that if N is odd, an additional σ_x-pulse has to be applied at the end of the DD period (at time T) to satisfy the condition $U^{\mathrm{control}}(T) = I_S$ [177]. The times Δ_j from (4.36) are called *Uhrig times* or *UDD times*. Uhrig furthermore showed that the UDD times are optimal in the sense that no other pulse timings can achieve a given DD order using less pulses. The UDD sequence achieves Nth order DD using N (or $N + 1$ if N is odd) pulses. Here, the number of required pulses scales only *linearly* in the decoupling order N. This is an exponential improvement over the scaling achieved by CDD where one would need 2^N pulses to achieve Nth order DD for pure dephasing noise. Fig. 4.1 shows the graphical construction of the UDD times (4.36).

Uhrigs idea was quickly extended to more general decoherence models and systems. First the decoherence Hamiltonian was allowed to contain errors of type σ_x and σ_z. The *Quadratic Dynamical Decoupling* (QDD) sequence by West et al. [177] consists of two nested (or concatenated) UDD sequences: a standard UDD sequence is applied at the so-called outer level and a modified UDD sequence with σ_z-instead of σ_x-pulses is used in the inner nesting level. On a first nesting level, pulses σ_x are applied at UDD times $t_j = \Delta_j T$ from Eq. (4.36) for $j = 1, \ldots, N$. On the second nesting level, σ_z-pulses are applied at nested UDD times

$$\Delta_{k-1} T + \Delta_j (\Delta_k - \Delta_{k-1}) T \qquad \text{for } j = 1, 2, \ldots, N,\ k = 1, 2, \ldots, N+1$$

where Δ_j are the Uhrig times (4.36) and $\Delta_0 \coloneqq 0$, $\Delta_{N+1} \coloneqq T$. In [177], QDD was shown numerically to outperform CDD and PDD in state preservation while using fewer pulses. Table 4.1 compares the performance of different DD schemes in terms of the decoupling order and number of pulses.

DD scheme	system	order N	number of pulses L
PDD (m cycles)	1 qubit	$N = 1$	$L = 4m$
CDD	1 qubit	$N \in \mathbb{N}$	$L = 4^N$
QDD	1 qubit	$N \in \mathbb{N}$	$L = (N+1)^2$
NUDD	n qubits	$N \in \mathbb{N}$	$L = (N+1)^{2n}$

Table 4.1: Comparison of different universal DD schemes for (multi-)qubit systems with general noise. Whereas the periodic DD (PDD) scheme can only achieve first order DD using 4 pulses, the concatenated DD (CDD) and quadratic DD (QDD) are able to achieve arbitrary decoupling order. The number of required pulses scales exponentially in the decoupling order for CDD and for QDD sequences it only scales quadratically. For the nested Uhrig DD (NUDD) sequence for n qubits, the scaling is still quadratic in the decoupling order and grows exponentially in the number of qubits.

In addition to investigating the behaviour in terms of the achieved decoupling order, there are also other figures of merit for the efficiency of a DD sequence: The overlap between an initial state and the time-evolved state. This 'actual error' may of course depend on properties of the initial state and of the environment (which enter the asymptotic scaling via the prefactors of non-vanishing higher orders in the Dyson or Magnus series) and the originally-prepared state. The performance of different decoupling schemes has been studied numerically in the literature [127]. An experimental comparison (for trapped ion qubits) between PDD and UDD [21] showed that for Ohmic noise spectra with sharp high-frequency cutoffs, UDD clearly outperforms PDD whereas for sub-Ohmic spectra, the performances were comparable.

Two formal proofs of Nth order DD of the QDD sequence were subsequently given by Wang and Liu [172] as well as by Kuo and Lidar [103]. In 2010, Mukhtar et al. [116, 117] generalised this further to two qubits, i.e., they proposed a pulse sequence deduced from UDD in order to protect any two-qubit state from decoherence.

4.4.3 Nested Uhrig dynamical decoupling

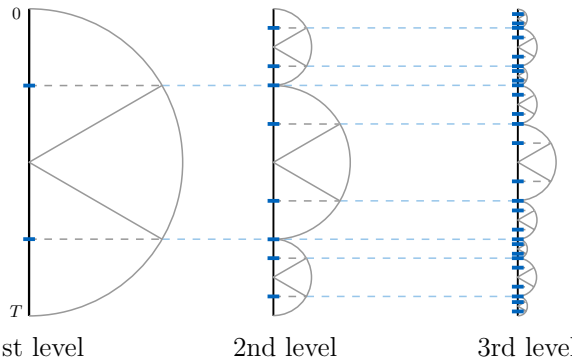


Figure 4.2: Recursive construction of the NUDD times for $N = 2$ with three nesting levels.

Let us present the *Nested Uhrig Dynamical Decoupling* (NUDD) scheme for $n \in \mathbb{N}$ qubits [116, 117, 172, 86]. This scheme can be regarded as the straightforward generalisation of two nesting levels of QDD to $2n$ nesting levels. On the n-qubit Hilbert space $\mathcal{H}_S = (\mathbb{C}^2)^{\otimes n}$, let $\sigma_\alpha = \sigma_{a_1} \otimes \cdots \otimes \sigma_{a_n}$ denote multi-qubit Pauli matrices for $\alpha = (a_1, \ldots, a_n) \in (\mathbb{Z}_2^2)^n$ where

$$\sigma_{(0,0)} = I_2 \ , \qquad \sigma_{(1,0)} = \sigma_x \ , \qquad \sigma_{(1,1)} = \sigma_y \ , \qquad \sigma_{(0,1)} = \sigma_z \ . \tag{4.37}$$

The decoherence Hamiltonian considered is of the form

$$H^{\text{orig}}(t) = \sum_{\alpha \in (\mathbb{Z}_2^2)^n} \sigma_\alpha \otimes B_\alpha(t) \ , \tag{4.38}$$

where the time dependent environment operators $B_\alpha(t) \in \mathcal{B}_{sa}(\mathcal{H}_E)$ for an environment $\mathcal{H}_E$ are assumed to have analytic time-dependence. The NUDD sequence is able to eliminate such decoherence up to Nth order, i.e., it is a universal Nth order DD scheme for n qubits. It consists of $2n$ nesting levels, such that two UDD-type nesting levels are associated with every qubit (similar to QDD). After numerical evidence for its validity, formal proofs were provided by Wang and Liu [172] (for an even but potentially different decoupling order on every nesting level) and Jiang and Imambekov [86] (for the same decoupling order on every nesting level which may be even or odd).

The NUDD pulse times are given by $t_\lambda^{\mathsf{NUDD}} = \Delta_\lambda T$ for $\lambda = (\ell_1, \ell_2, \ldots, \ell_{2n}) \in \{0, \ldots, N\}^{2n}$ where Δ_λ is recursively defined via

$$\Delta_{(\ell_1,\ldots,\ell_k)} \coloneqq \Delta_{\ell_k} + (\Delta_{\ell_k+1} - \Delta_{\ell_k})\Delta_{(\ell_1,\ldots,\ell_{k-1})} \ . \tag{4.39}$$

The recursive construction of the NUDD times is shown in Fig. 4.2 for $N = 2$. Let us first consider the case where the decoupling order N is even. Then the NUDD pulses are

$$U_\lambda^{\mathsf{NUDD}} = \begin{cases} (\sigma_x)_k & \text{if } \lambda = (0, \ldots, 0, \ell_{2k}, \ldots, \ell_{2n}), \text{ where } \ell_{2k} \neq 0 \\ (\sigma_z)_k & \text{if } \lambda = (0, \ldots, 0, \ell_{2k-1}, \ldots, \ell_{2n}), \text{ where } \ell_{2k-1} \neq 0 \ . \end{cases} \tag{4.40}$$

If N is odd, the NUDD sequence slightly changes due to the additional UDD pulse at the end of the cycle (at each level). The pulse times are the same as for even N, i.e., $t_\lambda^{\mathsf{NUDD}} = \Delta_\lambda T$, but there is an additional pulse $\prod_{i=1}^n (\sigma_y)_i$ at time T and the other pulses change to

$$U_\lambda^{\mathsf{NUDD}} = \begin{cases} \prod_{i=1}^k (\sigma_y)_i & \text{if } \lambda = (0, \ldots, 0, \ell_{2k}, \ldots, \ell_{2n}), \text{ where } \ell_{2k} \neq 0 \\ (\sigma_z)_k \prod_{i=1}^{k-1} (\sigma_y)_i & \text{if } \lambda = (0, \ldots, 0, \ell_{2k-1}, \ldots, \ell_{2n}), \text{ where } \ell_{2k-1} \neq 0 \ . \end{cases} \tag{4.41}$$

As a consequence, an NUDD sequence for n qubits of DD order N in every nesting level consists of $L = (N+1)^{2n}$ pulses, more precisely, $(N+1)^{2n} - 1$ if N is even and $(N+1)^{2n}$ if N is odd. The number of pulses scales *quadratically* in the decoupling order and exponentially in the number of qubits.

Fig. 4.3 exemplarily shows the four nesting levels of the two-qubit NUDD sequence of decoupling order $N = 2$. The NUDD sequence has been experimentally demonstrated with three nesting levels for two NMR qubits [146].

4.5 Dynamical decoupling for general matrix Lie groups

This section treats dynamical decoupling in terms of a general matrix Lie group G and its Lie algebra $\mathfrak{g}$. The advantage of the presented framework is that it captures DD in two relevant settings: First, higher order dynamical decoupling in finite-dimensional systems and second, homogenisation of multi-mode bosonic CV systems with quadratic Hamiltonians.

In both cases, the considered matrix Lie group G is associated with evolutions of the system: Firstly, in the case of $m \in \mathbb{N}$ qu*d*its, the system's Hilbert space is $\mathcal{H}_S = \mathbb{C}^{d^m}$ and the considered matrix Lie group is the unitary group $\mathsf{G} = \mathsf{U}(d^m)$. Secondly, in the case of $n_S \in \mathbb{N}$

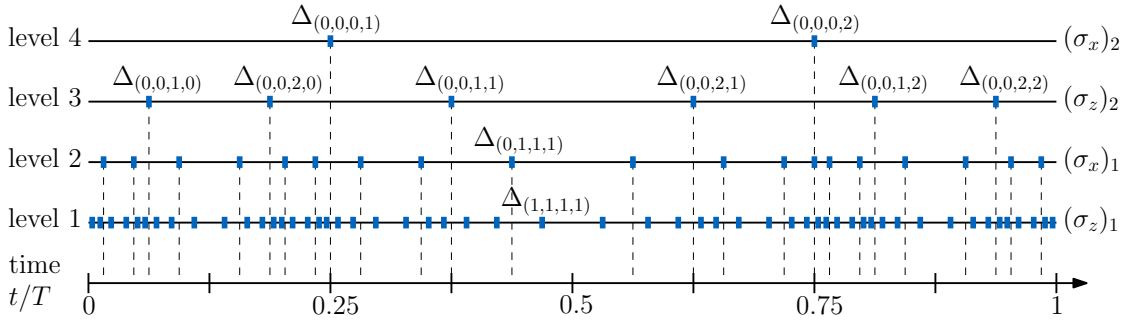


Figure 4.3: The pulse times (in blue) and the pulses (on the right axis) of the four nesting levels of the two-qubit NUDD scheme of decoupling order $N = 2$. In the outermost nesting level, σ_x-pulses are applied to qubit $n = 2$ at the standard UDD times; on the next level, σ_z-pulses are applied to qubit $n = 2$. On the two innermost nesting levels, the applied control pulses are $(\sigma_x)_1$ and $(\sigma_z)_1$, respectively. In total $(2+1)^{2\cdot 2} - 1 = 80$ pulses are used to achieve second order DD.

bosonic modes, the system's phase space is $\mathbb{R}^{2n_S}$, and the Lie group $\mathsf{G} = \mathsf{Sp}(2n_S)$ consists of symplectic matrices which are related to Gaussian unitary evolutions via the metaplectic representation. In slight abuse of notation, we will use the term „system's Hilbert space“ even in the bosonic setting when we actually mean the phase space $\mathcal{H}_S = \mathbb{R}^{2n_S}$ of n_S bosonic modes.

We start with Section 4.5.1 where we give mathematically precise definitions of decoherence and pulse sequences in the context of general matrix Lie groups. One central assumption concerns the commutation relations between pulses and decoherence terms. In Section 4.5.2, we derive sufficient conditions for the resulting evolution – decoherence interleaved with instantaneous pulses – to act trivially on the system (Theorem 4.14) and to reduce to a specific simplified system action (Corollary 4.15), in both cases approximately up to a certain order in time. These conditions are formulated in terms of integral equations which have to be satisfied for the respective pulse sequences. We note that this (together with the relation between qubit and bosonic systems from Section 4.7) will be key to our construction of Nth order bosonic homogenisation schemes from Nth order multi-qubit DD schemes in Section 4.8.2. The framework that we derive here is closely related to and generalises the universality proof of NUDD by Jiang and Imambekov [86]. We state that the *decoupling criterion* in Theorem 4.14 holds for the NUDD pulse sequence (a result proven in [86]) as well as for its simplification to the UDD pulse sequence in Section 4.5.3.

4.5.1 Setup

Let $\mathsf{G} \subset \mathsf{GL}(\mathbb{K}^{d_S})$ be a matrix Lie group where $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ and $d_S \in \mathbb{N}$. Let $\mathfrak{g}$ be the Lie algebra associated with G and let $\{Y_\alpha\}_{\alpha \in A}$ be a basis of $\mathfrak{g}$. We will use upper case letters X, Y, Z to denote Lie algebra elements, i.e., elements of $\mathfrak{g}$, and lower case letters x, y, z to denote Lie group elements, i.e., elements of G. Set $\mathcal{H}_S = \mathbb{K}^{d_S}$.

Let us fix a family of group elements $\{x_\beta\}_{\beta \in B} \subset \mathsf{G}$ which satisfies two assumptions: First, let each of the chosen group elements have an infinitesimal generator in $\mathfrak{g}$: for each $\beta \in B$, there is $\varphi \in \mathbb{K}$ with $|\varphi| = 1$ such that

$$x_\beta = \varphi\, e^{X_\beta} \qquad \text{for each } \beta \in B\ .$$

Second, let the group elements act diagonally by conjugation in the chosen basis of $\mathfrak{g}$. That is, there is a function $f : A \times B \to [0,1]$ such that

$$\mathsf{Ad}(x_\beta)(Y_\alpha) = (-1)^{f(\alpha,\beta)} Y_\alpha \qquad \text{for all } (\alpha,\beta) \in A \times B \ . \tag{4.42}$$

Here, Ad denotes the adjoint representation of the Lie group G which is defined as

$$\mathsf{Ad} : \mathsf{G} \to \mathsf{GL}(\mathfrak{g}) \quad , \quad \mathsf{Ad}(x)(Y) = x^{-1} Y x \quad \text{for all } x \in G \text{ and } Y \in \mathfrak{g} \ .$$

In the cases of interest, the function f in the exponent in Eq. (4.42) is the symplectic inner product modulo 2, i.e., $f(\alpha,\beta) = \langle \alpha, \beta \rangle$ for

$$\langle \alpha, \beta \rangle \coloneqq (\alpha^T J_{d_s} \beta) \mod 2 \ . \tag{4.43}$$

Consider the Lie algebra $\mathcal{B}(\mathcal{H}_E)$ of bounded operators acting on an environment Hilbert space $\mathcal{H}_E$. Let $T > 0$ denote the time after which DD is analysed. Decoherence is described by a time-dependent generator $X^{\text{orig}} \colon [0,T] \to \mathfrak{g} \otimes \mathcal{B}(\mathcal{H}_E)$ involving system-bath interactions. We will make a few assumptions on this generator. First, let X^{orig} be of the form

$$X^{\text{orig}}(t) = \sum_{\alpha \in A} Y_\alpha \otimes B_\alpha(t) \tag{4.44}$$

where Y_α are elements of the above defined basis of $\mathfrak{g}$ and $B_\alpha(t) \in \mathcal{B}(\mathcal{H}_E)$. Furthermore, we assume that the environment operators $B_\alpha(t)$ have analytic time dependence (similar to Definition 4.6), i.e., they satisfy

$$B_\alpha(t) = \sum_{r=0}^{\infty} b_{\alpha,r} t^r \qquad \text{where } b_{\alpha,r} \in \mathcal{B}(\mathcal{H}_E) \ . \tag{4.45}$$

Let $x^{\text{orig}} : [0,T] \to \mathsf{GL}(\mathcal{H}_S \otimes \mathcal{H}_E)$ be the time evolution generated by X^{orig}. More precisely, x^{orig} is defined by the equation

$$\frac{d}{dt} x^{\text{orig}}(t) = X^{\text{orig}}(t) x^{\text{orig}}(t) \quad \text{for } t \in (0,T) \quad \text{and} \quad x^{\text{orig}}(0) = I_S \otimes I_E \ ,$$

where I_S denotes the identity on the system S and I_E the environment E. In contrast to x^{orig}, the control evolution $x^{\text{control}} : [0,T] \to \mathsf{GL}(\mathcal{H}_S \otimes \mathcal{H}_E)$ acts non-trivially on the system only and is described by a sequence of instantaneously applied pulses. Let us specify the definition of a pulse sequence for the setup considered here.

Definition 4.10 *(Pulse sequence).* Let G, $\mathcal{H}_S$, B and $\{x_\alpha\}_{\alpha \in B}$ be defined as described above. A *pulse sequence* is defined by a tuple

$$\big(\{t_\lambda\}_{\lambda \in \Lambda}, \{x_{\beta(\lambda)}\}_{\lambda \in \Lambda} \big) \ ,$$

such that the following assumptions are satisfied: $\Lambda = \{\lambda_1, \ldots, \lambda_{|\Lambda|}\}$ is a finite set such that the pulse times $t_\lambda \in [0,T]$ satisfy $t_{\lambda_1} < t_{\lambda_2} < \cdots < t_{\lambda_{|\Lambda|}}$ and the map

$$\beta : \Lambda \mapsto B \qquad , \qquad \lambda \mapsto \beta(\lambda) \tag{4.46}$$

specifies which pulse is applied at which time. Furthermore, assume that $\prod_{j=1}^{|\Lambda|} x_{\beta(\lambda_j)} = I_S$ where I_S is the identity on $\mathcal{H}_S$.

Although the use of the function β might seem unnecessary at this point, it will prove useful later when considering the UDD and NUDD sequences. As a consequence, the control evolution $x^{\text{control}}(t)$ of the pulse sequence from Definition 4.10 is given by the product of all pulses applied up to time t, i.e.,

$$x^{\text{control}}(t) = \prod_{j=1}^{K_t} x_{\beta(\lambda_j)} \otimes I_E \ , \tag{4.47}$$

where $K_t \in \{1, \dots, |\Lambda|\}$ is the maximal index of pulses applied up to time t, cf. Eq. (4.24). Since every x_β has an infinitesimal generator by assumption, there is $X_{\beta(\lambda)} \in \mathfrak{g} \otimes \mathfrak{gl}(\mathcal{H}_E)$ such that $x_{\beta(\lambda)} = e^{X_{\beta(\lambda)}}$ up to a phase. Hence the evolution $x^{\text{control}}(t)$ is generated by

$$X^{\text{control}} \colon [0,T] \to \mathfrak{g} \otimes \mathfrak{gl}(\mathcal{H}_E) \quad , \quad X^{\text{control}}(t) = \sum_{j=1}^{K_t} \delta\big(t - t_{\lambda_j}\big) X_{\beta(\lambda_j)} \otimes I_E \ .$$

We will analyse the evolution obtained after time T, where the uncontrolled original evolution $x^{\text{orig}}(t)$ is interleaved with the instantaneous application of the pulses $x_{\beta(\lambda)}$ at times t_λ for every $\lambda \in \Lambda$. More precisely, the resulting evolution

$$x^{\text{res}} \colon [0,T] \to \mathsf{GL}(\mathcal{H}_S \otimes \mathcal{H}_E)$$

is generated by $X^{\text{res}} \coloneqq X^{\text{orig}} + X^{\text{control}}$.

To easily check all the assumptions, let us briefly list them here.

Definition 4.11 *(Setup).* Let G, $\mathfrak{g}$, and $\mathcal{H}_E$ be as defined above. An an evolution x^{orig} and a pulse sequence fall into the setup discussed here if they satisfy:

(i) There exists a basis $\{Y_\alpha\}_{\alpha \in A}$ of $\mathfrak{g}$ for some set A and a family of infinitesimally generated elements $\{x_\beta\}_{\beta \in B}$ of G for some set B such that Eq. (4.42) is satisfied;

(ii) The control pulses are chosen from the above family $\{x_\beta\}_{\beta \in B}$, i.e., the pulse sequence is given by a function $\beta : \Lambda \to B$ (cf. Definition 4.10);

(iii) The generator $X^{\text{orig}}(t) \in \mathfrak{g} \otimes \mathcal{B}(\mathcal{H}_E)$ of x^{orig} has the form of Eq. (4.44), i.e., it can be expanded in terms of the basis elements Y_α and some time-dependent operators $B_\alpha(t)$ on the environment which have analytic expansions of the form of Eq. (4.45).

Let us briefly justify why it is sufficient to only consider bounded operators on the environment $\mathcal{H}_E$. The two cases to which this setup is applied are those of multi-qubit DD (cf. Definition 3.9) and multi-mode bosonic homogenisation (cf. Definition 4.9). If the Hilbert space is finite-dimensional (which one might assume for the environment of a multi-qubit system), all self-adjoint operators are bounded. In the latter case, bosonic homogenisation, one instead deals with systems and environments that are both infinite-dimensional and on which the considered quadratic Hamiltonians are unbounded. However, recall that we conduct the whole analysis at the level of symplectic matrices: Moreover, in the case of CV homogenisation, we set $\mathcal{H}_S = \mathbb{R}^{2n_S}$ to be the n_S-mode phase space and $\mathcal{H}_E = \mathbb{C}$ to be trivial, both are finite-dimensional vector spaces. Setting $\mathcal{H}_E = \mathbb{C}$ is a suitable choice since in the context of homogenisation we *can neglect* the 'physical environment' in our analysis since we assume that the system is already decoupled from 'its physical environment $L^2(\mathbb{R}^{2n_E})$' (cf. Definition 4.9) such that the whole analysis restricts to the system part only[2].

[2] In fact, the composition of the system and its environment $L^2(\mathbb{R}^{2n_E})$ at the symplectic level cannot be described by a tensor product but instead by block matrices or direct sums.

4.5.2 Sufficient decoupling criteria

We now want to give conditions on the setup of Definition 4.11 such that $x^{\mathrm{res}}(T)$ acts trivially on the system, which would imply DD of $n \in \mathbb{N}$ qubits in the case of $\mathsf{G} = \mathsf{U}((\mathbb{C}^2)^{\otimes n})$, and that $x^{\mathrm{res}}(T)$ is reduced to a non-trivial but simplified form (which is related to bosonic homogenisation from Section 4.3.3).

As for finite-dimensional DD (cf. Section 3.3.2), it is convenient to change into the interaction picture associated with the control evolution – *the toggling frame* – and expand the respective time evolution in a Dyson series. Let us recall that the toggling frame evolution

$$x^{\mathrm{tf}} \colon [0,T] \to \mathsf{GL}(\mathcal{H}_S \otimes \mathcal{H}_E)$$

is defined as

$$x^{\mathrm{tf}}(t) = \big(x^{\mathrm{control}}(t)\big)^{-1} x^{\mathrm{res}}(t) \qquad \text{for all } t \in [0,T] \ .$$

We note that, due to $x^{\mathrm{control}}(T) = I_S$, the toggling frame evolution and the resulting evolution are equal up to a global phase after time T. Hence, $x^{\mathrm{res}}(T) = x^{\mathrm{tf}}(T)$ where the latter is expanded in a Dyson series in orders of N using the two following two lemmas.

Lemma 4.12 *(Toggling frame generator).* Consider a pulse sequence and an original evolution that fall into the setup of Definition 4.11.

Then the toggling frame generator is given by

$$X^{\mathrm{tf}}(t) = \sum_{\alpha \in A} F_\alpha\left(t/T\right) Y_\alpha \otimes B_\alpha(t) \tag{4.48}$$

where for every for $\alpha \in A$, the function $F_\alpha \colon [0,1] \to \{-1,1\}$ is defined by

$$F_\alpha(\tau) \coloneqq (-1)^{\sum_{j=1}^{K_t} \langle \alpha, \beta(\lambda_j)\rangle} \qquad \text{for } \tau = t/T \in [0,1] \ .$$

Here, K_t is given by Eq. (4.24) and the expression $\langle \cdot, \cdot \rangle$ by Eq. (4.43). □

In the following, we sometimes denote the functions by $F_{\alpha,\beta} = F_\alpha$ to emphasise its dependence on the map β. We note that the expressions 4.48 have previously been derived for the case $\mathsf{G} = \mathsf{U}((\mathbb{C}^2)^{\otimes n})$ by Liang and Imambekov [86]. Let us show that they also hold for the more general setup discussed here.

Proof. Consider the setup from the previous section. The toggling frame generator is of the form

$$X^{\mathrm{tf}}(t) = x^{\mathrm{control}}(t)^{-1} X^{\mathrm{orig}}(t) x^{\mathrm{control}}(t) \ .$$

We note that it is therefore independent of the phases in $x_{\beta(\lambda)}$ and $x^{\mathrm{control}}(t)$ which we will from now on neglect. Let us insert $x^{\mathrm{control}}(t)$ from Eq. (4.47) and $X^{\mathrm{orig}}(t)$ from Eq. (4.44) into $X^{\mathrm{tf}}(t)$ which gives

$$X^{\mathrm{tf}}(t) = \sum_{\alpha \in A} \left(\prod_{j=1}^{K_t} x_{\beta(\lambda_j)} \right)^{-1} Y_\alpha \left(\prod_{j=1}^{K_t} x_{\beta(\lambda_j)} \right) \otimes B_\alpha(t) \ .$$

Using the adjoint representation from Eq. (4.42), this is

$$\begin{aligned} X^{\mathrm{tf}}(t) &= \sum_{\alpha\in A} \Big((-1)^{\langle \alpha,\beta(\lambda_1)\rangle} (-1)^{\langle \alpha,\beta(\lambda_2)\rangle} \cdots (-1)^{\langle \alpha,\beta(\lambda_{K_t})\rangle}\ Y_\alpha \Big) \otimes B_\alpha(t) \\ &= \sum_{\alpha\in A} (-1)^{\sum_{j=1}^{K_t}\langle \alpha,\beta(\lambda_j)\rangle}\ Y_\alpha \otimes B_\alpha(t)\ . \end{aligned}$$

This is equal to the desired expression (4.48). ■

The expressions 4.48 can be used to compute the toggling frame evolution. Here, again Liang and Imambekov [86] derived the formulas for the case $\mathsf{G} = \mathsf{U}((\mathbb{C}^2)^{\otimes n})$ which we extend to the more general setup in the following Lemma.

Lemma 4.13 *(Dyson expansion of toggling frame evolution).* Consider the same assumptions as in Lemma 4.12. The Dyson series of the toggling frame evolution after time T is given by

$$x^{\mathrm{tf}}(T) = \sum_{k=0}^{\infty} \sum_{\vec{\alpha}\in A^k} \sum_{\vec{r}\in\mathbb{N}_0^k} \mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)} \left(\prod_{m=1}^{k} Y_{\alpha_m} \right) \otimes \left(\prod_{m=1}^{k} b_{\alpha_m,r_m} \right) T^{k+\sum_{l=1}^{k} r_l}\ , \tag{4.50}$$

where we defined the multi-indices $\vec{\alpha} \coloneqq (\alpha_1,\ldots,\alpha_k) \in A^k$ and $\vec{r} \coloneqq (r_1,\ldots,r_k) \in \mathbb{N}_0^k$ and where the scalars $\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}$ are defined as

$$\mathcal{F}^{\vec{r}}_{\vec{\alpha}} = \mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)} \coloneqq \int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} \prod_{l=1}^{k} \Big(F_{\alpha_m}(\tau_m) \tau_m^{r_m} \Big) d\tau_k \cdots d\tau_2 d\tau_1 \tag{4.51}$$

for $k \in \mathbb{N}$ and where the functions $F_{\alpha,\beta}$ are given by Eq. (4.49). □

We will sometimes write

$$\mathcal{F}^{\vec{r}}_{\vec{\alpha}} = \mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)} \left(\{F_{\alpha_m,\beta}\}_{m=1}^{k} \right)$$

to emphasise the dependence of the scalars on the functions $F_{\alpha,\beta}$.

Proof. The toggling frame evolution $x^{\mathrm{tf}}(T)$ generated by (4.48) can be expanded in a Dyson series since A is finite and

$$\max_{t\in[0,T]} \|X^{\mathrm{tf}}(t)\| = \max_{t\in[0,T]} \| \sum_{\alpha\in A} F_\alpha(t/T) Y_\alpha \otimes B_\alpha(t)\| \leq \max_{t\in[0,T]} \sum_{\alpha\in A} \|Y_\alpha\| \|B_\alpha(t)\| < \infty\ .$$

The Dyson series (cf. Definition 3.7) of the toggling frame evolution is defined as

$$x^{\mathrm{tf}}(T) = \sum_{k=0}^{\infty} \int_0^T \int_0^{s_1} \cdots \int_0^{s_{k-1}} X^{\mathrm{tf}}(s_1) X^{\mathrm{tf}}(s_2) \cdots X^{\mathrm{tf}}(s_k) ds_k \ldots ds_2 ds_1\ .$$

A substitution of the integration variables and insertion of the toggling frame generator (4.48)

from Lemma 4.12 yields

$$\begin{aligned}
x^{\mathrm{tf}}(T) &= \sum_{k=0}^{\infty} T^k \int_0^1 \int_0^{\tau_1} \dots \int_0^{\tau_{k-1}} X^{\mathrm{tf}}(\tau_1 T) X^{\mathrm{tf}}(\tau_2 T) \dots X^{\mathrm{tf}}(\tau_k T)\, d\tau_k \dots d\tau_2 d\tau_1 \\
&= \sum_{k=0}^{\infty} T^k \int_0^1 \int_0^{\tau_1} \dots \int_0^{\tau_{k-1}} \Big(\sum_{\alpha_1 \in A} F_{\alpha_1}(\tau_1) Y_{\alpha_1} \otimes B_{\alpha_1}(\tau_1 T) \Big) \cdot \\
&\qquad \Big(\sum_{\alpha_2 \in A} F_{\alpha_2}(\tau_2) Y_{\alpha_2} \otimes B_{\alpha_2}(\tau_2 T) \Big) \cdots \Big(\sum_{\alpha_k \in A} F_{\alpha_k}(\tau_k) Y_{\alpha_k} \otimes B_{\alpha_k}(\tau_k T) \Big) d\tau_k \dots d\tau_2 d\tau_1 \\
&= \sum_{k=0}^{\infty} \sum_{\vec{\alpha} \in A^k} T^k \int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} \Big(\prod_{m=1}^{k} F_{\alpha_m}(\tau_m) Y_{\alpha_m} \Big) \otimes \Big(\prod_{m=1}^{k} B_{\alpha_m}(\tau_m T) \Big) d\tau_k \dots d\tau_2 d\tau_1 \ ,
\end{aligned}$$

where in the last line, the multi-index $\vec{\alpha} \coloneqq (\alpha_1, \dots, \alpha_k) \in A^k$ for $k \in \mathbb{N}$ is introduced. Let us further compute the integrand, namely the term

$$\prod_{m=1}^{k} B_{\alpha_m}(\tau_m T) = \prod_{m=1}^{k} \left(\sum_{r_m=0}^{\infty} b_{\alpha_m, r_m} T^{r_m} \tau_m^{r_m} \right) = \sum_{\vec{r} \in \mathbb{N}_0^k} \left(\prod_{m=1}^{k} b_{\alpha_m, r_m} \tau_m^{r_m} T^{r_m} \right) \ .$$

Here, we use the analytic expansion of the environment operators from Eq. (4.45) and introduce the multi-index $\vec{r} \coloneqq (r_1, r_2, \dots, r_k) \in \mathbb{N}_0^k$. Inserting this into the Dyson series of $x^{\mathrm{tf}}(T)$ becomes

$$\begin{aligned}
x^{\mathrm{tf}}(T) &= \sum_{k=0}^{\infty} \sum_{\vec{\alpha} \in A^k} T^k \int_0^1 \int_0^{\tau_1} \dots \int_0^{\tau_{k-1}} \left(\prod_{m=1}^{k} F_{\alpha_m}(\tau_m) Y_{\alpha_m} \right) \cdot \\
&\qquad \otimes \sum_{\vec{r} \in \mathbb{N}_0^k} \left(\prod_{m=1}^{k} b_{\alpha_m, r_m} \tau_m^{r_m} T^{r_m} \right) \, d\tau_k \dots d\tau_2 d\tau_1 \\
&= \sum_{k=0}^{\infty} \sum_{\vec{\alpha} \in A^k} \sum_{\vec{r} \in \mathbb{N}_0^k} T^{k + r_1 + r_2 + \dots + r_k} \left(\prod_{m=1}^{k} Y_{\alpha_m} \right) \otimes \left(\prod_{m=1}^{k} b_{\alpha_m, r_m} \right) \\
&\qquad \cdot \int_0^1 \int_0^{\tau_1} \dots \int_0^{\tau_{k-1}} \left(\prod_{m=1}^{k} F_{\alpha_m}(\tau_m) \tau_m^{r_m} \right) \, d\tau_k \dots d\tau_2 d\tau_1 \\
&= \sum_{k=0}^{\infty} \sum_{\vec{\alpha} \in A^k} \sum_{\vec{r} \in \mathbb{N}_0^k} T^{k + \sum_{m=1}^{k} r_m} \left(\prod_{m=1}^{k} Y_{\alpha_m} \right) \otimes \left(\prod_{m=1}^{k} b_{\alpha_m, r_m} \right) \mathcal{F}^{(r_1, \dots, r_k)}_{(\alpha_1, \dots, \alpha_k)}(\{F_{\alpha_m, \beta}\}_{m=1}^{k}) \ ,
\end{aligned}$$

where $\mathcal{F}^{(r_1, \dots, r_k)}_{(\alpha_1, \dots, \alpha_k)}(\{F_{\alpha_m, \beta}\}_{m=1}^{k})$ is given by Eq. (4.51). ■

Eq. (4.50) shows the Dyson expansion of the toggling frame evolution $x^{\mathrm{tf}}(T)$. One can directly read off the Nth order term in T – more precisely, the term of order $k + \sum_{m=1}^{k} r_m$ in T. It acts trivially on the system if for every tuple $(k, \vec{r}) \in \mathbb{N} \times \mathbb{N}_0^k$ such that $N = k + \sum_{m=1}^{k} r_m$ one of the two following conditions is satisfied:

(i) either the product $\prod_{m=1}^{k} Y_{\alpha_m}$ amounts to the identity (up to a complex phase), i.e, there is a $c \in \mathbb{K}$ such that $Y_{\alpha_k} \cdots Y_{\alpha_1} = c I_S$ for all $(\alpha_1, \dots, \alpha_k) \in A^k$);

(ii) or the scalars $\mathcal{F}^{(r_1, \dots, r_k)}_{(\alpha_1, \dots, \alpha_k)}$ vanish for all $(\alpha_1, \dots, \alpha_k) \in A^k$.

To ensure that all terms up to order N in T of the Dyson series vanish, it is necessary that $\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}$ vanishes for the parameters for which the first statement (i) does not hold. This gives the following statement (cf. [86, 103]).

Theorem 4.14 *(Decoupling criterion [86])***.** Consider the same assumptions as in Lemma 4.12. Let $N \in \mathbb{N}$ and let $\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}$ be defined as in (4.51). Assume that

$$\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}\left(\{F_{\alpha_m,\beta}\}_{m=1}^k\right) = 0 \quad \begin{cases} \text{for all } k \in \mathbb{N},\ r_1,\ldots,r_k \in \mathbb{N}_0, \text{ and } \alpha_1,\ldots,\alpha_k \in A \\ \text{such that } k + r_1 + \ldots + r_k \leq N,\ Y_{\alpha_k}\cdots Y_{\alpha_1} \not\propto I_S \end{cases} \tag{4.52}$$

where we write $B \not\propto I_S$ if B is not a scalar multiple of the identity I_S.

Then there is a constant $C \geq 0$ independent of N and an operator $\widetilde{B} \in \mathcal{B}(\mathcal{H}_E)$ such that

$$\|x^{\mathrm{res}}(T) - I_S \otimes \widetilde{B}\| \leq C\, T^{N+1}\ . \tag{4.53}$$

where $\|\cdot\|$ denotes a norm on $\mathcal{B}(\mathcal{H}_S \otimes \mathcal{H}_E)$. □

Proof. This is a direct application of Lemma 4.13: The toggling frame evolution after time T is given by Eq. (4.50). From this equation, one can directly read off the Nth order term in T as described above. Eq. (4.52) states a condition under which $x^{\mathrm{tf}}(T)$ acts trivially on the system up to order N in T. Since $x^{\mathrm{tf}}(T) = x^{\mathrm{res}}(T)$, the same holds for $x^{\mathrm{res}}(T)$. ■

Let us add two remarks: First we note that the constant C in Eq. (4.53) depends on the norm of the original environment operators $B_\alpha(t)$. Second, we emphasise that the scalars $\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}$ depend on the family of functions $\{F_{\alpha_m,\beta}\}_{m=1}^k$ from Eq. (4.49). The latter functions are defined by the pulse times t_λ and the map β from (4.46) that specifies which pulse $x_{\beta(\lambda)}$ is applied at time t_λ. As a consequence, the scalars $\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}$ fully capture the decoupling properties of the associated pulse sequence by the statement (4.52).

In some cases – such as homogenisation of bosonic modes (cf. Section 4.3.3) – the strong form of decoupling as in Eq. (4.53) may be achievable, but a weaker form is still beneficial for information processing tasks. That is why we define a weaker form of Eq. (4.53): Here, the resulting evolution may act non-trivially on the system but it is reduced to a particular form, specified by a single basis element $Y \in \mathfrak{g}$, up to order N in T.

Corollary 4.15 *(Modified decoupling/homogenisation criterion)***.** Consider the same assumptions as in Lemma 4.12. Let $N \in \mathbb{N}$ and $\gamma \in A$ be fixed. Assume that

$$\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}\left(\{F_{\alpha_m,\beta}\}_{m=1}^k\right) = 0 \quad \begin{cases} \text{for all } k \in \mathbb{N}, r_1,\ldots,r_k \in \mathbb{N}_0, \alpha_1,\ldots,\alpha_k \in A \\ \text{such that } k + r_1 + \ldots + r_k \leq N,\ Y_{\alpha_k}\cdots Y_{\alpha_1} \not\propto \{I_S, Y_\gamma\}. \end{cases}$$

Then there are a constant $C \geq 0$ and operators $\tilde{B}_1, \tilde{B}_2 \in \mathcal{B}(\mathcal{H}_E)$ such that

$$\|x^{\mathrm{res}}(T) - I_S \otimes \tilde{B}_1 - Y_\gamma \otimes \tilde{B}_2\| \leq C\, T^{N+1}\ .$$

□

Proof. The proof is analogous to the one of Theorem 4.14. From Eq. (4.50), one can again read off the Nth order term in T. ■

Note that in Corollary 4.15, we specify an element $Y \in \mathfrak{g}$ by the index $\gamma \in A$, i.e., we write Y_γ. This (at first glance cumbersome) notation will prove useful later when considering the NUDD pulse sequence.

4.5.3 Examples of decoupling achieving pulse sequences

Let us stay in the above framework and present two functions $\{F_{\alpha,\beta}\}_{\alpha\in A}$ that satisfy the conditions of Theorem 4.14: the functions associated with the UDD and the NUDD sequence, both of them are presented in Sections 4.4.2 and 4.4.3, respectively.

NUDD sequence functions

A first example concerns the NUDD pulse sequence. Let us shows that it falls into the framework defined above and that its associated functions $F_{\alpha,\beta}$ satisfy Eq. (4.52). Note that this result has been shown before since the whole above analysis (DD on general matrix Lie groups) is itself a generalisation of the NUDD universality proof by Jiang and Imambekov [86].

We make a little adaption compared to the notation from Section 4.4.3 (where we considered the NUDD pulse sequence): here, we consider NUDD for $m+1$ qubits which are labelled from qubit 0 to qubit m. In this new notation, the pulses are applied at times $t_\lambda^{\mathsf{NUDD}} = \Delta_\lambda T$ with Δ_λ from Eq. (4.39) for

$$\lambda \in \Lambda \coloneqq \{0, 1, \ldots, N\}^{2(m+1)} \ . \tag{4.54}$$

As described in Section 4.4.3, the decoherence is described by the Hamiltonian H^{orig} from Eq. (4.38) for $n = m+1$. Here, the multi-qubit Pauli operators

$$\sigma_\alpha = \sigma_{a_0} \otimes \cdots \otimes \sigma_{a_m} \quad \text{for} \quad \alpha \in (\mathbb{Z}_2^2)^{m+1} \tag{4.55}$$

depend on the single-qubit Pauli matrices σ_{α_k} from Eqs. (4.37). Furthermore, let the environment operators $B_\alpha(t)$ have analytic series expansions that satisfy Eq. (4.45). Hence the Hamiltonian $H^{\mathrm{orig}}(t)$ falls into the framework of Definition 4.11 defined above. Let $\mathfrak{g} = \mathfrak{u}(2^{m+1})$ be the Lie algebra of the unitary group with basis $\{\sigma_\alpha\}_{\alpha\in(\mathbb{Z}_2^2)^{m+1}}$ and let $\mathsf{G} \coloneqq \mathsf{U}(2^{m+1})$ be the associated Lie group of unitaries on $(\mathbb{C}^2)^{\otimes(m+1)}$. The adjoint action of G is given by

$$\sigma_\beta^{-1}\sigma_\alpha\sigma_\beta = (-1)^{\langle\alpha,\beta\rangle}\sigma_\alpha \qquad \text{for } \alpha,\beta \in (\mathbb{Z}_2^2)^{m+1} \ , \tag{4.56}$$

i.e., it satisfies Eq. (4.42) where the function

$$f(\alpha,\beta) = \langle\alpha,\beta\rangle = \sum_{j=0}^{m} a_j^T J b_j$$

is the symplectic inner product between the indices $\alpha, \beta \in (\mathbb{Z}_2^2)^{m+1}$.

The NUDD pulses are given by Eq. (4.40) for N even and by Eq. (4.41) for N odd, i.e., they are multi-qubit Pauli-matrices $\sigma_{\beta^{\mathsf{NUDD}}(\lambda)}$ where for Λ from (4.54) the map $\beta^{\mathsf{NUDD}} : \Lambda \to (\mathbb{Z}_2^2)^{m+1}$ is defined as follows: For N even it is given by

$$\beta^{\mathsf{NUDD}}(\lambda) = \begin{cases} (\underbrace{0,\ldots,0}_{2j},1,0,0,\ldots,0) & \text{if } \lambda = (0,\ldots,0,\ell_{2j+1},\ldots,\ell_{2m+1}), \text{ where } \ell_{2j+1} \neq 0 \\ (\underbrace{0,0,\ldots,0}_{2j+1},1,0,\ldots,0) & \text{if } \lambda = (0,\ldots,0,\ell_{2j},\ldots,\ell_{2m+1}), \text{ where } \ell_{2j} \neq 0 \ . \end{cases}$$

Since the pulses for N odd are defined slightly differently, we have

$$\beta^{\mathsf{NUDD}}(\lambda) = \begin{cases} (\underbrace{1,1,\ldots,1}_{2j+1},1,0,\ldots,0) & \text{if } \lambda = (0,\ldots,0,\ell_{2j+1},\ldots,\ell_{2m+1}),\ \text{where } \ell_{2j+1} \neq 0 \\ (\underbrace{1,\ldots,1}_{2j},0,1,0,\ldots,0) & \text{if } \lambda = (0,\ldots,0,\ell_{2j},\ldots,\ell_{2m+1}),\ \text{where } \ell_{2j} \neq 0 \end{cases}$$

in this case. In summary, the NUDD sequence falls into the framework of Definition 4.11 and its DD properties are fully captured by the associated functions $F_{\alpha,\beta^{\mathsf{NUDD}}}$ from Eq. (4.49), more precisely by the scalars $\mathcal{F}_{\vec{\alpha}}^{\vec{r}}(\{F_{\alpha_m,\beta^{\mathsf{NUDD}}}\}_{m=1}^k)$ for $\alpha_1,\ldots,\alpha_k \in (\mathbb{Z}_2^2)^{m+1}$ from Eq. (4.51).

For the map β^{NUDD}, a direct computation shows that

$$\sum_{\mu=1}^{K_t} \langle \alpha, \beta^{\mathsf{NUDD}}(\mu_j)\rangle = \alpha \cdot \lambda \qquad \text{for } \tau \in [\Delta_\lambda, \Delta_{\lambda^+}) \ .$$

Here, K_t is from Eq. (4.24), the label

$$\lambda_+ \coloneqq \min\{\mu \in \Lambda \mid \Delta_\mu > \Delta_\lambda\} \tag{4.57}$$

is used to denote the pulse that follows the pulse with label λ and

$$\alpha \cdot \lambda \coloneqq \sum_{j=0}^{m} a_j \cdot \begin{pmatrix} \ell_{2j} \\ \ell_{2j+1} \end{pmatrix}$$

is the scalar product between the indices α and λ. Together with the following lemma, this implies universality of the NUDD sequence.

Lemma 4.16 *(Properties of the NUDD pulse times [86])*. Let $N \in \mathbb{N}$ and $\Lambda = \{0,\ldots,N\}^{2m+2}$. Consider the NUDD times Δ_λ from Eq. (4.39) for $\lambda \in \Lambda$. For $\alpha \in (\mathbb{Z}_2^2)^{m+1} =: A$ define the piecewise constant functions

$$\begin{aligned} &F_\alpha^{\mathsf{NUDD}} : [0,1] \to \{-1,1\} \ , \\ &F_\alpha^{\mathsf{NUDD}}(\tau) = (-1)^{\alpha\cdot\lambda} \qquad \text{for} \quad \tau \in [\Delta_\lambda, \Delta_{\lambda^+}) \ , \end{aligned}$$

where λ_+ is defined in Eq. (4.57). Then

$$\mathcal{F}_{\vec{\alpha}}^{\vec{r}}\left(\{F_{\alpha_i}^{\mathsf{NUDD}}\}_{i=1}^k\right) = 0 \ \begin{cases} \text{for all } k \in \mathbb{N},\ \vec{r} \in \mathbb{N}_0^k,\ \text{and } \vec{\alpha} \in A^k \text{ such that} \\ k + r_1 + \ldots + r_k \leq N \text{ and } \bigoplus_{i=1}^k \alpha_i \neq \big((0,0),\ldots,(0,0)\big) \end{cases}$$

where for $k \in \mathbb{N}$, $\vec{r} \coloneqq (r_1,\ldots,r_k)$, $\vec{\alpha} \coloneqq (\alpha_1,\ldots,\alpha_k)$ and $\oplus$ denotes addition modulo two. □

Note that $F_\alpha^{\mathsf{NUDD}} = F_{\alpha,\beta^{\mathsf{NUDD}}}$ which we write to shorten notation. For the proof of Lemma 4.16, we refer to the original work by Jiang and Imambekov [86]. The basic idea is to represent the recursive integrations over τ_1 to τ_k in $\mathcal{F}_{(\alpha_1,\ldots,\alpha_k)}^{(r_1,\ldots,r_k)}(\{F_{\alpha_i}^{\mathsf{NUDD}}\}_{i=1}^k)$ as a discrete quantum walk on a functional space where one chooses the basis of the latter such that it can suitably represent the piecewise constant functions F_α^{NUDD}.

Lemma 4.16 implies that the NUDD sequence of order N in every of the $2(m+1)$ nesting level achieves Nth order DD. Furthermore, it implies that any pulse sequence which is associated with the functions F_α^{NUDD} for $\alpha \in (\mathbb{Z}_2^2)^{m+1}$ achieves Nth DD in the sense of Theorem 4.14.

UDD sequence functions

The second family of functions $F_{\alpha,\beta}$ that we present here is associated with the UDD sequence. As discussed in Section 4.4.2, it can be regarded as a special case of the NUDD sequence with a single nesting level, i.e., with only σ_x-pulses.

Here, the two sets from the general setup in Definition 4.11 are given by $A = \{(0,0),(0,1)\}$ and $B = \{(1,0)\}$, i.e., we consider pure dephasing noise and only $\sigma_{(1,0)} = \sigma_x$ pulses for the Pauli matrices from Eq. (4.37). Recall that the UDD times $t_j^{\mathsf{UDD}} = \Delta_j T$ for $j = 0,1,\ldots,N$ are given by Eq. (4.36). Using again the qubit commutation relations (4.56), one finds for the UDD sequence that $\beta^{\mathsf{UDD}}(j) = (1,0)$ for all $j = 1,\ldots,N$, and that $\beta^{\mathsf{UDD}}(0) = (0,0)$. Hence

$$(-1)^{\langle \sum_{i=1}^{K_t} \alpha, \beta^{\mathsf{UDD}}(\lambda_i)\rangle} = (-1)^{\sum_{i=1}^{j} \alpha} = (-1)^{a_2 j} \qquad \text{for } t \in [\Delta_j T, \Delta_{j+1} T) \ ,$$

where $\alpha = (a_1, a_2) \in A$ and K_t from Eq. (4.24). As a consequence, the phases in $F_{\alpha,\beta^{\mathsf{UDD}}}(t)$, and hence the DD properties, do not depend on the label a_1. Furthermore, the second label a_2 is given by 0 or 1 for the two Pauli matrices that appear in the dephasing noise channel $\sigma_{(0,0)} = I$ and $\sigma_{(0,1)} = \sigma_z$, respectively.

Lemma 4.16 for a single nesting level directly implies the following statement.

Lemma 4.17 *(Properties of the Uhrig pulse times).* Let $N \in \mathbb{N}$ and Δ_j be given by Eq. (4.36) for $j = 0,\ldots,N$. For $\alpha \in \mathbb{Z}_2$, define the function $F_\alpha^{\mathsf{UDD}} \colon [0,1] \to \{-1,1\}$ via

$$\begin{aligned} F_0^{\mathsf{UDD}}(\tau) &= 1 && \text{for all } \tau \in [0,1] \text{ and} \\ F_1^{\mathsf{UDD}}(\tau) &= (-1)^j && \text{for all } \tau \in [\Delta_j, \Delta_{j+1}) \ . \end{aligned}$$

Then

$$\mathcal{F}_{\vec{\alpha}}^{\vec{r}} \left(\{F_{\alpha_i}^{\mathsf{UDD}}\}_{i=1}^k \right) = 0 \quad \begin{cases} \text{for all } k \in \mathbb{N},\ \vec{r} \in \mathbb{N}_0^k, \text{ and } \vec{\gamma} \in \{0,1\}^k \\ \text{such that } k + r_1 + \ldots + r_k \leq N \text{ and } \bigoplus_{i=1}^k \alpha_i = 1, \end{cases}$$

where for $k \in \mathbb{N}$, $\vec{r} \coloneqq (r_1,\ldots,r_k)$, $\vec{\alpha} \coloneqq (\alpha_1,\ldots,\alpha_k)$ and $\oplus$ denotes addition modulo two. □

Similar to the NUDD functions F_α^{NUDD} we write $F_\alpha^{\mathsf{UDD}} = F_{\alpha,\beta^{\mathsf{UDD}}}$ to simplify the notation. Note furthermore that the condition $\bigoplus_{i=1}^k \alpha_i \neq 0$ is equivalent to $\bigoplus_{i=1}^k \alpha_i = 1$.

Lemma 4.17 does not guarantee universal dynamical decoupling for a single qubit. Only decoherence Hamiltonians of pure dephasing, i.e., $H^{\mathrm{orig}}(t) = I \otimes B_0(t) + \sigma_z \otimes B_1(t)$, are approximately averaged out. We use this lemma in the context of higher order bosonic decoherence suppression schemes (cf. Section 4.6).

4.6 Higher order bosonic decoherence suppression

Let us return to the CV setting. Here, we consider bosonic decoherence suppression in the sense of Definition 4.8 and propose a novel and efficient Nth order decoherence suppression scheme.

We first set the stage by defining the passive Gaussian unitary pulse in Section 4.6.1: this is the same pulses as proposed by Burgarth, Arenz and Hillier [12] for their first order scheme

but we adapt the pulse timings to prove higher order decoherence suppression. In Section 4.6.2, we construct a concrete pulse sequence which uses N applications of this pulse to achieve Nth order decoherence suppression – hence admitting the same linear scaling as the Uhrig DD scheme. We choose the (non-equidistant) pulse times associated with the Uhrig DD scheme. More precisely, we formulate a sufficient condition for Nth order decoherence suppression in terms of the pulse timings: we find the same integral equations as for the UDD sequence (cf. Lemma 4.17). In Section 4.6.3, we analyse the question how sequences with this (same) pulse but with different pulse timings perform in comparison to the Uhrig decoherence suppression scheme proposed before.

4.6.1 A bosonic pulse for decoherence suppression

Definition 4.18 *(Decoherence suppression pulse).* On a quantum system of $n_S \in \mathbb{N}$ bosonic modes, define the unitary $U \coloneqq (U_{\mathsf{rot}}(\pi))^{\otimes n_S}$ where $U_{\mathsf{rot}}(\pi)$ is the passive Gaussian unitary from Eq. (2.43) in Section 2.2.5.

As computed in Section 2.2.5, the unitary U acts as the sign flip on all quadratures, i.e., as

$$U^\dagger Q_k U \coloneqq -Q_k \quad , \quad U^\dagger P_k U \coloneqq -P_k$$

for $k = 1, \ldots, n_S$ and its associated symplectic matrix $M \in \mathsf{Sp}(2n_S, \mathbb{R})$ is

$$M \coloneqq -I_{2n_S} \ . \tag{4.58}$$

The matrix M is an exponential of πJ_{n_S}, i.e., it is generated by $\pi I_{2n_S} \in \mathfrak{sp}(2n_S, \mathbb{R})$. This unitary U was previously used in Theorem 4 of [12] where it was shown to achieve decoherence suppression up to first order.

In the following we will consider CV pulse sequences at the symplectic level (cf. Definition 4.7), more precisely, those of the form

$$\left(\{t_j\}_{j=1}^L, \{M\}_{j=1}^L\right) \ , \quad \text{where } M - I_{2n_S} \text{ as in Eq. (4.58)} \ , \tag{4.59}$$

and analyse their (higher order) decoherence suppression properties. As a consequence, the only freedom of choice is in picking the number of pulses L and their timings $\{t_j\}_{j=1}^L$. In Section 4.6.2, we will prove Nth order decoherence suppression is achieved when using the pulse timings of the Uhrig dynamical decoupling sequence whereas in Section 4.6.3, we compare the Uhrig pulse timings to other pulse timings and discuss the efficiency in terms of decoherence suppression order and dependence on pulse timings.

Decoherence suppression

Consider a pulse sequence of the form of Eq. (4.59) in order to analyse the decoherence suppression properties.

Since such pulses change the signs of the quadrature operators (acting as $M = -I_{2n_S}$ on the vector of system quadratures R), the symplectic control evolution of the pulse sequence (4.59) is of the simple form

$$S^{\text{control}}(t) = \left(\prod_{j=1}^{K_t} M\right) \oplus I_{2n_S} = (-1)^{\sum_{j=1}^{K_t} 1} I_{2n_S} \oplus I_{2n_E} \tag{4.60}$$

where $M = -I_{2n_S}$ and K_t is defined in Eq. (4.24). Let the function $\sigma : [0,1] \to \{-1,1\}$ be given such that

$$\sigma(t/T) \coloneqq (-1)^{\sum_{j=1}^{K_t} 1} = \begin{cases} 1 & \text{if } t \in [t_j, t_{j+1}) \text{ for } j \text{ even} , \\ -1 & \text{if } t \in [t_j, t_{j+1}) \text{ for } j \text{ odd} , \end{cases} \tag{4.61}$$

where we set $t_0 \coloneqq 0$ and $t_{L+1} \coloneqq T$. The function σ defines a pulse sequence (4.59) since the normalised times $t_j/T \in [0,T]$ correspond to the values at which σ changes the sign. As a consequence, the control evolution (4.60) takes the simple form

$$S^{\text{control}}(t) = \sigma(t/T) I_{2n_S} \oplus I_{2n_E} .$$

The setup we consider is that of Definition 4.6, i.e., system and environment interact via an original generator $X^{\text{orig}} : [0,T] \to \mathsf{Sp}(2n_S + 2n_E, \mathbb{R})$ with analytic time dependence. Then the generator of the toggling frame evolution from Eq. (4.27) can be written as

$$X^{\text{tf}}(t) \coloneqq S^{\text{control}}(t)^{-1} X^{\text{orig}}(t) S^{\text{control}}(t) = \begin{pmatrix} X_{SS}(t) & \sigma(t/T) X_{SE}(t) \\ \sigma(t/T) X_{ES}(t) & X_{EE}(t) \end{pmatrix} , \tag{4.62}$$

i.e., the system as well as the environment part both do not change over time whereas the off-diagonal blocks mixing the system and the environment quadratures change the sign.

4.6.2 Uhrig decoherence suppression for bosonic systems

Let us give a sufficient condition on the function σ, or equivalently on the pulse times $\{t_j\}_{j=1}^{L}$, for decoherence suppression up to order N in T in the sense of Definition 4.8.

Lemma 4.19 *(Sufficient condition for decoherence suppression).* Let $N \in \mathbb{N}$ and let furthermore $\sigma : [0,1] \to \{-1,1\}$ be a piecewise constant function that satisfies $\sigma(0) = 1$ and changes the sign N times. Suppose that this function satisfies

$$\int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} \sigma(\tau_k)^{\gamma_k} \cdots \sigma(\tau_1)^{\gamma_1} \tau_k^{r_k} \cdots \tau_1^{r_1} \, d\tau_k \cdots d\tau_2 d\tau_1 = 0 \tag{4.63}$$

for all $k \in \mathbb{N}$, $r_1, \ldots, r_k \in \mathbb{N}_0$ and $\gamma_1, \ldots, \gamma_k \in \mathbb{Z}_2$ such that

$$k + \sum_{m=1}^{k} r_m \leq N \qquad \text{and} \qquad \bigoplus_{m=1}^{k} \gamma_m = 1 ,$$

where $\oplus$ denotes summation modulo 2.

Then the pulse sequence from Eq. (4.59) where $\{t_j/T\}_{j=1}^{N}$ are the values at which σ changes the sign achieves Nth order bosonic decoherence suppression in the sense of Definition 4.8. □

The proof strategy is the following: One expands the toggling frame evolution $S^{\text{tf}}(T)$ in a Dyson series, which is expressed in terms of the function σ. One concludes the proof by showing that the condition (4.63) implies that the higher order (larger than N) terms of the off-diagonal parts $(S^{\text{tf}}(T))_{ES}$ and $(S^{\text{tf}}(T))_{SE}$ of $S^{\text{tf}}(T)$ vanish.

Proof. By the assumption (4.23) on the pulse sequence we have $S^{\text{tf}}(T) = S^{\text{res}}(T)$. Hence Nth order decoherence suppression in the form of Eq. (4.31) is equivalent to the statement that the

off-diagonal terms $(S^{\mathrm{tf}}(T))_{SE}$ and $(S^{\mathrm{tf}}(T))_{ES}$ of the toggling frame evolution (4.29) vanish up to order N in T.

Let us therefore consider the Dyson series of the toggling frame evolution $S^{\mathrm{tf}}(T)$ which is given by Eq. (4.28). It is a $2(n_S+n_E)\times 2(n_S+n_E)$ matrix of the block form

$$S^{\mathrm{tf}}(T)=\begin{pmatrix}(S^{\mathrm{tf}}(T))_{SS} & (S^{\mathrm{tf}}(T))_{SE}\\ (S^{\mathrm{tf}}(T))_{ES} & (S^{\mathrm{tf}}(T))_{EE}\end{pmatrix}$$

where the two off-diagonal (more precisely, off block diagonal) matrices have Dyson series'

$$\begin{aligned}(S^{\mathrm{tf}}(T))_{SE}&=\sum_{k=0}^{\infty}\int_0^T\int_0^{s_1}\dots\int_0^{s_{k-1}}\Big(X^{\mathrm{tf}}(s_1)X^{\mathrm{tf}}(s_2)\cdots X^{\mathrm{tf}}(s_k)\Big)_{SE}\,ds_k\dots ds_2ds_1\ ,\\ (S^{\mathrm{tf}}(T))_{ES}&=\sum_{k=0}^{\infty}\int_0^T\int_0^{s_1}\dots\int_0^{s_{k-1}}\Big(X^{\mathrm{tf}}(s_1)X^{\mathrm{tf}}(s_2)\cdots X^{\mathrm{tf}}(s_k)\Big)_{ES}\,ds_k\dots ds_2ds_1\ .\end{aligned}\tag{4.64}$$

Let us start by computing the expression

$$\Big(X^{\mathrm{tf}}(s_1)X^{\mathrm{tf}}(s_2)\cdots X^{\mathrm{tf}}(s_k)\Big)_{SE}\tag{4.65}$$

for $s_1,\dots,s_k\in[0,T]$. For $k\in\mathbb{N}$ and a pair $(C,D)\in\{S,E\}^2$ it will be convenient to introduce the set $\nu_k(C,D)$ of sequences $(A,B)=(A_1,\dots,A_k,B_1,\dots,B_k)$ which satisfy

$$(A,B)=(C,B_1,\dots,B_{k-1},B_1,\dots,B_{k-1},D)\ .\tag{4.66}$$

Since every $X^{\mathrm{tf}}(s_i)$ for $i=1,\dots,k$ is a matrix of the form

$$X^{\mathrm{tf}}(s_k)=\begin{pmatrix}X^{\mathrm{tf}}_{SS}(s_k) & X^{\mathrm{tf}}_{SE}(s_k)\\ X^{\mathrm{tf}}_{ES}(s_k) & X^{\mathrm{tf}}_{EE}(s_k)\end{pmatrix}\ ,$$

the matrix product in (4.65) can be expressed as

$$\Big(X^{\mathrm{tf}}(s_1)X^{\mathrm{tf}}(s_2)\cdots X^{\mathrm{tf}}(s_k)\Big)_{SE}=\sum_{A,B\in\nu_k(S,E)}X^{\mathrm{tf}}_{A_1B_1}(s_1)\cdots X^{\mathrm{tf}}_{A_kB_k}(s_k)\ ,\tag{4.67}$$

where we used the set of sequence $\nu_k(S,E)$ from Eq. (4.66). For the pulse sequence (4.59), we computed the form of the toggling frame generator in Eq. (4.62). An insertion of this (4.62) into Eq. (4.67) yields for $k\in\mathbb{N}$

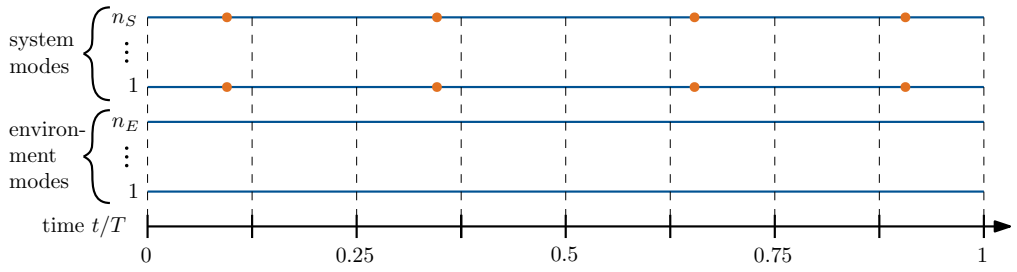
$$\Big(X^{\mathrm{tf}}(s_1)\cdots X^{\mathrm{tf}}(s_k)\Big)_{SE}=\sum_{(A,B)\in\nu_k(S,E)}\sigma(\tfrac{s_1}{T})^{\alpha_1}\cdots\sigma\left(\tfrac{s_k}{T}\right)^{\alpha_k}X_{A_1B_1}(s_1)\cdots X_{A_kB_k}(s_k)\ ,\tag{4.68}$$

where we defined the expression

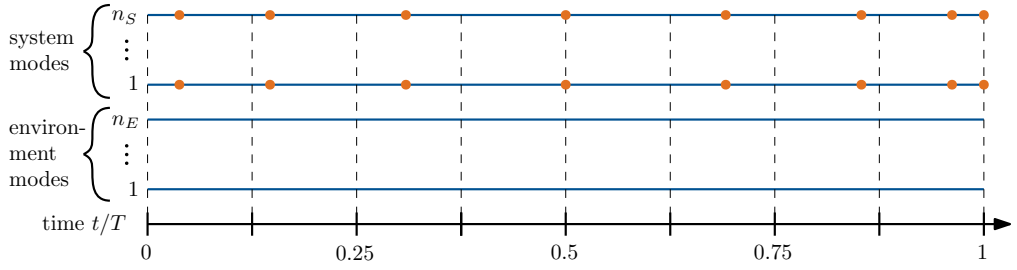
$$\alpha(C,D)\coloneqq\begin{cases}0 & \text{if } (C,D)\in\{(S,S),(E,E)\}\\ 1 & \text{if } (C,D)\in\{(S,E),(E,S)\}\end{cases}$$

and, to shorten notation, we write $\alpha_i\coloneqq\alpha(A_i,B_i)$ for $i=1,\dots,k$ and for $(A,B)\in\nu_k(S,E)$. Note that property (4.66) implies that

$$\bigoplus_{i=1}^{k}\alpha_i\coloneqq\sum_{i=1}^{k}\alpha(A_i,B_i)\mod 2=1\ ,\tag{4.69}$$



(a) $N = 4$



(b) $N = 7$

Figure 4.4: CV Uhrig decoherence suppression schemes from Theorem 4.20: the orange dots signify application of the pulse unitary $U_{\mathrm{rot}}(\pi)$ (from Definition 4.18) on the respective mode and solid blue lines represent evolution under the decoherence Hamiltonian $H^{\mathrm{orig}}(t)$ on system and environment in between the pulse times.

for all $(A, B) \in \nu_k(S, E)$ and all $(A, B) \in \nu_k(E, S)$.

By assumption, furthermore, the submatrices of the original generator $X^{\mathrm{orig}}(t)$ admit the analytic expansions

$$X_{A_iB_i}(t) = \sum_{r_i=0}^{\infty} X_{A_iB_i,r_i} t^{r_i}$$

where $A_i, B_i \in \{S, E\}$ and where $i = 1, \ldots, k$. Let us now insert this together with Eq. (4.68) into Eq. (4.64). This implies

$$\begin{aligned}(S^{\mathrm{tf}}(T))_{SE} = \sum_{k=0}^{\infty} \int_0^T \int_0^{s_1} \cdots \int_0^{s_{k-1}} \sum_{(A,B)\in\nu_k(S,E)} \left(\sigma\left(\frac{s_1}{T}\right)^{\alpha(A_1,B_1)} \cdots \sigma\left(\frac{s_k}{T}\right)^{\alpha(A_k,B_k)}\right) \\ \cdot \Big(\sum_{r_1=0}^{\infty} X_{A_1B_1,r_1} s_1^{r_1}\Big) \cdots \Big(\sum_{r_k=0}^{\infty} X_{A_kB_k,r_k} s_k^{r_k}\Big) ds_k \ldots ds_2 ds_1 \\ = \sum_{k=0}^{\infty} \sum_{r_1,\ldots,r_k=0}^{\infty} \sum_{(A,B)\in\nu_k(S,E)} T^{k+r_1+\cdots+r_k} \left(\prod_{i=1}^{k} X_{A_iB_i,r_i}\right) \\ \cdot \int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} \left(\prod_{i=1}^{k} \sigma(\tau_i)^{\alpha(A_i,B_i)} \tau_i^{r_i}\right) d\tau_k \ldots d\tau_2 d\tau_1\end{aligned}$$

where in the second step, we performed a variable substitution. By assumption

$$\int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} \left(\prod_{i=1}^{k} \sigma(\tau_i)^{\alpha_i} \tau_i^{r_i}\right) d\tau_k \ldots d\tau_2 d\tau_1 = 0$$

for all $k \in \mathbb{N}$, $r_1, \ldots, r_k \in \mathbb{N}_0$ and $\alpha_1, \ldots, \alpha_k \in \mathbb{Z}_2$ such that

$$k + \sum_{i=1}^{k} r_i \leq N \qquad \text{and} \qquad \bigoplus_{i=1}^{k} \alpha_i = 1 \ . \tag{4.70}$$

Here, the last condition of (4.70) holds for $\alpha_i = \alpha(A_i, B_i)$ due to property (4.69) of all sequences $(A, B) \in \nu_k(S, E)$. Hence the terms in $(S^{\mathrm{tf}}(T))_{SE}$ of order higher than N vanish. In summary, we proved that

$$(S^{\mathrm{tf}}(T))_{SE} = O(T^{N+1}) \ . \tag{4.71}$$

A similar reasoning for $(S^{\mathrm{tf}}(T))_{ES}$ yields that

$$\begin{aligned}(S^{\mathrm{tf}}(T))_{ES} = \sum_{k=0}^{\infty} \sum_{r_1,\ldots,r_k=0}^{\infty} \sum_{(A,B)\in\nu_k(E,S)} T^{k+r_1+\cdots+r_k} \left(\prod_{j=1}^{k} X_{A_j B_j, r_j} \right) \\ \cdot \int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} \left(\prod_{i=1}^{k} \sigma(\tau_m)^{\alpha(A_i,B_i)} \tau_i^{r_i} \right) d\tau_k \ldots d\tau_2 d\tau_1 \ .\end{aligned}$$

Here, property (4.69) of sequences $(A, B) \in \nu_k(E, S)$ as well as the assumption (4.63) guarantee $(S^{\mathrm{tf}}(T))_{ES} = O(T^{N+1})$. Together with Eq. (4.71) and using that $S^{\mathrm{res}}(T) = S^{\mathrm{tf}}(T)$ this implies that there are two constants $c_1, c_2 > 0$ such that

$$\begin{aligned}\|S^{\mathrm{res}}(T) - (S^{\mathrm{tf}}(T))_{SS} \oplus (S^{\mathrm{tf}}(T))_{EE}\|^2 &= \left\| \begin{pmatrix} 0 & (S^{\mathrm{tf}}(T))_{SE} \\ (S^{\mathrm{tf}}(T))_{ES} & 0 \end{pmatrix} \right\|^2 \\ &= \|(S^{\mathrm{tf}}(T))_{SE}\|^2 + \|(S^{\mathrm{tf}}(T))_{SE}\|^2 \\ &\leq (c_1^2 + c_2^2) T^{2(N+1)} \ .\end{aligned}$$

By choosing $S_{SS} \coloneqq (S^{\mathrm{tf}}(T))_{SS}$ and $S_{EE} \coloneqq (S^{\mathrm{tf}}(T))_{EE}$ one obtains that the resulting evolution $S^{\mathrm{res}}(T)$ satisfies Eq. (4.31), i.e., the pulse sequence achieves decoherence suppression in the sense of Definition 4.8. ■

A concrete pulse sequence will achieve Nth order decoherence suppression if its associated function σ (or equivalently its associated pulse times) satisfies the integral equation (4.63). Let us now construct a pulse sequence which satisfies this sufficient condition. Recall the definition of the scalars $\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}$ in Eq. (4.51) of Lemma 4.13: One can easily see that

$$\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)} \left(\{\sigma^{\alpha_i}\}_{i=1}^k \right) = \int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} \prod_{i=1}^{k} (\sigma(\tau_i)^{\alpha_i} \tau_i^{r_i}) \, d\tau_k \ldots d\tau_2 d\tau_1 \ .$$

Lemma 4.19 hence gives a sufficient condition for decoherence suppression: a pulse sequence defined by the function σ achieves Nth order decoherence suppression if σ satisfies

$$\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)} \left(\{\sigma^{\alpha_i}\}_{i=1}^k \right) = 0 \quad \left\{ \begin{array}{l} \text{for all } k \in \mathbb{N},\ r_1, \ldots, r_k \in \mathbb{N}_0,\ \alpha_1, \ldots, \alpha_k \in \{0,1\} \\ \text{such that } k + \sum_{i=1}^k r_i \leq N \text{ and } \bigoplus_{i=1}^k \alpha_i = 1. \end{array} \right. \tag{4.72}$$

An example of such a function is the UDD pulse time function (as shown in Lemma 4.17).

Theorem 4.20 *(Uhrig decoherence suppression)***.** Consider a system of $n_S \in \mathbb{N}$ bosonic modes and an environment of $n_E \in \mathbb{N}$ bosonic modes. Let $X^{\text{orig}} : [0,T] \to \mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$ be a time-dependent analytic generator that satisfies Definition 4.6.

Then the (symplectic) pulse sequence

$$\left(\{t_j^{\mathsf{UDD}}\}_{j=1}^N, \{-I_{n_S}\}_{j=1}^N\right)$$

for UDD pulse times $t_j^{\mathsf{UDD}} = T\Delta_j$ where Δ_j from Eq. (4.36) and for the symplectic pulse $-I_{n_S}$ associated with the unitary U from Definition 4.18 achieves Nth order decoherence suppression in the sense of Definition 4.8. □

Proof. Let us first show that $\sigma(t)^0 = F_0^{\mathsf{UDD}}(t)$ and $\sigma(t)^1 = F_1^{\mathsf{UDD}}(t)$ for $t \in [0,T]$. This can be seen since

$$\sigma(\tau)^\gamma = (-1)^{j\gamma} = F_\gamma^{\mathsf{UDD}}(\tau) \quad \text{for } \tau \in [\Delta_j, \Delta_{j+1}) \ ,$$

for $\gamma \in \{0,1\}$. This function σ satisfies Eq. (4.63) by Lemma 4.17. As the assumptions of Lemma 4.19 are satisfied this lemma implies that the above pulse sequence achieves Nth order decoherence suppression in the sense of Definition 4.8. ■

We note that there are two remarkable features of this result: the number of required pulses to achieve a suppression order N is equal to this parameter N – i.e., it grows linearly with N. This number is furthermore independent of both the number of system modes n_S as well as the number of environment modes n_E. As a consequence, the pulse sequences are extremely simple and resource-efficient. Fig. 4.4 shows this bosonic UDD decoherence suppression scheme of order $N = 4$ an order $N = 7$.

4.6.3 Decoherence suppression of a single mode for arbitrary pulse times

Here we consider the effect of a pulse sequence of the pulse from Definition 4.6.1 but with arbitrary pulse times $\{t_j\}_{j=1}^L$. We are particularly interested in the question which pulse times are optimal in terms of decoherence suppression order and which pulse times achieve best effective decoherence suppression in dependence of certain aspects of the environment and the initial state.

Optimality of UDD times

In Theorem 4.20, we showed the following: we established sufficient conditions on the pulse times such that the pulse sequence – using the pulse U from Definition 4.18 – achieves Nth order CV decoherence suppression. More precisely, these conditions are given by the integral equations (4.63) where the function σ defines the pulse times via Eq. (4.61). Furthermore, they are the *same* as Uhrig [161] found to achieve Nth order dynamical decoupling of single-qubit against pure dephasing noise using σ_x-pulses: As shown by Uhrig, the UDD times satisfy these integral equations and they are furthermore *optimal* in the sense that they require the lowest number L of pulses to achieve Nth order DD/decoherence suppression. In this sense – achieving the highest decoherence suppression order using the least number of pulses – the *UDD times are optimal in the bosonic setting* as well. More precisely, under the

assumptions that we only use the sign flip pulse from Definition 4.18 and that decoherence is described by quadratic Hamiltonians with analytic time dependence, the UDD pulse times achieve the optimal scaling between the number of required pulse times and decoupling order – a linear scaling.

Other measures of pulse sequences

The asymptotic scaling described by the relation between the number of pulses and decoupling order is not the only relevant figure of merit. When additional information on the initial state (to be protected) and the environment is at hand, it might be more relevant to directly compute the *error* experienced by this initial state and compare this error of different pulse schemes. For dynamical decoupling in the finite-dimensional setting, one wants to know how well a given initial system state ρ_S is protected from decoherence, i.e., how well it agrees with the resulting system state $\rho_S^{\text{res}} = \text{tr}_E(\rho^{\text{res}})$, where $\rho^{\text{res}} := U^{\text{res}}(T)\,(\rho_S \otimes \rho_E)\,U^{\text{res}}(T)^\dagger$ denotes the resulting system-environment state after time T under the decoherence evolution and the pulse sequence.

Recall from Definition 4.8 that a bosonic pulse sequence achieves (exact) decoherence suppression if $U^{\text{res}}(T) = U_S \otimes U_E$ where U_S and U_E are system and environment unitaries, respectively. Hence the relevant question in the bosonic setting is how well the map

$$\rho_S \mapsto \rho_S^{\text{res}} = \text{tr}_E\left(U^{\text{res}}(T)\,(\rho_S \otimes \rho_E)\,U^{\text{res}}(T)^\dagger\right) \tag{4.73}$$

corresponds to a unitary transformation. If furthermore the initial environment state is assumed to be Gaussian (e.g. a thermal state), this simplifies to the following task: quantify how well the map (4.73) corresponds to a *unitary Gaussian transformation* on a given state. A measure of the 'non-Gaussian-unitarity' then quantifies the error of the decoherence suppression scheme. Here, we choose the quantity

$$\max_{S \in \mathsf{Sp}(2n_S,\mathbb{R})} \left\| S V_S S^T - V_S^{\text{res}}(T) \right\|$$

as such a measure, where V_S is the covariance matrix (cf. Definition 2.26) of an Gaussian initial system state ρ_S and $V_S^{\text{res}}(T)$ is the covariance matrix of the resulting system state ρ_S^{res}. This is a suitable measure for the unitarity since we find that this norm difference does not depend on the initial system state ρ_S, as long as it is Gaussian, but only on the properties of the original Hamiltonian, the pulse sequence and the environment temperature.

In this section, we will compute this error for a concrete example system for bosonic decoherence suppression: a single bosonic mode, an environment initially in thermal equilibrium, a specific noise model and the bosonic decoherence suppression pulse sequence with pulse U from Definition 4.18 and a priori unspecified times $\{t_j\}_{j=1}^L$. Surprisingly we find that the quantity which characterises this non-unitarity in our model is the *same* as the quantity characterising the non-optimality of qubit DD in the spin boson model (cf. next paragraph).

This analysis is inspired by the finite-dimensional case where the UDD pulse times are compared to more general pulse times.

Optimality of pulse times in the spin-boson model

Let us first briefly recap the case of a finite-dimensional system: In the *spin boson model*, the system $\mathcal{H}_S = \mathbb{C}^2$ of a single qubit is coupled to an environment of bosonic modes via the

Hamiltonian

$$H^{\mathrm{sb}} \coloneqq \sum_{i=1}^{\infty} \omega_i b_i^\dagger b_i + \sigma_z \sum_{i=1}^{\infty} \lambda_i (b_i^\dagger + b_i) \tag{4.74}$$

Here, $b_i^\dagger$ and b_i are creation and annihilation operators on the environment, respectively, and ω_i, λ_i are real parameters[3]. In [162], Uhrig analyses the dynamical decoupling properties of σ_x-pulses at priori unspecified pulse times $t_1, t_2, \ldots, t_L$. He investigates the effect of various cut-offs of the environment spectral density

$$J(\omega) \coloneqq \sum_{i=1}^{\infty} \lambda_i^2 \delta(\omega - \omega_i) \ .$$

He finds that the harder the high-frequency cut-off the better the pulse sequence with UDD times performs.

Let system and environment initially be in a tensor product state where the environment is in thermal equilibrium at inverse temperature β and

$$|\psi_{\mathsf{init}}\rangle \coloneqq e^{i\pi/4\sigma_x}|0\rangle = \frac{1}{\sqrt{2}}\left(I + i\sigma_x\right)|0\rangle \tag{4.75}$$

is the system's initial state – the $+1$-eigenstate in σ_z-basis rotated about the x-axis by the amount $\pi/2$. The figure of merit considered is the so-called signal at time T, a quantity defined as

$$s(T) \coloneqq \langle\psi_{\mathsf{init}}|U^{\mathrm{res}}(T)^\dagger\, \sigma_y\, U^{\mathrm{res}}(T)|\psi_{\mathsf{init}}\rangle \ .$$

It corresponds to the expectation value of σ_y in the time-evolved state, i.e., the initial state from Eq. (4.75) undergoes the evolution $U^{\mathrm{res}}(T)$ followed by a σ_y-measurement.

If no pulse sequence is applied, then $U^{\mathrm{res}}(T) = U^{\mathrm{orig}}(T) = e^{-iTH^{\mathrm{sb}}}$. As a consequence the signal is given by $s(T) = e^{-2\chi_0(T)}$, i.e., it is the inverse exponential of the parameter

$$\chi_0(T) \coloneqq \int_0^\infty \frac{J(\omega)}{\omega^2} \coth\left(\frac{\beta\omega}{2}\right) 4\sin^2\left(\frac{\omega T}{2}\right) d\omega = \sum_{i=1}^{\infty} \frac{\lambda_i^2}{\omega_i^2} \coth\left(\frac{\beta\omega_i}{2}\right) 4\sin^2\left(\frac{\omega_i T}{2}\right).$$

Note that this reproduces an earlier result from the same author [161]. If a pulse sequence of σ_x-pulses at times $t_1, t_2, \ldots, t_L$ is applied, then the resulting evolution

$$U^{\mathrm{res}}(T) = e^{-i(T-t_L)H^{\mathrm{sb}}} \sigma_x e^{-i(t_L - t_{L-1})H^{\mathrm{sb}}} \sigma_x \cdots e^{-i(t_2-t_1)H^{\mathrm{sb}}} \sigma_x e^{-it_1 H^{\mathrm{sb}}}$$

gives the signal $s(T) = e^{-2\chi(T)}$ (cf. [162]). It is the inverse exponential of the parameter

$$\begin{aligned} \chi(T) &\coloneqq \int_0^\infty \frac{S_\beta(\omega)}{\omega^2} |y_L(\omega T)|^2 d\omega \\ &= \int_0^\infty \frac{J(\omega)}{\omega^2} \coth\left(\frac{\beta\omega}{2}\right) |y_L(\omega T)|^2 d\omega \\ &= \sum_i \frac{\lambda_i^2}{\omega_i^2} |y_L(\omega_i T)|^2 \coth\left(\frac{\beta\omega_i}{2}\right) \ . \end{aligned} \tag{4.76}$$

[3] We note that there are other conventions for the parameters in H^{sb} – such as the one used in the work [162] on optimality of the UDD sequence by Uhrig – where $\lambda_i \mapsto \lambda_i/2$ compared to Eq. (4.74). For easier comparison with our bosonic model, we consider Uhrig's results for the spin boson model in the form of Eq. (4.74).

Here, $S_\beta(\omega) \coloneqq J(\omega)\coth(\beta\omega/2)$ is the noise spectrum of system-bath couplings and the filter function

$$y_L(z) \coloneqq 1 - e^{iz} + 2\sum_{j=1}^{L}(-1)^j e^{izt_j/T} \tag{4.77}$$

encodes all properties of the pulse sequence.

In the following, we translate this analysis to the bosonic setting. More precisely, we construct a specific system-environment model in which we find that the same expression (4.76) characterises the optimality of the pulse times in the exact same way as within the finite-dimensional setting.

Pure bosonic model and decoherence suppression

The model we consider is a pure-bosonic relative of the so-called spin-boson model.

Let one system mode with mode operators Q and P (we write Q and P instead of Q_1^S and P_1^S for the system quadratures in order to shorten notation) be coupled to several bosonic modes of the environment Q_i and P_i for $i = 1, 2, \ldots, n_E$. Let the original system-environment Hamiltonian introducing decoherence be of the form

$$H^{\text{orig}} = \frac{1}{2}\sum_{i=1}^{n_E}\omega_i(Q_i^2 + P_i^2) + Q\sum_{i=1}^{n_E}\lambda_i Q_i \ . \tag{4.78}$$

The Hamiltonian H^{orig} captures quadratic terms in the mode operators, i.e., it is a special case of the more general form $H^{\text{orig}} = \sum_{j,k=1}^{2(1+n_E)} A_{jk}(t)R_jR_k$ from Eq. (4.19) for the vector of quadratures $R^T = (Q, P, Q_1, Q_2, \ldots, Q_{n_E}, P_1, P_2, \ldots, P_{n_E})$ and for the matrix

$$A = \begin{pmatrix} 0 & 0 & \lambda^T & 0_{1\times n_E} \\ 0 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ \lambda & 0_{n_E\times 1} & \Omega & 0 \\ 0_{n_E\times 1} & 0_{n_E\times 1} & 0 & \Omega \end{pmatrix} \in \mathbb{R}^{2(1+n_E)\times 2(1+n_E)} \ , \tag{4.79}$$

where

$$\lambda \coloneqq \begin{pmatrix}\lambda_1 & \lambda_2 & \cdots & \lambda_{n_E}\end{pmatrix}^T \qquad , \qquad \Omega \coloneqq \operatorname{diag}(\omega_1, \omega_2, \ldots, \omega_{n_E}) \ .$$

Note that the matrix $A(t) = A$ does not dependent on time.

We will analyse the unitarity of the resulting evolution $U^{\text{res}}(T)$ after application of the decoupling pulse

$$U \coloneqq e^{i\frac{\pi}{2}(Q^2+P^2)} \tag{4.80}$$

from Definition 4.18 at times $t_1, t_2, \ldots, t_L \in [0, T]$ and evolving under $U^{\text{orig}}(t) \coloneqq e^{-itH^{\text{orig}}}$ in between. This resulting evolution is of the form

$$U^{\text{res}}(T) \coloneqq e^{-i(T-t_L)H^{\text{orig}}}(U \otimes I_E)\cdots e^{-i(t_2-t_1)H^{\text{orig}}}(U \otimes I_E)e^{-it_1H^{\text{orig}}} \ . \tag{4.81}$$

We will show for general pulse times that $U^{\text{res}}(T)$ acts jointly on the system and the environment and that it is not of a tensor product form $U_S^{\text{res}}(T) \otimes U_E^{\text{res}}(T)$ for two Gaussian unitaries $U_S^{\text{res}}(T) \in \mathsf{U}(L^2(\mathbb{R}))$ and $U_E^{\text{res}}(T) \in \mathsf{U}(L^2(\mathbb{R}^{n_E}))$: the above pulse sequence – pulse U applied at times $t_1, t_2, \ldots, t_L \in [0, T]$ – *does not achieve decoherence suppression* in the sense of Definition 4.8. For this general case we will introduce a quantity which characterises the error made, i.e., a measure for the amount of non-decoherence-suppression: the

non-Gaussian-unitarity of $U^{\text{res}}(T)$. We will characterise the non-unitarity and thereby non-decoherence suppression property by a parameter, which is similar to the so-called 'signal' from Eq. (4.76).

In order to do so, we assume that at time $t = 0$, the single system mode $\mathcal{H}_S = L^2(\mathbb{R})$ is decoupled from the environment such that the latter is in thermal equilibrium of inverse temperature $\beta \in \mathbb{R}_+$: the joint system-environment state is in tensor product form $\rho_S \otimes \rho_E(\beta)$ where $\rho_S \in \mathcal{D}(\mathcal{H}_S)$ denotes the original system state and

$$\rho_E(\beta) = \frac{e^{-\beta \frac{1}{2} \sum_{i=1}^{n_E} \omega_i (Q_i^2 + P_i^2)}}{\operatorname{tr}\left(e^{-\beta \frac{1}{2} \sum_{i=1}^{n_E} \omega_i (Q_i^2 + P_i^2)} \right)} \tag{4.82}$$

is the thermal state of inverse temperature β (cf. Section 2.2.3).

Theorem 4.21 *(Optimality of Uhrig times for CV decoherence suppression).* Consider the setup described above. Let V_S be the covariance matrix of the original system state $\rho_S \in \mathcal{D}(L^2(\mathbb{R}))$ and let $\rho_E(\beta)$ be the thermal state on $n_E \in \mathbb{N}$ environment modes for inverse temperature β from Eq. (4.82).

Then the 2×2-block (i.e., the system block) of the covariance matrix of

$$\rho^{\text{res}}(T) := U^{\text{res}}(T) \left(\rho_S \otimes \rho_E(\beta) \right) U^{\text{res}}(T)^\dagger$$

can be written as

$$V_S^{\text{res}}(T) = \begin{pmatrix} 1 & 0 \\ x^{\text{res}} & 1 \end{pmatrix} V_S \begin{pmatrix} 1 & 0 \\ x^{\text{res}} & 1 \end{pmatrix}^T + \begin{pmatrix} 0 & 0 \\ 0 & y^{\text{res}} \end{pmatrix}$$

where the two real parameters x^{res} and y^{res} are given by

$$x^{\text{res}} := \sum_{i=1}^{n_E} \frac{\lambda_i^2}{\omega_i^2} \Big(\omega_i T - \sin(\omega_i T)(\mathsf{Re}(y_L(\omega_i T)) + 1) + (\cos(\omega_i T) - 1)\mathsf{Im}(y_L(\omega_i T)) \Big) ,$$

$$y^{\text{res}} := \sum_{i=1}^{n_E} \frac{\lambda_i^2}{\omega_i^2} |y_L(\omega_i T)|^2 \coth\left(\frac{\beta \omega_i}{2} \right) ,$$

for y_L from Eq. (4.77). □

The remainder of this section is dedicated to the proof of this result. The proof of Theorem 4.21 is divided into three steps: Steps one and two in Lemma 4.22 and 4.23, analyse the resulting evolution $U^{\text{res}}(T)$. It will be convenient to write it in terms of the operator

$$V(t) := U^{\text{orig}}(t)^\dagger U U^{\text{orig}}(t) = e^{itH^{\text{orig}}} U e^{-itH^{\text{orig}}} , \tag{4.83}$$

i.e., as the pulse U in the interaction picture associated with the original evolution $U^{\text{orig}}(t)$.

Step 0: Preliminary computations of symplectic matrices Since the proofs throughout this section are conducted at the level of the symplectic group and its associated Lie algebra, we start with some preliminary computations, where we compute the symplectic quantities associated with the relevant unitaries. First, recall that $U^{\text{orig}}(t) := e^{-itH^{\text{orig}}} = e^{-itR^T AR/2}$ is associated with a symplectic evolution $S^{\text{orig}}(t) := e^{tJA}$. for the symmetric matrix A from

Eq. (4.79). Direct computation shows that

$$S^{\text{orig}}(t) = e^{tJA} = \begin{pmatrix} 1 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ x(t,\Omega,\lambda) & 1 & w^T(t,\Omega,\lambda) & v^T(t,\Omega,\lambda) \\ v(t,\Omega,\lambda) & 0_{n_E\times 1} & \cos(\Omega t) & \sin(\Omega t) \\ w(t,\Omega,\lambda) & 0_{n_E\times 1} & -\sin(\Omega t) & \cos(\Omega t) \end{pmatrix} \tag{4.84}$$

where

$$v(t,\Omega,\lambda) \coloneqq \sum_{m=1}^{\infty}(-1)^m \frac{\Omega^{2m-1}t^{2m}}{(2m)!}\,\lambda = \Omega^{-1}(\cos(\Omega t) - I)\lambda\ , \tag{4.85}$$

$$w(t,\Omega,\lambda) \coloneqq -\sum_{m=0}^{\infty}(-1)^m \frac{\Omega^{2m}t^{2m+1}}{(2m+1)!}\,\lambda = -\Omega^{-1}\sin(\Omega t)\lambda\ , \tag{4.86}$$

$$x(t,\Omega,\lambda) \coloneqq -\lambda^T\sum_{m=1}^{\infty}(-1)^m \frac{\Omega^{2m-1}t^{2m+1}}{(2m+1)!}\,\lambda = -\lambda^T\Omega^{-2}(\sin(\Omega t) - \Omega t)\lambda\ . \tag{4.87}$$

Furthermore, we will need the inverse of the original evolution and its associated symplectic matrix, i.e., $U^{\text{orig}}(t)^\dagger \coloneqq e^{itH^{\text{orig}}} = e^{itR^TAR/2}$ and $S^{\text{orig}}(-t) \coloneqq e^{-tJA}$ where

$$S^{\text{orig}}(-t) = e^{-tJA} = \begin{pmatrix} 1 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ -x(t,\Omega,\lambda) & 1 & -w^T(t,\Omega,\lambda) & v^T(t,\Omega,\lambda) \\ v(t,\Omega,\lambda) & 0_{n_E\times 1} & \cos(\Omega t) & -\sin(\Omega t) \\ -w(t,\Omega,\lambda) & 0_{n_E\times 1} & \sin(\Omega t) & \cos(\Omega t) \end{pmatrix}\ .$$

Here, the sign change $t \mapsto -t$ implies $v(-t,\Omega,\lambda) = v(t,\Omega,\lambda)$, $w(-t,\Omega,\lambda) = -w(t,\Omega,\lambda)$, and $x(-t,\Omega,\lambda) = -x(t,\Omega,\lambda)$. Another quantity, that we will use, is the symplectic matrix associated with the unitary

$$e^{it\tilde{H}} \coloneqq e^{-itU^\dagger H^{\text{orig}}U} = e^{-\frac{i}{2}tR^T\tilde{A}R} \tag{4.88}$$

which is $e^{tJ\tilde{A}}$. Here, the Hamiltonian $\tilde{H}$ is defined as the conjugation of the original Hamiltonian H^{orig} with the pulse unitary U from Eq. (4.80), i.e., as

$$\tilde{H} \coloneqq -U^\dagger H^{\text{orig}}U = -\Big(\frac{1}{2}\sum_{i=1}^{n}\omega_i\left(Q_i^2 + P_i^2\right) - Q\sum_{i=1}^{n}\lambda_i Q_i\Big) = \frac{1}{2}\sum_{j,k}\tilde{A}_{jk}R_jR_k\ . \tag{4.89}$$

It agrees with the original Hamiltonian H^{orig} up to a sign change of Ω. In other words, the map $A \mapsto \tilde{A}$ corresponds to the substitution $\Omega \mapsto -\Omega$. Under this substitution, $w(t,-\Omega,\lambda) = w(t,\Omega,\lambda)$ and $\cos(-\Omega t) = \cos(\Omega t)$ both stay unchanged (cf. Eq. (4.86) where only even powers of Ω appear), whereas $v(t,-\Omega,\lambda) = -v(t,\Omega,\lambda)$, $x(t,-\Omega,\lambda) = -x(t,-\Omega,\lambda)$ and $\sin(-\Omega t) = \sin(\Omega t)$ change sign (cf. Eqs. (4.85) and (4.87) where only odd powers of Ω appear). As a consequence, we find that the symplectic matrix associated with (4.88) is given by

$$e^{tJ\tilde{A}} = \begin{pmatrix} 1 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ -x(t,\Omega,\lambda) & 1 & w^T(t,\Omega,\lambda) & -v^T(t,\Omega,\lambda) \\ -v(t,\Omega,\lambda) & 0_{n_E\times 1} & \cos(\Omega t) & -\sin(\Omega t) \\ w(t,\Omega,\lambda) & 0_{n_E\times 1} & \sin(\Omega t) & \cos(\Omega t) \end{pmatrix}\ . \tag{4.90}$$

Its inverse

$$e^{-tJ\tilde{A}} = \begin{pmatrix} 1 & 0 & 0_{1\times n} & 0_{1\times n} \\ x(t,\Omega,\lambda) & 1 & -w^T(t,\Omega,\lambda) & -v^T(t,\Omega,\lambda) \\ -v(t,\Omega,\lambda) & 0_{n\times 1} & \cos(\Omega t) & \sin(\Omega t) \\ -w(t,\Omega,\lambda) & 0_{n\times 1} & -\sin(\Omega t) & \cos(\Omega t) \end{pmatrix}$$

is related to the unitary evolution

$$e^{it\tilde{H}} \coloneqq e^{itU^\dagger H^{\text{orig}}U} = e^{\frac{i}{2}tR^T\tilde{A}R}\ .$$

Lemma 4.22 *(Step 1: conjugated pulse).* Let U be the unitary pulse from Eq. (4.80), let H^{orig} from Eq. (4.78) generate the original, uncontrolled evolution and let $U^{\text{res}}(t)$ be the resulting evolution from Eq. (4.81). Then $V(t)$ from Eq. (4.83) can be written as

$$V(t) = Ue^{iK(t)} \qquad \text{and} \qquad V(t) = e^{-iK(t)}U \tag{4.91}$$

where for $t \in \mathbb{R}$ and $z \in \mathbb{R}$ we define

$$K(t) := Q \sum_{k=1}^{n_E} \frac{\lambda_k}{\omega_k} \left(\mathsf{Re}(f(\omega_k t))Q_k + \mathsf{Im}(f(\omega_k t))P_k\right) \ , \tag{4.92}$$

$$f(z) := 2i(e^{iz} - 1) = -2\sin(z) + 2i(\cos(z) - 1) \ . \tag{4.93}$$

□

Proof. To show the first equation in (4.91), let us use the definition of $V(t)$ from Eq. (4.83) and rewrite the quantity $U^\dagger V(t)$ as

$$U^\dagger V(t) := U^\dagger e^{itH^{\text{orig}}} U e^{-itH^{\text{orig}}} = e^{itU^\dagger H^{\text{orig}} U} e^{-itH^{\text{orig}}} = e^{-it\tilde{H}} e^{-itH^{\text{orig}}} \ . \tag{4.94}$$

Here, the Hamiltonian $\tilde{H}$ is defined in Eq. (4.89). The remainder of the proof is conducted at the level of symplectic matrices. First, one uses the considerations from the preliminary step 0 – Eqs. (4.84) and (4.90) – to compute the symplectic matrix associated with the right hand side of Eq. (4.94). This is

$$e^{tJ\tilde{A}} e^{tJA} = \begin{pmatrix} 1 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ 0 & 1 & 2w^T(t,\Omega,\lambda) & 2v^T(t,\Omega,\lambda) \\ -2v(t,\Omega,\lambda) & 0_{n_E\times 1} & I_{n_E\times n_E} & 0_{n_E\times n_E} \\ 2w(t,\Omega,\lambda) & 0_{n_E\times 1} & 0_{n_E\times n_E} & I_{n_E\times n_E} \end{pmatrix} \tag{4.95}$$

where we used that $-v(t,\Omega,\lambda) + \cos(\Omega t)v(t,\Omega,\lambda) - \sin(\Omega t)w(t,\Omega,\lambda) = -2v(t,\Omega,\lambda)$ as well as $w(t,\Omega,\lambda) + \sin(\Omega t)v(t,\Omega,\lambda) + \cos(\Omega t)w(t,\Omega,\lambda) = 2w(t,\Omega,\lambda)$.

Second, define the operator

$$K_\kappa := Q \sum_i \left(\mathsf{Re}(\kappa_i)Q_i + \mathsf{Im}(\kappa_i)P_i\right) = \frac{1}{2} \sum_{j,k=1}^{2(1+n_E)} B_{jk}(\kappa) R_j R_k \tag{4.96}$$

where $\kappa_i \in \mathbb{C}$ are complex numbers for $i \in \{1, 2, \ldots, n_E\}$ and where the symmetric matrix $B(\kappa)$ is given by

$$B(\kappa) := \begin{pmatrix} 0 & 0 & \mathsf{Re}(\kappa)^T & \mathsf{Im}(\kappa)^T \\ 0 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ \mathsf{Re}(\kappa) & 0_{n_E\times 1} & 0_{n_E\times n_E} & 0_{n_E\times n_E} \\ \mathsf{Im}(\kappa) & 0_{n_E\times 1} & 0_{n_E\times n_E} & 0_{n_E\times n_E} \end{pmatrix} \in \mathbb{C}^{2(1+n_E)\times 2(1+n_E)} \tag{4.97}$$

and depends on the two real vectors $\mathsf{Re}(\kappa), \mathsf{Im}(\kappa) \in \mathbb{R}^{n_E}$. A simple computation shows that the symplectic matrix associated with the unitary e^{iK_κ} generated by the Hamiltonian K_κ is given by

$$e^{-JB(\kappa)} = \begin{pmatrix} 1 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ 0 & 1 & \mathsf{Re}(\kappa)^T & \mathsf{Im}(\kappa)^T \\ -\mathsf{Im}(\kappa) & 0_{n_E\times 1} & I_{n_E\times n_E} & 0_{n_E\times n_E} \\ \mathsf{Re}(\kappa) & 0_{n_E\times 1} & 0_{n_E\times n_E} & I_{n_E\times n_E} \end{pmatrix} \ . \tag{4.98}$$

A comparison between Eq. (4.95) and Eq. (4.98) shows that $e^{tJ\tilde{A}}e^{tJA} = e^{-JB(\kappa)}$ for the choice of parameters

$$\mathsf{Re}(\kappa) = 2w(t,\Omega,\lambda) \qquad , \qquad \mathsf{Im}(\kappa) = 2v(t,\Omega,\lambda) \ . \tag{4.99}$$

At the level of the unitaries this implies that $U^\dagger V(t) = e^{iK_\kappa}$ for κ defined in Eq. (4.99) where $K_\kappa = K(t)$ from Eq. (4.92).

For the other equation $V(t)U^\dagger = e^{-iK(t)}$, the proof is similar. One first computes

$$V(t)U^\dagger = e^{itH^{\mathrm{orig}}}Ue^{-itH^{\mathrm{orig}}}U^\dagger = e^{itH^{\mathrm{orig}}}e^{-itUH^{\mathrm{orig}}U^\dagger} = e^{itH^{\mathrm{orig}}}e^{it\tilde{H}}$$

where $\tilde{H} \coloneqq -UH^{\mathrm{orig}}U^\dagger = -U^\dagger H^{\mathrm{orig}}U$ as before. A comparison between

$$e^{-tJA}e^{-tJ\tilde{A}} = \begin{pmatrix} 1 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ 0 & 1 & -2w^T(t,\Omega,\lambda) & -2v^T(t,\Omega,\lambda) \\ 2v(t,\Omega,\lambda) & 0_{n_E\times 1} & I_{n_E\times n_E} & 0_{n_E\times n_E} \\ -2w(t,\Omega,\lambda) & 0_{n_E\times 1} & 0_{n_E\times n_E} & I_{n_E\times n_E} \end{pmatrix}$$

and Eq. (4.98) shows that $V(t)U^\dagger = e^{itH^{\mathrm{orig}}}e^{it\tilde{H}} = e^{iK_{-\kappa}}$ for the parameters κ from Eq. (4.99). Since $K_{-2w(t,\Omega,\lambda)-2iv(t,\Omega,\lambda)} = -K_{2w(t,\Omega,\lambda)+2iv(t,\Omega,\lambda)} =: -K(t)$, this implies the desired equation $V(t)U^\dagger = e^{-iK(t)}$. ■

Lemma 4.23 *(Step 2: resulting evolution).* The resulting evolution $U^{\mathrm{res}}(T)$ can be written as

$$U^{\mathrm{res}}(T) = e^{-iTH^{\mathrm{orig}}}e^{-iK^{\mathrm{res}}(T)}e^{i\varphi^{\mathrm{res}}(T)} \tag{4.100}$$

for some $\varphi^{\mathrm{res}}(T) \in \mathbb{R}$ and a Hamiltonian

$$K^{\mathrm{res}}(T) \coloneqq \frac{1}{\sqrt{2}}\sum_{i=1}^{n_E}\frac{\lambda_i}{\omega_i}\left(f_L(\omega_i T)a_i^\dagger + \overline{f_L(\omega_i T)}a_i\right) \ , \tag{4.101}$$

$$f_L(\omega_i T) \coloneqq \sum_{j=1}^{L}(-1)^j f(\omega_i t_j) \ , \tag{4.102}$$

for $f(z) \coloneqq 2i(e^{iz}-1)$ from Eq. (4.93). □

Proof. The resulting evolution $U^{\mathrm{res}}(T)$ is defined by Eq. (4.81). A recursive identification of the operators $V(t_1), V(t_2), \cdots, V(t_L)$ from right to left in Eq. (4.81) implies that

$$\begin{aligned} U^{\mathrm{res}}(T) &= e^{-i(T-t_L)H^{\mathrm{orig}}}U\cdots e^{-it_2H^{\mathrm{orig}}}\underbrace{e^{it_1H^{\mathrm{orig}}}Ue^{-it_1H^{\mathrm{orig}}}}_{=V(t_1)} \\ &= e^{-i(T-t_L)H^{\mathrm{orig}}}U\cdots e^{-it_3H^{\mathrm{orig}}}\underbrace{e^{it_2H^{\mathrm{orig}}}Ue^{-it_2H^{\mathrm{orig}}}}_{=V(t_2)}V(t_1) \\ &= e^{-iTH^{\mathrm{orig}}}V(t_L)\cdots V(t_2)V(t_1) \ . \end{aligned} \tag{4.103}$$

To further compute the resulting evolution $U^{\mathrm{res}}(T)$, one may use the form of $V(t)$ from in Lemma 4.22. Inserting $V(t) = Ue^{iK(t)} = e^{-iK(t)}U$ for $K(t)$ from (4.92) into Eq. (4.103) gives

$$\begin{aligned} U^{\mathrm{res}}(T) &= e^{-iTH^{\mathrm{orig}}}e^{-iK(t_L)}U^2e^{iK(t_{L-1})}\cdots U^2e^{iK(t_3)}e^{-iK(t_2)}U^2e^{iK(t_1)} \\ &= e^{-iTH^{\mathrm{orig}}}e^{-iK(t_L)}e^{iK(t_{L-1})}\cdots e^{iK(t_3)}e^{-iK(t_2)}e^{iK(t_1)} \\ &= e^{-iTH^{\mathrm{orig}}}\prod_{j=1}^{L}e^{-i(-1)^jK(t_j)} \ . \end{aligned} \tag{4.104}$$

Note that we used that $U^2 = I$ as well as the fact that L is even which is necessary for the condition $U^{\text{control}}(T) = I$ to hold.

Let us now show that $U^{\text{res}}(T)$ from (4.104) satisfies $U^{\text{res}}(T) = e^{-iTH^{\text{orig}}} e^{-iK_\kappa} e^{i\varphi^{\text{res}}(T)}$ for some $\varphi^{\text{res}}(T) \in \mathbb{R}$ and a suitably chosen Hamiltonian K_κ of the form of Eq. (4.96).

To this end, consider the operator $K(t)$ from (4.92). It commutes with $K(s)$ up to an imaginary number since they satisfy the commutation relation

$$[K(t), K(s)] = \frac{1}{2} \sum_i \frac{\lambda_i^2}{\omega_i^2} \left(\overline{f(\omega_i t)} f(\omega_i s) - f(\omega_i t) \overline{f(\omega_i s)} \right) =: i\varphi(t,s) I$$

for all $t, s \in \mathbb{R}$, where $\varphi : [0,T] \times [0,T] \to \mathbb{R}$ is a real-valued function. As a consequence, all further nested commutators of $K(t)$ vanish, e.g., $[K(t), [K(s), K(r)]] = 0$ for all $r, s, t \in \mathbb{R}$ and the Baker-Campbell-Hausdorff formula $e^A e^B = e^{A+B} e^{\frac{1}{2}[A,B]}$ therefore applies for $A = K(t)$ and $B = K(s)$. Recursive insertion of the BCH formula into Eq. (4.104) yields

$$\begin{aligned} U^{\text{res}}(T) &= e^{-iTH^{\text{orig}}} e^{-iK(t_L)} \cdots e^{iK(t_3)} e^{-iK(t_2)+iK(t_1)} e^{\frac{1}{2}[K(t_2),K(t_1)]} \\ &= e^{-iTH^{\text{orig}}} e^{-i\sum_{j=1}^{L}(-1)^j K(t_j)} e^{i\varphi^{\text{res}}(T)} \end{aligned} \tag{4.105}$$

where $\varphi^{\text{res}}(T) \coloneqq \sum_{k=1}^{L-1} \sum_{l=1}^{k} (-1)^{k+l} \varphi(t_{k+1}, t_l)$.

The exact form of $\varphi^{\text{res}}(T)$ is irrelevant since it enters into $U^{\text{res}}(T)$ as a complex phase that vanishes if an operator (such as $\rho_S \otimes \rho_E(\beta)$) is conjugated with $U^{\text{res}}(T)$. The exponent in the second term of (4.105) can be rewritten in terms of a complex vector κ as a Hamiltonian K_κ from Eq. (4.96). It takes the form

$$-i \sum_{j=1}^{L} (-1)^j K(t_j) = -\frac{1}{\sqrt{2}} \sum_{k=1}^{n_E} \frac{\lambda_k}{\omega_k} \left(f_L(\omega_k T) a_k^\dagger + \overline{f_L(\omega_k T)} a_k \right) , \tag{4.106}$$

where $f_L(\omega_i T)$ is defined as in Eq. (4.102) for $i = 1, 2, \ldots, n_E$. This finishes the proof since Eq. (4.106) coincides with the desired expression from Eq. (4.101). ■

In the third step, let us finalise the proof of Theorem 4.21.

Step 3: Proof of Theorem 4.21. By Lemma 4.23, the resulting evolution is of the form of Eq. (4.100) where $K^{\text{res}}(T) = K_{\tilde{\kappa}}$ from Eq. (4.96) for the parameters $\tilde{\kappa} \coloneqq \Omega^{-1} f_L(\Omega T) \lambda$, i.e.,

$$\mathsf{Re}(\tilde{\kappa}_i) = -2\frac{\lambda_i}{\omega_i} \sum_{j=1}^{L} (-1)^j \sin(\omega_i t_j) \ , \ \mathsf{Im}(\tilde{\kappa}_i) = 2\frac{\lambda_i}{\omega_i} \sum_{j=1}^{L} (-1)^j \left(\cos(\omega_i t_j) - 1 \right) , \tag{4.107}$$

for $i = 1, 2, \ldots, n_E$. As a consequence, the resulting evolution $U^{\text{res}}(T)$ is (up to the complex phase) associated with the symplectic matrix $S^{\text{res}}(T) = e^{TJA} e^{JB(\tilde{\kappa})}$ for a symmetric matrix $B(\tilde{\kappa})$ of the form of Eq. (4.97) such that

$$K^{\text{res}}(T) = K_{\tilde{\kappa}} = \frac{1}{2} \sum_{j,k=1}^{2(1+n_E)} B_{jk}(\tilde{\kappa}) R_j R_k \ .$$

More precisely, the latter matrix $B(\tilde{\kappa})$ depends on the parameter $\tilde{\kappa}$. Direct computation shows that

$$S^{\text{res}}(T) = e^{TJA} e^{JB(\tilde{\kappa})} = \begin{pmatrix} 1 & 0 & 0_{1\times n_E} & 0_{1\times n_E} \\ x^{\text{res}} & 1 & a^T & b^T \\ c & 0_{n_E\times 1} & \cos(\Omega T) & \sin(\Omega T) \\ d & 0_{n_E\times 1} & -\sin(\Omega T) & \cos(\Omega T) \end{pmatrix} \tag{4.108}$$

where

$$\begin{aligned} x^{\mathrm{res}} &\coloneqq x(T,\Omega,\lambda) + w^T(T,\Omega,\lambda)\mathsf{Im}(\tilde{\kappa}) - v^T(T,\Omega,\lambda)\mathsf{Re}(\tilde{\kappa}) \ , \\ a &\coloneqq -\mathsf{Re}(\tilde{\kappa}) + w(T,\Omega,\lambda) \ , \\ b &\coloneqq -\mathsf{Im}(\tilde{\kappa}) + v(T,\Omega,\lambda) \ , \\ c &\coloneqq v(T,\Omega,\lambda) + \cos(\Omega T)\mathsf{Im}(\tilde{\kappa}) - \sin(\Omega T)\mathsf{Re}(\tilde{\kappa}) \ , \\ d &\coloneqq w(T,\Omega,\lambda) - \sin(\Omega T)\mathsf{Im}(\tilde{\kappa}) - \cos(\Omega T)\mathsf{Re}(\tilde{\kappa}) \ , \end{aligned} \tag{4.109}$$

for $x(T,\Omega,\lambda)$, $v(T,\Omega,\lambda)$, and $w(T,\Omega,\lambda)$ from Eqs. (4.87), (4.85) and (4.86), respectively, and $\tilde{\kappa}$ from Eq. (4.107).

We will use this symplectic matrix to compute the covariance matrix of the resulting state. Since the initial state $\rho_S \otimes \rho_E(\beta)$ is of product form, its covariance matrix is given by the direct sum $V_S \oplus V_E$ of the system's state's covariance matrix $V_S \in \mathbb{R}^{2\times 2}$ and the covariance matrix

$$V_E = \begin{pmatrix} \mathrm{diag}\left(\coth\left(\frac{\beta\omega_1}{2}\right),\dots,\coth\left(\frac{\beta\omega_{n_E}}{2}\right)\right) & 0_{n_E\times n_E} \\ 0_{n_E\times n_E} & \mathrm{diag}\left(\coth\left(\frac{\beta\omega_1}{2}\right),\dots,\coth\left(\frac{\beta\omega_{n_E}}{2}\right)\right) \end{pmatrix}$$

of the environment's thermal state. Let us denote the covariance matrix of the resulting state $\rho^{\mathrm{res}} = U^{\mathrm{res}}(T)(\rho_S \otimes \rho_E)U^{\mathrm{res}}(T)^\dagger$ by V^{res}. Its the principal (2×2)-part – defined as $V_S^{\mathrm{res}} \coloneqq (V^{\mathrm{res}})_{2\times 2} = \left(S^{\mathrm{res}}(T)(V_S \oplus V_E)(S^{\mathrm{res}}(T))^T\right)_{2\times 2}$ – is computed to be

$$V_S^{\mathrm{res}} = S_{11}^{\mathrm{res}}(T)V_S(S_{11}^{\mathrm{res}}(T))^T + S_{12}^{\mathrm{res}}(T)V_E(S_{12}^{\mathrm{res}}(T))^T$$

where

$$S_{11}^{\mathrm{res}}(T) = \begin{pmatrix} 1 & 0 \\ x^{\mathrm{res}} & 1 \end{pmatrix} \in \mathbb{R}^{2\times 2} \quad , \quad S_{12}^{\mathrm{res}}(T) = \begin{pmatrix} 0_{1\times n_E} & 0_{1\times n_E} \\ a^T & b^T \end{pmatrix} \in \mathbb{R}^{2\times 2n_E} \ ,$$

are the upper left 2×2 block and the upper right $2\times 2n_E$ block of the resulting evolution $S^{\mathrm{res}}(T)$ from Eq. (4.108), respectively. A short calculation shows that

$$\begin{aligned} V_S^{\mathrm{res}} &\coloneqq S_{11}^{\mathrm{res}}(T)V_S(S_{11}^{\mathrm{res}}(T))^T + S_{12}^{\mathrm{res}}(T)V_E(S_{12}^{\mathrm{res}}(T))^T \\ &= \begin{pmatrix} 1 & 0 \\ x^{\mathrm{res}} & 1 \end{pmatrix} V \begin{pmatrix} 1 & 0 \\ x^{\mathrm{res}} & 1 \end{pmatrix}^T + \begin{pmatrix} 0_{1\times n_E} & 0_{1\times n_E} \\ a^T & b^T \end{pmatrix} V_E \begin{pmatrix} 0_{n_E\times 1} & a \\ 0_{n_E\times 1} & b \end{pmatrix} \\ &= \begin{pmatrix} 1 & 0 \\ x^{\mathrm{res}} & 1 \end{pmatrix} V \begin{pmatrix} 1 & 0 \\ x^{\mathrm{res}} & 1 \end{pmatrix}^T + \begin{pmatrix} 0 & 0 \\ 0 & y^{\mathrm{res}} \end{pmatrix} \end{aligned}$$

where

$$y^{\mathrm{res}} = \sum_{e\in\{a,b\}} e^T \mathrm{diag}\left(\coth\left(\frac{\beta\omega_1}{2}\right),\dots,\coth\left(\frac{\beta\omega_{n_E}}{2}\right)\right) e = \sum_{i=1}^{n_E}(a_i^2+b_i^2)\coth\left(\frac{\beta\omega_i}{2}\right). \tag{4.110}$$

For $i = 1,\dots,n_E$, we can furthermore identify

$$\begin{aligned} a_i &= 2\frac{\lambda_i}{\omega_i}\sum_{j=1}^{L}(-1)^j\sin(\omega_i t_j) - \frac{\lambda_i}{\omega_i}\sin(\omega_i T) = \frac{\lambda_i}{\omega_i}\mathsf{Im}(y_L(\omega_i T)) \ , \\ b_i &= -2\frac{\lambda_i}{\omega_i}\sum_{j=1}^{L}(-1)^j\big(\cos(\omega_i t_j)-1\big) + \frac{\lambda_i}{\omega_i}\big(\cos(\omega_i T)-1\big) = -\frac{\lambda_i}{\omega_i}\mathsf{Re}(y_L(\omega_i T)) \ , \end{aligned}$$

where the real and imaginary parts of $y_L(\omega_j T)$ are computed from Eq. (4.77). Inserting this into y^{res} from Eq. (4.110) gives

$$y^{\text{res}} = \sum_{i=1}^{n_E} \left(\mathsf{Im}(y_L(\omega_i T))^2 + \mathsf{Re}(y_L(\omega_i T))^2\right) \coth\left(\frac{\beta\omega_i}{2}\right)$$

and hence y^{res} as in Theorem 4.21. Another direct computation yields that x^{res} from Eq. (4.109) is given by

$$\begin{aligned} x^{\text{res}} &= x(T,\Omega,\lambda) + w^T(T,\Omega,\lambda)\mathsf{Im}(\tilde{\kappa}) - v^T(T,\Omega,\lambda)\mathsf{Re}(\tilde{\kappa}) \\ &= \sum_i \frac{\lambda_i^2}{\omega_i^2}\Big(\omega_i T - \sin(\omega_i T) - 2\sin(\omega_i T)\sum_{j=1}^{L}(-1)^j(\cos(\omega_i t_j) - 1) \\ &\qquad + 2(\cos(\omega_i T) - 1)\sum_{j=1}^{L}(-1)^j \sin(\omega_i t_j)\Big) \\ &= \sum_i \frac{\lambda_i^2}{\omega_i^2}\Big(\omega_i T - \sin(\omega_i T)\mathsf{Im}(y_L(\omega_i T))(\cos(\omega_i T) - 1) - \mathsf{Re}(y_L(\omega_i T))\sin(\omega_i T)\Big) \end{aligned}$$

using the definitions of $y_L(z)$ and that $\sum_{j=1}^{L}(-1)^j = 0$ since L is even. ■

Consequences of Theorem 4.21

We have thus shown that in this model, the covariance matrix of the system undergoes the transformation

$$V_S \mapsto V_S^{\text{res}}(T) = S V_S S^T + y^{\text{res}} \qquad \text{where} \qquad S = \begin{pmatrix} 1 & 0 \\ x^{\text{res}} & 1 \end{pmatrix} \in \mathsf{Sp}(2,\mathbb{R})$$

is a symplectic matrix depending on the scalars x^{res} from Eq. (4.109) and y^{res}. More precisely, the latter is given by

$$\begin{aligned} y^{\text{res}} = \chi(T) &:= \int_0^\infty \frac{S_\beta(\omega)}{\omega^2}|y_L(\omega T)|^2 d\omega = \int_0^\infty \frac{J(\omega)}{\omega^2}|y_L(\omega T)|^2 d\omega \\ &= \sum_{i=1}^{n_E} \frac{\lambda_i^2}{\omega_i^2}|y_L(\omega_i T)|^2 \coth\left(\frac{\beta\omega_i}{2}\right) . \end{aligned}$$

The quantity y^{res} plays the role of a form of a residual: The smaller it is, the closer the map $V \mapsto V_S^{\text{res}}(T)$ is to a conjugation with a symplectic matrix. If $\chi(T) = 0$, then $V_S \mapsto S V_S S^T$ and the map $\rho_S \mapsto \text{tr}_E(U^{\text{res}}(T)(\rho_S \otimes \rho_E)U^{\text{res}}(T)^\dagger)$ corresponds to a Gaussian unitary channel. More precisely, under this condition one obtains $U^{\text{res}}(T) = U_S \otimes U_E$, i.e., the pulse sequence would achieve exact decoherence suppression. The quantity $\chi(T)$ quantifies the error of bosonic decoherence suppression. Since this quantity is exactly the same as Eq. (4.76) – quantifying the efficiency of DD in the spin-boson setting – we conclude that the discussion of optimality of pulse times directly translates from the qubit to the bosonic setting. This parameter $\chi(T)$ is defined as the overlap between the noise spectrum $S(\omega)$ and the function $|y_L(T)|$ where the latter quantity depends on the pulse times. In the qubit setting, the optimality of pulse times depends on the UV (i.e., high frequency) cutoff of $S(\omega)$. We have shown in Theorem 4.21 that this dependence on certain properties of the noise spectrum directly translates to the CV setting.

Pasini and Uhrig [127] found that in the presence of a finite number of environment modes or a hard high frequency cutoff, the Uhrig times are optimal. Furthermore, if the noise spectrum has a soft high frequency cutoff, then the optimal pulse times resemble periodic DD (i.e., with equidistant pulse times), more precisely they correspond to the Carr-Purcell-Meiboom-Gill (CPMG) cycle [34, 112]. By Theorem 4.21, this behaviour also applies to the optimality of pulse timings of the bosonic model discussed in this section.

4.7 Relating qubit to bosonic systems

In this section, we exhibit a similarity between $m+1$-qubit systems and CV systems of 2^m modes. This is the key proof ingredient for the Nth order bosonic homogenisation property of the novel schemes proposed in Section 4.8 since these schemes are derived from Nth order multi-qubit DD pulse sequences. More precisely, we will relate certain properties of the Lie group associated with an $m+1$-qubit system, i.e., the unitary group $\mathsf{U}(2^{m+1})$, and its Lie algebra, to the Lie group of the 2^m mode phase space, i.e., the symplectic group $\mathsf{Sp}(2\cdot 2^m, \mathbb{R})$, and its Lie algebra.

In Section 4.7.1, we introduce the operators on the 2^m-mode phase space which play a major role in this relation, and investigate some of their properties. In Section 4.7.2 we relate them to the Pauli matrices in the multi-qubit setting.

4.7.1 Operators on the symplectic group and its Lie algebra

In analogy to the Pauli matrices on multi-qubit systems we define certain symplectic matrices on the 2^m-mode phase space that satisfy similar commutation relations.

Definition 4.24 *(Symplectic 'Paulis').* On $\mathbb{R}^2$ define the matrices

$$S_{(0,0)} \coloneqq \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad S_{(1,1)} = y \coloneqq \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix},$$
$$S_{(1,0)} = x \coloneqq \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad S_{(0,1)} = z \coloneqq \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Furthermore, on $\mathbb{R}^{2\cdot 2^m} \cong (\mathbb{R}^2)^{\otimes(m+1)}$ set

$$S_\alpha \coloneqq S_{a_0} \otimes S_{a_1} \otimes \cdots \otimes S_{a_m} \qquad \text{for } \alpha \coloneqq (a_0, a_1, \ldots, a_m) \in (\mathbb{Z}_2^2)^{m+1}, \tag{4.111}$$

where $a_k \in \mathbb{Z}_2^2$ for every $k = 0, \ldots, m$. Define the subset $\Gamma \subset (\mathbb{Z}_2^2)^{m+1}$ as

$$\Gamma \coloneqq \left\{\alpha \in (\mathbb{Z}_2^2)^{m+1} \mid (\delta(\alpha) + \delta_{a_0,(0,1)} + \delta_{a_0,(1,0)}) \mod 2 = 1\right\},$$

where $\delta_{a_0,(0,1)}$ denotes the Kronecker delta for two variables and

$$\delta(\alpha) \coloneqq |\{j \in \{0, 1 \ldots, m\} \mid a_j = (1,1)\}|,$$

as well as the subset $\tilde{\Gamma} \subset (\mathbb{Z}_2^2)^{m+1}$

$$\tilde{\Gamma} \coloneqq \left\{\beta \in (\mathbb{Z}_2^2)^{m+1} \mid b_0 = (0,0) \text{ or } b_0 = (1,1)\right\}.$$

Here, Greek letters are used for elements of $(\mathbb{Z}_2^2)^{m+1}$ and Latin letters for their respective components, e.g. $\alpha \coloneqq (a_0, a_1, \ldots, a_m) \in (\mathbb{Z}_2^2)^{m+1}$ and $\beta \coloneqq (b_0, b_1, \ldots, b_m) \in (\mathbb{Z}_2^2)^{m+1}$.

The matrices $S_\alpha \in \mathbb{R}^{2\cdot 2^m \times 2\cdot 2^m}$ defined above may be associated with a quantum system of 2^m bosonic modes since the phase space of the latter is given by $\mathbb{R}^{2\cdot 2^m}$. Hence it will be a first step to analyse which of these matrices are elements of the symplectic group $\mathsf{Sp}(2\cdot 2^m, \mathbb{R})$ and its Lie algebra $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$, respectively. In terms of the S_α, the matrix J_{2^m} which defines the symplectic form, can be expressed as

$$J_{2^m} = \begin{pmatrix} 0 & I_{2^m} \\ -I_{2^m} & 0 \end{pmatrix} = -S_{(1,1)} \otimes S_{(0,0)} \otimes \cdots \otimes S_{(0,0)} =: -S_\gamma \ , \tag{4.112}$$

where $\gamma = (c_0, \ldots, c_m)$ satisfies $c_0 = (1,1)$ and $c_i = (0,0)$ for every $i = 1, \ldots, m$.

The four matrices I, x, y, z are all orthogonal and I, x, z are symmetric whereas y is antisymmetric. Although they are not all symplectic, some of them are (what we will show in Theorem 4.25). Note that these matrices are closely related to the single-qubit Pauli matrices since they satisfy $S_{(0,0)} = \sigma_{(0,0)}$, $S_{(1,0)} = \sigma_x = \sigma_{(1,0)}$, $S_{(1,1)} = -i\sigma_y = -i\sigma_{(1,1)}$ and $S_{(0,1)} = \sigma_z = \sigma_{(0,1)}$ with the notation for Paulis from Eq. (4.37). We will show that they also satisfy similar commutation relations as the Pauli matrices.

The following theorem concerns the matrices S_α from Eq. (4.111) and states some of their properties. Note that these properties are similar to those obeyed by the multi-qubit Pauli matrices on the unitary group and its associated Lie algebra (cf. Table 4.2).

Theorem 4.25 *(Relation between qubit and bosonic systems).* For $m \in \mathbb{N}$, consider the symplectic group $\mathsf{Sp}(2\cdot 2^m, \mathbb{R})$ and its associated Lie algebra $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$ from Definition 2.29. Let $\Gamma, \tilde{\Gamma}$ be the sets and let $\{S_\alpha\}_{\alpha\in(\mathbb{Z}_2^2)^{m+1}}$ be the matrices from Definition 4.24. Then the following statements hold.

(a) The family $\{S_\alpha\}_{\alpha\in\Gamma}$ forms a basis of the Lie algebra $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$.

(b) The family $\{S_\beta\}_{\beta\in\tilde{\Gamma}}$ are orthogonal symplectic matrices, i.e.,

$$S_\beta \in \mathsf{Sp}(2\cdot 2^m, \mathbb{R}) \cap \mathsf{O}(2\cdot 2^m, \mathbb{R}) \qquad \text{for every } \beta \in \tilde{\Gamma} \ .$$

(c) They satisfy the commutation relations

$$S_\alpha^{-1} S_\beta S_\alpha = (-1)^{\langle\alpha,\beta\rangle} S_\alpha \qquad \text{for all } \alpha \in \Gamma, \beta \in \tilde{\Gamma} \tag{4.113}$$

where the quantity $\langle\cdot,\cdot\rangle$ is the symplectic inner product as in Eq. 4.43, i.e., given by

$$\langle\alpha,\beta\rangle \coloneqq \sum_{j=0}^{m} a_j^T \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} b_j = \alpha^T J_{2^m} \beta \ .$$

□

Proof. (a) The proof is divided into three steps. We first show that $S_\alpha \in \mathfrak{sp}(2\cdot 2^m, \mathbb{R})$ for all $\alpha \in \Gamma$; the second part proves that all elements of $\{S_\alpha\}_{\alpha\in\Gamma}$ are linearly independent and third, we show that the number of elements in Γ is equal to the dimension of the Lie algebra $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$ when the latter is regarded as a vector space over $\mathbb{R}$. In summary, this shows that the family $\{S_\alpha\}_{\alpha\in\Gamma}$ forms a basis of $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$.

Let us start with some useful computations. Consider the commutation and anticommutation relations between the matrices $S_{(1,0)}$, $S_{(1,1)}$, $S_{(0,1)}$, and $S_{(0,0)}$ which are

$$\{S_{(1,0)}, S_{(1,1)}\} = 0 = \{S_{(1,0)}, S_{(0,1)}\} = \{S_{(0,1)}, S_{(1,1)}\} \tag{4.114}$$

$$[S_{(0,0)}, S_{(0,1)}] = 0 = [S_{(0,0)}, S_{(1,1)}] = [S_{(0,0)}, S_{(1,0)}] \ . \tag{4.115}$$

Furthermore, recall that by Eq. (4.112), the symplectic matrix J_{2^m} can be written as the tensor product of $-S_{(1,1)}$ on the first factor and $S_{(0,0)}$ on the other m factors. By Eqs. (4.114) and (4.115), it satisfies

$$\{S_{(1,0)} \otimes S_{(a_1,\dots,a_m)}, J_{2^m}\} = 0 = \{S_{(1,0)} \otimes S_{(a_1,\dots,a_m)}, J_{2^m}\} \ , \tag{4.116}$$

$$[S_{(0,0)} \otimes S_{(a_1,\dots,a_m)}, J_{2^m}] = 0 = [S_{(0,0)} \otimes S_{(a_1,\dots,a_m)}, J_{2^m}] \ . \tag{4.117}$$

For the first step of the proof, let $\alpha \in \Gamma$ be given and consider the corresponding matrix $S_\alpha \in \mathbb{R}^{2\cdot 2^m \times 2\cdot 2^m}$. We use Eqs. (4.116) and (4.117) as well as the relations

$$S_{(0,0)}^T = S_{(0,0)}, \quad S_{(1,0)}^T = S_{(1,0)}, \quad S_{(1,1)}^T = -S_{(1,1)}, \quad S_{(0,1)}^T = S_{(0,1)} \ ,$$

to compute the term

$$\begin{aligned} S_\alpha^T J_{2^m} + J_{2^m} S_\alpha &= (-1)^{\delta(\alpha)} S_\alpha J_{2^m} + J_{2^m} S_\alpha \\ &= (-1)^{\delta(\alpha)} (-1)^{\delta_{a_0,(1,0)} + \delta_{a_0,(0,1)}} J_{2^m} S_\alpha + J_{2^m} S_\alpha \\ &= \big((-1)^{\delta(\alpha) + \delta_{a_0,(1,0)} + \delta_{a_0,(0,1)}} + 1\big) J_{2^m} S_\alpha \ . \end{aligned}$$

By definition of the set Γ, we have $\big(\delta(\alpha) + \delta_{a_0,(1,0)} + \delta_{a_0,(0,1)}\big) \mod 2 = 1$ and hence

$$S_\alpha^T J_{2^m} + J_{2^m} S_\alpha = 0 \ .$$

Since the Lie algebra $\mathfrak{sp}(2 \cdot 2^m, \mathbb{R})$ is defined as the set of matrices $M \in \mathbb{R}^{2^m \times 2^m}$ such that $M^T J_{2^m} + J_{2^m} M = 0$, we conclude that $S_\alpha \in \mathfrak{sp}(2 \cdot 2^m, \mathbb{R})$.

Second note that by linear independence of the matrices x, y, z, it is straightforward to see that all elements of $\{S_\alpha\}_{\alpha \in \Gamma}$ are linearly independent.

For the last step of the proof, let us compare the number of elements in the set Γ, i.e. the value $|\Gamma|$, to the dimension of the symplectic Lie algebra $\mathfrak{sp}(2 \cdot 2^m, \mathbb{R})$. The latter is equal to $2 \cdot 2^m + 2^m = 2^{m+1} + 2^m$.

By definition of Γ, the value $|\Gamma|$ is equal to the number of $\alpha = (a_0, \dots, a_m) \in (\mathbb{Z}_2^2)^{m+1}$ such that $\big(\delta(\alpha) + \delta_{a_0,(1,0)} + \delta_{a_0,(0,1)}\big) \mod 2 = 1$, where

$$\delta(\alpha) \coloneqq |\{j \in \{0, \dots, m\} \mid a_j = (1,1)\}| \ .$$

We will use only simple combinatorics to compute this number.

Let $\alpha = (a_0, \dots, a_m) \in \Gamma$. Then by definition of Γ, it satisfies

$$\delta\big((a_1, \dots, a_m)\big) \mod 2 = \begin{cases} 0 & \text{if } a_0 \in \{(0,1), (1,1), (1,0)\} \ , \\ 1 & \text{if } a_0 = (0,0) \ , \end{cases} \tag{4.118}$$

where in analogy to $\delta(\alpha)$, we defined $\delta\big((a_1, \dots, a_m)\big) \coloneqq |\{j \in \{1, 2, \dots, m\} | \, a_j = (1,1)\}|$. Let us denote by e_m and o_m the number of vectors $(b_1, \dots, b_m) \in (\mathbb{Z}_2^2)^m$ such that

$$\delta\big((b_1, \dots, b_m)\big) \mod 2 = 0, \quad \text{respectively,} \quad \delta\big((b_1, \dots, b_m)\big) \mod 2 = 1 \ .$$

We can easily see that $e_1 = 3$ and $o_1 = 1$. For an arbitrary $m \in \mathbb{N}$, these numbers must add up to 4^m i.e, they satisfy $o_m + e_m = 4^m$. They can be recursively defined as

$$e_{m+1} = e_1 e_m + o_1 o_m \qquad , \qquad o_{m+1} = e_1 o_m + o_1 e_m \ , \tag{4.119}$$

which simply reflects the way how even and/or odd numbers add up.

Lemma 4.26. For $m \in \mathbb{N}$, we have $e_m = 2^{2m-1} + 2^{m-1}$. □

Proof. Let us show this by induction over m. For $m = 1$, the claim is consistent with $e_1 = 3$. For the inductive step $m \to m+1$, let us use the relation $o_m = 4^m - e_m$ and $o_1 = 1$ to compute $e_{m+1} = e_1 e_m + o_1(4^m - e_m) = 3e_m + (4^m - e_m) = 2^{2(m+1)-1} + 2^m$. ■

Eq. (4.118) implies that the number of elements in Γ is given by $|\Gamma| = 3e_m + 1o_m$. Using Eq. (4.119), this is equal to $|\Gamma| = e_{m+1}$ which, by Lemma 4.26, is given by $e_{m+1} = 2^{2m+1} + 2^m = 2 \cdot 2^m + 2^m$. The latter expression corresponds to the desired dimension of $\mathfrak{sp}(2 \cdot 2^m, \mathbb{R})$ such that one obtains $|\Gamma| = \dim(\mathfrak{sp}(2 \cdot 2^m, \mathbb{R}))$.

(b) Let $\beta \coloneqq (b_0, b_1, \ldots, b_m) \in \tilde{\Gamma}$. The matrix S_β is orthogonal since all its factors satisfy

$$S_{(0,0)}^T S_{(0,0)} = I_2 = S_{(1,0)}^T S_{(1,0)} = S_{(1,1)}^T S_{(1,1)} = S_{(0,1)}^T S_{(0,1)} = I_2 \ .$$

Note that we did not use any property of Γ, i.e., we showed the more general statement that $S_\alpha \in \mathsf{O}(2 \cdot 2^m, \mathbb{R})$ for every $\alpha \in (\mathbb{Z}_2^2)^{m+1}$.

To show that S_β is symplectic, let us recall that a matrix M is symplectic if and only if $MJ^TM = J$. We use the (anti-)commutation relations (4.116) and (4.117) to compute

$$S_\beta^T J_{2^m} S_\beta = (-1)^{\delta_{b_0,(1,0)} + \delta_{b_0,(0,1)}} S_\beta^T S_\beta J_{2^m} = (-1)^{\delta_{b_0,(1,0)} + \delta_{b_0,(0,1)}} J_{2^m} \ .$$

This is equal to J_{2^m} if and only if $b_0 \in \{(0,0),(1,1)\}$. But by definition of $\beta \in \tilde{\Gamma}$, we have that $b_0 \in \{(0,0),(1,1)\}$. As a consequence

$$S_\beta \in \mathsf{Sp}(2 \cdot 2^m, \mathbb{R}) \cap \mathsf{O}(2 \cdot 2^m, \mathbb{R}) \ .$$

(c) Let us consider the commutation relations between x, y, and z, first. These matrices satisfy $xy = z$, $yz = x$ and $zx = -y$ and hence the relations

$$\begin{aligned} x^T x x &= x \quad , \quad x^T y x = -y \quad , \quad x^T z x = -z \ , \\ y^T x y &= -x \quad , \quad y^T y y = y \quad , \quad y^T z y = -z \ , \\ z^T x z &= -x \quad , \quad z^T y z = -y \quad , \quad z^T z z = z \ . \end{aligned} \tag{4.120}$$

On the multi-mode level this translates to $S_\beta^T S_\alpha S_\beta = (-1)^{\langle \alpha, \beta \rangle} S_\alpha$ for all $\alpha, \beta \in (\mathbb{Z}_2^2)^{m+1}$, where

$$a_j^T \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} b_j = \begin{cases} 0 & \text{if } a_j = b_j \text{ or } a_j = (0,0) \text{ or } b_j = (0,0) \\ 1 & \text{else} \end{cases} \ .$$

This is also true in the special case of $\alpha \in \Gamma$, $\beta \in \tilde{\Gamma}$.

■

4.7.2 Relation between multi-qubit and multi-mode CV operators

Let us now relate multi-mode bosonic systems to multi-qubit systems. More precisely, let $m \in \mathbb{N}_0$. We relate the properties of the symplectic group $\mathsf{Sp}(2 \cdot 2^m, \mathbb{R})$ and its Lie algebra $\mathfrak{sp}(2 \cdot 2^m, \mathbb{R})$ considered in Theorem 4.25 to those of the unitary group $\mathsf{U}(2^{m+1})$ and its Lie algebra $\mathfrak{u}(2^{m+1})$.

$m+1$ qubits	2^m bosonic modes
Hilbert space $\mathcal{H}_S = (\mathbb{C}^2)^{\otimes m+1}$ $\mathsf{U}(2^{m+1})$ $\mathfrak{u}(2^{m+1})$	Phase space $\mathbb{R}^{2\cdot 2^m}$ $\mathsf{Sp}(2\cdot 2^m, \mathbb{R})$ $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$
Multi-qubit Pauli operators on $(\mathbb{C}^2)^{\otimes m+1}$ $\sigma_\alpha = \sigma_{a_0} \otimes \cdots \otimes \sigma_{a_m}$ where $\alpha = (a_0, \ldots, a_m) \in (\mathbb{Z}_2^2)^{m+1}$ are tensor products of the Pauli matrices $\sigma_{(0,0)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $\sigma_{(1,0)} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $\sigma_{(1,1)} = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$, $\sigma_{(0,1)} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	Multi-mode operators on $\mathbb{R}^{2\cdot 2^m}$ $S_\alpha = S_{a_0} \otimes \cdots \otimes S_{a_m}$ where $\alpha = (a_0, \ldots, a_m) \in (\mathbb{Z}_2^2)^{m+1}$ are tensor products of the matrices $S_{(0,0)} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$, $S_{(1,0)} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $S_{(1,1)} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$, $S_{(0,1)} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
$\{\sigma_\alpha\}_{\alpha\in(\mathbb{Z}_2^2)^{m+1}}$ is a basis of $\mathfrak{u}(2^{m+1})$	$\{S_\alpha\}_{\alpha\in\Gamma}$ is a basis of $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$ where $\Gamma \subset (\mathbb{Z}_2^2)^{m+1}$ as in Definition 4.24
$\{\sigma_\beta\}_{\beta\in(\mathbb{Z}_2^2)^{m+1}} \subset \mathsf{U}(2^{m+1})$	$\{S_\beta\}_{\beta\in\tilde{\Gamma}} \subset \mathsf{Sp}(2\cdot 2^m, \mathbb{R}) \cap \mathsf{O}(2^{m+1}, \mathbb{R})$, where $\tilde{\Gamma} \subset (\mathbb{Z}_2^2)^{m+1}$ as in Definition 4.24,
$\sigma_\beta^{-1}\sigma_\alpha\sigma_\beta = (-1)^{\langle\alpha,\beta\rangle}\sigma_\alpha$ for $\alpha, \beta \in (\mathbb{Z}_2^2)^{m+1}$	$S_\beta^{-1}S_\alpha S_\beta = (-1)^{\langle\alpha,\beta\rangle}S_\alpha$ for $\alpha \in \Gamma, \beta \in \tilde{\Gamma}$.

Table 4.2: List of analogous properties between the quantum systems of $m+1$ qubits and 2^m bosonic modes. In both systems, the operators σ_α and S_α satisfy the same commutation relations. We will make use of this similarity to construct bosonic homogenisation sequences from multi-qubit DD sequences. But note that we also observe two differences: the sets $\Gamma \subset (\mathbb{Z}_2^2)^{m+1}$ as well as $\tilde{\Gamma} \subset (\mathbb{Z}_2^2)^{m+1}$, which only occur in the bosonic setting, are restricted compared to $(\mathbb{Z}_2^2)^{m+1}$ in the multi-qubit setting. The relevant restriction in the context of CV homogenisation concerns the set $\tilde{\Gamma}$ as the passive Gaussian unitary pulses correspond to symplectic matrices S_β where $\beta \in \tilde{\Gamma}$. As a consequence, the set of possible pulses (on the symplectic level) is restricted to $\{S_\beta\}_{\beta\in\tilde{\Gamma}}$ compared to $\{\sigma_\beta\}_{\beta\in(\mathbb{Z}_2^2)^{m+1}}$ from which the DD pulses for multiple qubits can be chosen.

On the one hand, we consider a quantum system of 2^m bosonic modes with associated phase space $\mathbb{R}^{2\cdot 2^m}$. The multi-mode operators $S_\alpha \in \mathbb{R}^{2\cdot 2^m \times 2\cdot 2^m}$ satisfy the commutation relations (4.113) by part (c) of Theorem 4.25, i.e.,

$$S_\beta^{-1} S_\alpha S_\beta = (-1)^{\langle\alpha,\beta\rangle} S_\alpha \qquad \text{for all } \alpha \in \Gamma, \beta \in \tilde{\Gamma} \ ,$$

Moreover, the family $\{S_\alpha\}_{\alpha\in\Gamma}$ forms a basis of the Lie algebra $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$ by part (a) and $S_\beta \in \mathsf{Sp}(2\cdot 2^m, \mathbb{R}) \cap \mathsf{O}(2\cdot 2^m, \mathbb{R})$ for every $\beta \in \tilde{\Gamma}$ by part (b).

On the other hand, we have a system of $m+1$ qubits. Recall the multi-qubit Pauli matrices σ_α from Eq. (4.55) for $\alpha \coloneqq (a_0, a_1, \ldots, a_m) \in (\mathbb{Z}_2^2)^{m+1}$ where we use the notation for the single-qubit Pauli matrices from Eqs. (4.37). With this notational convention, the multi-qubit Pauli matrices satisfy the commutation relations

$$\sigma_\beta^T \sigma_\alpha \sigma_\beta = (-1)^{\langle\alpha,\beta\rangle} \sigma_\alpha \qquad \text{for all } \alpha,\ \beta \in (\mathbb{Z}_2^2)^{m+1}, \tag{4.121}$$

where $\{\sigma_\alpha\}_{\alpha\in(\mathbb{Z}_2^2)^{m+1}}$ is a basis of $\mathfrak{u}(2^{m+1})$ and $\sigma_\beta \in \mathsf{U}(2^{m+1})$ for all $\beta \in (\mathbb{Z}_2^2)^{m+1}$.

Table 4.2 shows a direct comparison between the multi-qubit setting and the CV setting. The reader might directly notice the similarity between the commutation relations (4.113)

and (4.121). This similarity is key to our construction of bosonic homogenisation schemes from qubit decoupling schemes: it is exactly these phases (4.113) and (4.121) that appear in the Dyson series of the resulting evolution.

However, there is also a crucial difference: in the CV setting, one considers elements of the two sets Γ and $\tilde{\Gamma}$ which are strict subsets of $(\mathbb{Z}_2^2)^{m+1}$. This restriction can be directly translated to a conceptual difference between DD in the both settings. It is the mathematical reason for the fact that DD in CV cannot be achieved in the same strong as standard DD for finite-dimensional systems, i.e., a general CV system evolution cannot be rendered trivial, only *homogenised*, as first shown in [12]: in our framework, this is due to the fact that the set of possible unitary pulses is restricted, i.e, one can only choose pulses associated with a symplectic matrix from the family $\{S_\beta\}_{\beta\in\tilde{\Gamma}}$. As a consequence, decoherence terms proportional to $S_\gamma = J_{n_S}$ cannot be averaged out by such a pulse sequence.

4.8 Higher order bosonic homogenisation

In this section, we construct higher order bosonic homogenisation sequences in the sense of Definition 4.9.

The pulses and a few of their properties are presented in Section 4.8.1. In Section 4.8.2 we show how bosonic homogenisation schemes of order N on 2^m modes can be obtained from $m+1$-qubit DD of the same order. This construction builds on both the Lie algebraic formulation of DD from Section 4.5 and the relation between multi-qubit and CV systems from Section 4.7. In Section 4.8.3 this procedure is applied to the multi-qubit NUDD scheme (cf. Section 4.4.3). The resulting bosonic nested Uhrig homogenisation schemes are efficient in the sense that they use $(N+1)^{2m+1}$ pulses to homogenise 2^m system modes up to order N.

4.8.1 Bosonic pulses for homogenisation

Let us start with some preliminary remarks on the notation. Since an essential ingredient to our homogenisation scheme construction is the relation between bosonic and multi-qubit systems from Section 4.7, let us assume that the number of system modes n_S is a power of 2, i.e., there exists $m \in \mathbb{N}_0$ such that $n_S = 2^m$. Instead of labelling the system's quadrature operators with indices from 1 to n_S, we will introduce a new label

$$\nu = (v_1, v_2, \ldots, v_m) \in \mathbb{Z}_2^m \coloneqq \{0,1\}^m \tag{4.122}$$

and label the quadrature operators by these bitstrings ν as

$$Q_\nu = Q_{(v_1,v_2,\ldots,v_m)} \quad , \quad P_\nu = P_{(v_1,v_2,\ldots,v_m)} \ , \quad \text{where } v_j \in \{0,1\} \text{ for } j = 1,\ldots,m \ .$$

It will be useful to order the labels (4.122) using the bijective map

$$\mathbb{Z}_2^m \to \{1,\ldots,2^m\} \quad , \quad (v_1,\ldots,v_m) \mapsto 1 + \sum_{k=1}^{m} 2^{m-k} v_k \ ,$$

e.g. when considering matrices in $\mathsf{Sp}(2\cdot 2^m, \mathbb{R})$. Then, the mode operators can be arranged in the vector $R^T = (R_1, \ldots, R_{2^{m+1}})$ as follows:

$$\begin{aligned} R_1 = Q_{(0,\ldots,0,0)} \quad , \quad R_2 = Q_{(0,\ldots,0,1)} \quad , \quad R_3 = Q_{(0,\ldots,1,0)} \ , \ \cdots \ , \quad R_{2^m} = Q_{(1,\ldots,1,1)} \ , \\ R_{2^m+1} = P_{(0,\ldots,0,0)} \ , \ R_{2^m+2} = P_{(0,\ldots,0,1)} \ , \ R_{2^m+3} = P_{(0,\ldots,1,0)} \ , \ \cdots \ , \ R_{2^{m+1}} = P_{(1,\ldots,1,1)} \ . \end{aligned} \tag{4.123}$$

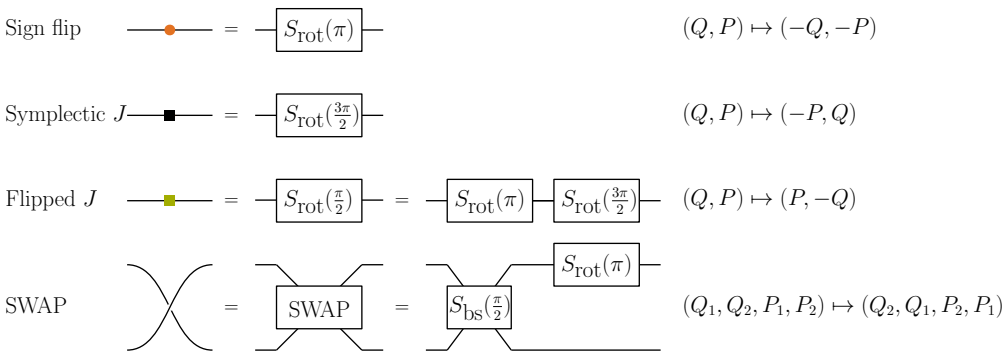


Figure 4.5: Circuit representation of the four basic bosonic gates for the constructed decoherence suppression and homogenisation sequences: the three single-mode gates 'sign flip', J, and sign 'flipped J' are the phase gates $S_{\mathrm{rot}}(\zeta)$ (which are defined in Eq. (2.42) of Section 2.2.5) for the angles $\zeta = \pi, 3\pi/2, \pi/2$, respectively, and the two-mode SWAP gate can be represented as the subsequent application of a two-mode beam splitter $S_{\mathrm{bs}}(\pi/2)$ (cf. Eq. (2.47)) and a phase space rotation $S_{\mathrm{rot}}(\pi)$. They are related to the considered symplectic matrices y_0, x_j and z_j from Lemma 4.28 as depicted in Fig. 4.6.

Since for $\beta \in \tilde{\Gamma}$ the matrices S_β from Definition 4.24 are symplectic and exponentially generated by elements of $\mathfrak{sp}(2\cdot 2^m, \mathbb{R})$ up to a complex phase, the associated Gaussian unitaries form a suitable set for unitary pulses of homogenisation schemes. The following Gaussian unitaries will be relevant as pulses in the context of our bosonic NUDD homogenisation schemes, derived in Section 4.8.3.

Definition 4.27 *(Bosonic pulses).* Let $m \in \mathbb{N}$. Let furthermore the Gaussian unitaries U_{y_0}, U_{x_j} and U_{z_j} for $j = 1, \dots m$ acting on $n_S = 2^m$ bosonic system modes be defined (up to global phases) by their action on the bosonic quadrature operators of the system, i.e., as

$$\begin{aligned}
U_{y_0}^\dagger Q_\nu U_{y_0} &\coloneqq P_\nu \ , \\
U_{y_0}^\dagger P_\nu U_{y_0} &\coloneqq -Q_\nu \ , \\
U_{x_j}^\dagger Q_{(v_1,\dots,v_m)} U_{x_j} &\coloneqq Q_{(v_1,\dots,v_{j-1},1-v_j,v_{j+1},\dots,v_m)} \ , \\
U_{x_j}^\dagger P_{(v_1,\dots,v_m)} U_{x_j} &\coloneqq P_{(v_1,\dots,v_{j-1},1-v_j,v_{j+1},\dots,v_m)} \ , \\
U_{z_j}^\dagger Q_{(v_1,\dots,v_m)} U_{z_j} &\coloneqq (-1)^{v_j} Q_{(v_1,\dots,v_m)} \ , \\
U_{z_j}^\dagger P_{(v_1,\dots,v_m)} U_{z_j} &\coloneqq (-1)^{v_j} P_{(v_1,\dots,v_m)}
\end{aligned}$$

where the bit-strings $\nu = (v_1, \dots, v_m) \in \{0,1\}^m$ label the 2^m bosonic modes.

Let us show a few properties of these unitaries. In order to do this, it is useful to consider these properties at the level of their corresponding symplectic matrices according to the metaplectic representation. It will be convenient to identify the vector space $\mathbb{R}^{2\cdot 2^m}$ with $\mathbb{R} \otimes (\mathbb{R}^2)^{\otimes m}$ on which the matrices S_α are defined, which themselves are tensor products of the (2×2)-matrices I, x, y, z from Definition 4.24.

Furthermore, it is useful to introduce a two-mode SWAP unitary V in analogy to the two-qubit SWAP gate from (2.58). It acts on the quadrature operators of modes 1 and 2 as $V^\dagger Q_1 V = Q_2$, $V^\dagger P_1 V = P_2$ and $V^\dagger Q_2 V = Q_1$, $V^\dagger P_2 V = P_1$, respectively, i.e., it interchanges quadratures of one mode with those of the other, cf. Fig. 4.5.

Lemma 4.28. The unitaries from Definition 4.27 satisfy the following properties:

(a) Let $S_{(b_0,\ldots,b_m)}$ for $(b_0,\ldots,b_m)\in\tilde{\Gamma}$ be the matrices from Definition 4.24. The symplectic matrix associated with U_{y_0} is

$$y_0 \coloneqq y\otimes I_2^{\otimes m} = S_{(b_0,\ldots,b_m)}\ , \tag{4.124}$$

where $b_0=(1,1)$ and all other $b_1=\cdots=b_m=(0,0)$. For $j=1,\ldots m$, the unitaries U_{x_j} and U_{z_j} have associated symplectic matrices

$$x_j \coloneqq I_2^{\otimes j}\otimes x\otimes I_2^{\otimes(m-j)} = S_{(c_0,\ldots,c_m)}\ , \tag{4.125}$$

$$z_j \coloneqq I_2^{\otimes j}\otimes z\otimes I_2^{\otimes(m-j)} = S_{(d_0,\ldots,d_m)}\ , \tag{4.126}$$

respectively, where $c_j=(1,0)$, $d_j=(0,1)$ and $c_k=d_k=(0,0)$ for all $k\neq j$.

(b) They are tensor products of either passive single-mode or passive two-mode Gaussian unitaries.

(c) The associated symplectic matrices y_0, x_j and z_j for $j=1,\ldots m$ are exponentials of elements of $\mathfrak{sp}(2\cdot 2^m,\mathbb{R})$.

□

Proof. (a) Using the ordering 4.123 of the quadratures, one can check by direct computation that the symplectic matrices from (4.124), (4.125) and (4.126) satisfy the equations

$$U_{y_0}^\dagger R_j U_{y_0} = \sum_k \left(S_{\big((1,1),(0,0),\ldots,(0,0)\big)}\right)_{jk} R_k = \sum_{k=1}^{2^{m+1}} (y_0)_{jk} R_k\ ,$$

$$U_{x_i}^\dagger R_j U_{x_i} = \sum_k \Big(S_{\big((0,0),\ldots,(0,0),\ \underbrace{(1,0)}_{\text{position } i},(0,0),\ldots,(0,0)\big)}\Big)_{jk} R_k = \sum_{k=1}^{2^{m+1}} (x_i)_{jk} R_k\ ,$$

$$U_{z_i}^\dagger R_j U_{z_i} = \sum_k \Big(S_{\big((0,0),\ldots,(0,0),\ \underbrace{(0,1)}_{\text{position } i},(0,0),\ldots,(0,0)\big)}\Big)_{jk} R_k = \sum_{k=1}^{2^{m+1}} (z_i)_{jk} R_k\ .$$

for $j=1,\ldots,2^{m+1}$ and $i=0,\ldots,m$.

(b) The associated symplectic matrices y_0, x_j as well as z_j are orthogonal, which translates to passivity at the level of the Gaussian unitaries (cf. Section 2.2.4). They are products from the passive Gaussian single-mode phase rotations and the two-mode beam splitter (described in Section 2.2.5) which we now analyse in more detail.

The unitary U_{y_0} is the 2^m-fold tensor product of $U_{\mathrm{rot}}(3\pi/2)$, i.e., it acts as the single-mode rotation in phase space from Eq. (2.41) for $\zeta=3\pi/2$ on all modes $\nu\in\mathbb{Z}_2^m$, cf. Fig. 4.5).

Note that a two-mode SWAP gate acts on the mode operators as shown in Fig. 4.5. For $j=1,\ldots,m$, the unitary U_{x_j} is the tensor product of 2^{m-1} SWAP operations between pairs of modes. More precisely, it swaps all pairs of modes which differ on the jth entry of their index $\nu=(v_1,\ldots,v_m)\in\{0,1\}^m$, i.e., it interchanges the modes with labels $\nu=(v_1,\ldots,v_j,\ldots,v_m)$ and $\overline{\nu}^j\coloneqq(v_1,\ldots,1-v_j,\ldots,v_m)$. Moreover, it can be

implemented by the subsequent application of a beam splitter operations and a single-mode rotation in the following way: the two-mode SWAP gate can be decomposed as

$$S_{\mathrm{bs}}(\pi/2)\,(I \oplus S_{\mathrm{rot}}(\pi)) = \mathsf{SWAP}$$

where $S_{\mathrm{bs}}(\pi/2)$ is the beam splitter of transmittivity 0, cf. Eq. (2.48), and $S_{\mathrm{rot}}(\pi)$ is a single-mode phase rotation about the angle π, cf. Eq. (2.43).

For $j = 1, \ldots, m$, the unitary U_{z_j} acts as the same single-mode unitary on half of the modes, i.e., the modes with indices ν such that $v_j = 1$, which is the phase rotation

$$U_{\mathrm{rot}}(\pi) = e^{-i\frac{\pi}{2}(Q_\nu^2+P_\nu^2)} \ .$$

Moreover recall that this is the same unitary as used for decoherence suppression in Definition 4.18.

(c) Let us recall the identity $e^{\omega J_1} = \cos(\omega)I_2 + \sin(\omega)J_1$ for $\omega \in \mathbb{R}$ which we already used in Section 2.2.5. We note that on the k-mode phase space $\mathbb{R}^{2k}$ for $k \in \mathbb{R}$, this can be generalised to $e^{\omega J_k} = \cos(\omega)I_{2k} + \sin(\omega)J_k$. This can be used to find the generator of y_0 by computing

$$e^{\pi/2 y_0} = e^{-\pi/2 J_1 \otimes I^{\otimes m}} = \big(\cos(\pi/2)I_2 - \sin(\pi/2)J_1\big) \otimes I^{\otimes m} = y_0 \ .$$

Obviously, $y_0^T J + J y_0 = 0$, i.e., y_0 is generated by $\pi y_0/2 \in \mathfrak{sp}(2 \cdot 2^m, \mathbb{R})$. For $j = 1, \ldots, m$ direct computation yields

$$e^{-\pi/2 y \otimes I_2^{\otimes m} + \pi/2 y \otimes I_2^{\otimes(j-1)} \otimes x \otimes I_2^{\otimes(m-j)}} = e^{\pi/2 y_0(-I_2^{\otimes m+1} + x_j)} = -y_0(y_0 x_j) = x_j$$

and similarly that $e^{-\pi/2 y \otimes I_2^{\otimes m} + \pi/2 y \otimes I_2^{\otimes(j-1)} \otimes z \otimes I_2^{\otimes(m-j)}} = z_j$. This shows that for $j = 1, \ldots, m$, the symplectic matrices x_j and z_j are exponentials of

$$\frac{\pi}{2} y_0(-I_2^{\otimes m+1} + x_j) \quad \text{and} \quad \frac{\pi}{2} y_0(-I_2^{\otimes m+1} + z_j) \ ,$$

respectively, which themselves are elements of $\mathfrak{sp}(2 \cdot 2^m, \mathbb{R})$ since

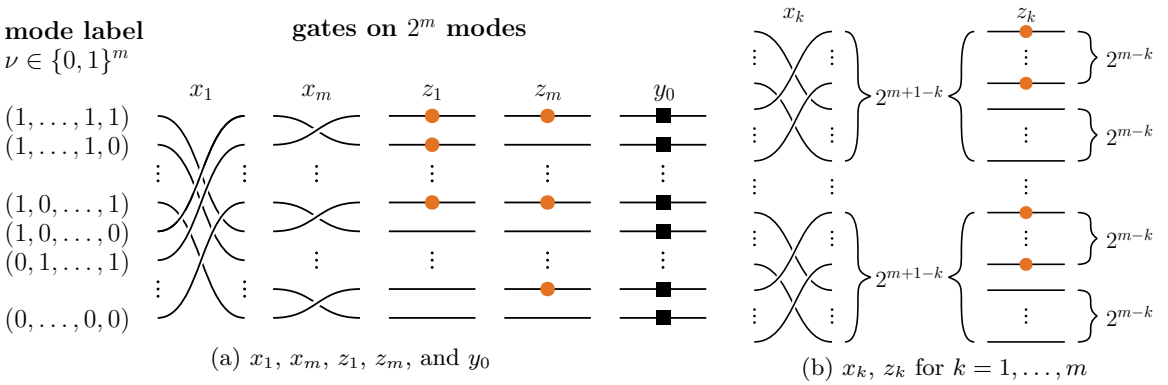
$$\begin{aligned} \big(y_0(-I_2^{\otimes m+1} + x_j)\big)^T J + J y_0(-I_2^{\otimes m+1} + x_j) = 0 \ , \\ \big(y_0(-I_2^{\otimes m+1} + z_j)\big)^T J + J y_0(-I_2^{\otimes m+1} + z_j) = 0 \ , \end{aligned}$$

using the commutation relations (4.120) as well as $y_0 = -J$.

■

Let us discuss one property of the Gaussian unitaries U_{y_0}, U_{x_j}, and U_{z_j} from part (b) of the above proof: namely how they can be written in terms of the basic Gaussian unitaries from Section 2.2.5, i.e, in terms of the phase space rotations and the beam splitter. For $j = 1, \ldots, m$, the U_{x_j}-pulse corresponds to a product of SWAP gates between pairs of modes where the pairings (of swapped modes) are determined by the index j: one swaps the modes for which the index ν differs on the jth entry. Since SWAP gates corresponds to the product of a beam splitter and a phase gate (on one of the modes), the Gaussian unitary U_{x_j} can furthermore be written as

$$U_{x_j} = \prod_{\nu \in \mathcal{V}_j} U_{\mathrm{rot},\overline{\nu}_j}(\pi)\, U_{\mathrm{bs},\nu,\overline{\nu}_j}(\pi/2) = \prod_{\nu \in \mathcal{V}_j} U_{\mathrm{bs},\nu,\overline{\nu}^j}(\pi/2)\, U_{\mathrm{rot},\nu}(\pi) \ .$$



(a) x_1, x_m, z_1, z_m, and y_0 (b) x_k, z_k for $k = 1, \ldots, m$

Figure 4.6: Bosonic pulses for homogenisation sequences (cf. Definition 4.27) and their effect on the 2^m modes for $m \in \mathbb{N}$. The passive Gaussian unitaries $y_0, x_1, \ldots, x_m$ and $z_1, \ldots, z_m$ act on the 2^m modes in terms of the three basic single- and two-mode operations from Fig. 4.5. For $k = 1, \ldots, m$, the pulse x_k acts as a product of SWAP gates between pairs of modes that differ on the kth entry v_k of $\nu = (v_1, \ldots, v_m)$; the pulse z_k acts as the sign flip (a single-mode phase rotation) on the modes ν such that $v_j = 1$ and the pulse y_0 acts as the symplectic J (a different phase rotation) on all modes.

Here, we define the set $\mathcal{V}_j := \{\nu \in \mathbb{Z}_2^m | v_j = 1\}$, i.e., the above product is taken over all indices ν which satisfy $v_j = 1$ and $\overline{\nu}^j$ denotes the mode index that is equal to ν except at the jth position (where it is the opposite of v_j) and $U_{\mathrm{rot},\overline{\nu}_j}(\pi)$ acts as $U_{\mathrm{rot}}(\pi)$ (cf. (2.43)) on the mode with label $\overline{\nu}_j$, Furthermore, $U_{\mathrm{bs},a,b}(\pi/2)$ denotes the beam splitter of transmittivity 0 (cf. Eq. (2.48)) between mode a and b. The unitary U_{y_0} acts as

$$U_{y_0} = \left(U_{\mathrm{rot}}(3\pi/2)\right)^{\otimes(2^{m+1})} .$$

For $j = 1, \ldots, m$, the Gaussian unitary U_{z_j} can be written as

$$U_{z_j} = \prod_{\nu \in \mathcal{V}_j} U_{\mathrm{rot},\nu}(\pi) .$$

The unitaries (more precisely, the circuit representation of their associated symplectic matrices) are presented in Fig. 4.6.

4.8.2 Higher order bosonic homogenisation from higher order multi-qubit DD

The main result of this section relates $(m+1)$-qubit DD schemes to 2^m-mode bosonic homogenisation schemes.

We start with an Nth order DD scheme for $m+1$ qubits. More precisely, the chosen pulse sequence must satisfy two conditions: first, all pulses are tensor products of single qubit Pauli matrices and second, decoupling is achieved in the sense of Theorem 4.14 from Section 4.5. By the second assumption, one may work within the framework of Section 4.5. By the first assumption, the pulses are multi-qubit Paulis, i.e., they can be written as σ_β from Eq. (4.37) for $\beta \in (\mathbb{Z}_2^2)^{m+1}$. We will establish a map which maps these multi-qubit pulses to bosonic pulses such that the latter pulse sequence inherits its decoherence properties from the multi-qubit sequence, i.e., it achieves Nth order bosonic homogenisation.

Let us state a substitution rule

$$\mathsf{U}(2^{m+1}) \ni A \mapsto B \in \mathsf{Sp}(2 \cdot 2^m, \mathbb{R}) \cap \mathrm{O}(2^{m+1}, \mathbb{R})$$

which maps certain unitaries on $m+1$ qubits to symplectic orthogonal matrices on the 2^{m+1}-dimensional phase space. A naive idea for this map would be to choose

$$\sigma_\beta \mapsto S_\beta \quad , \quad \text{for all } \beta \in (\mathbb{Z}_2^2)^{m+1} \tag{4.127}$$

where $S_\beta \in \mathbb{R}^{2\cdot 2^m \times 2\cdot 2^m}$ are the matrices from Definition 4.24. But this would not yield a valid bosonic pulse scheme since there are $\beta \in (\mathbb{Z}_2^2)^{m+1}$ such that S_β is not symplectic. Following the construction from Section 4.7, it is therefore necessary to adapt this simplified idea (4.127): Instead we will consider the map $\sigma_\beta \to S_{\beta'}$ where β' differs from β only on the zeroth entry.

Definition 4.29 *(Substitution rule).* Let $m \in \mathbb{N}$ and $\beta \coloneqq (b_0, b_1, \ldots, b_m) \in (\mathbb{Z}_2^2)^{m+1}$ and let β' be defined as

$$\beta' \coloneqq (b_0', b_1, \ldots, b_m) \quad \text{where} \quad b_0' \coloneqq \begin{cases} (0,0) & \text{if } b_0 \in \{(0,0),(0,1)\} \\ (1,1) & \text{if } b_0 \in \{(1,0),(1,1)\} \end{cases} . \tag{4.128}$$

The *substitution rule* is defined as the map $\mathsf{U}(2^{m+1}) \to \mathsf{Sp}(2 \cdot 2^m, \mathbb{R}) \cap \mathrm{O}(2^{m+1}, \mathbb{R})$,

$$\sigma_\beta \mapsto S_{\beta'} \ .$$

For the Pauli matrices on qubits 0 and $i = 1, \ldots, m$, the substitution rule (4.128) implies

$$\sigma_{x_0} \mapsto y_0 \quad , \quad \sigma_{y_0} \mapsto y_0 \quad , \quad \sigma_{z_0} \mapsto I_{2\cdot 2^m} \quad , \quad \sigma_{x_i} \mapsto x_i \quad , \quad \sigma_{y_i} \mapsto y_i \quad , \quad \sigma_{z_i} \mapsto z_i \ . \tag{4.129}$$

We will subsequently show that the substitution rule actually translates Nth order DD in the qubit setting to Nth order homogenisation in the bosonic setting. Key to this construction will be the identification of similarities between the multi-qubit Lie group and algebra with the CV Lie group and algebra that we studied in Section 4.7. Recall that we derived that the Pauli matrices σ_α and the symplectic matrices S_α satisfy similar commutation relations (4.121) and (4.113), respectively. Since in the qubit-setting, it is precisely the phases $\sigma_\beta^{-1}\sigma_\alpha\sigma_\beta = (-1)^{\langle\alpha,\beta\rangle}$ that are responsible for the vanishing terms in the Dyson expansion of the toggling frame evolution, there is justified hope that this translates to the bosonic setting. It is obvious that for fixed α and β the two expressions $\langle\alpha,\beta\rangle$ and $\langle\alpha,\beta'\rangle$ are usually equal since β and β' may only differ on the 0th entry. We will furthermore show that the distinction between β' and β is exactly what differentiates multi-qubit DD from CV homogenisation.

Let us first show how CV pulse sequences, which act on a system that is already decoupled from its environment, fall into the framework presented in Section 4.5 and state a sufficient condition for Nth order homogenisation within this framework. Since for homogenisation schemes in the sense of Definition 4.9, we assume that system and environment are decoupled, decoherence is introduced by the original Hamiltonian

$$H^{\mathrm{orig}}(t) \coloneqq \frac{1}{2} \sum_{j,k=1}^{2\cdot 2^m} A_{jk}(t) R_j R_k \ , \tag{4.130}$$

acting on the system only. Here, for every $t \in \mathbb{R}$, $A(t) \in \mathbb{R}^{2\cdot 2^m \times 2\cdot 2^m}$ is a symmetric matrix and R is the vector of system quadratures in a suitable order (choose the ordering (4.123)).

Lemma 4.30 *(Sufficient condition for bosonic homogenisation).* Let Δ be a finite ordered. On 2^m bosonic modes, consider a CV pulse sequence of the form

$$\big(\{t_\lambda\}_{\lambda\in\Lambda}, \{S_{\beta(\lambda)}\}_{\lambda\in\Lambda}\big) \ , \tag{4.131}$$

where the pulses $S_{\beta(\lambda)} \in \mathsf{Sp}(2\cdot 2^m, \mathbb{R})$ are defined by a function

$$\beta : \Lambda \to \tilde{\Gamma}$$

and where the matrices $S_\beta \in \mathsf{Sp}(2\cdot 2^m, \mathbb{R})$ as well as the set $\tilde{\Gamma}$ are given in Definition 4.24. Let the functions $F_{\alpha,\beta}$ and the scalars $\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}$ be defined as in Lemmas 4.12 and 4.13, respectively, and assume that the following holds:

$$\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}\Big(\{F_{\alpha_l,\beta}\}_{l=1}^k\Big) = 0 \quad \begin{cases} \text{for } k\in\mathbb{N},\ r_1,\ldots,r_k\in\mathbb{N}_0, \text{ and } \alpha_1,\ldots,\alpha_k \in (\mathbb{Z}_2^2)^{m+1} \\ \text{such that } k + r_1 + \ldots + r_k \leq N \text{ and} \\ \bigoplus_{l=1}^k \alpha_l \notin \{\big((0,0),\ldots,(0,0)\big), \big((1,1),(0,0),\ldots,(0,0)\big)\}. \end{cases} \tag{4.132}$$

Then the pulse sequence (4.131) satisfies Nth order homogenisation. □

The proof idea is the following: We will first show that the above pulse sequence falls into the framework from Section 4.5. In a second step we will apply Corollary 4.15 from that section to proof the homogenisation property.

Proof. By assumption, $S^{\mathrm{orig}}(t) \in \mathsf{Sp}(2\cdot 2^m + 2n_E, \mathbb{R})$ is generated by an element $X^{\mathrm{orig}}(t) \in \mathfrak{sp}(2\cdot 2^m + 2n_E, \mathbb{R})$ which is of the form

$$X^{\mathrm{orig}}(t) = \begin{pmatrix} X^{\mathrm{orig}}_{SS}(t) & X^{\mathrm{orig}}_{SE}(t) \\ X^{\mathrm{orig}}_{ES}(t) & X^{\mathrm{orig}}_{EE}(t) \end{pmatrix} \qquad \text{where } X^{\mathrm{orig}}_{SE}(t) = 0 = X^{\mathrm{orig}}_{ES}(t) \ . \tag{4.133}$$

Here X^{orig}_{SS} is the original generator that is associated with the Hamiltonian (4.130). Since the generator (4.133) is block-diagonal as well as the symplectic pulses (acting non-trivial on the system only), we restrict our attention to the system only throughout this proof and neglect the environment. In slight abuse of notation, we omit the index SS and write X^{orig} to denote X^{orig}_{SS}.

Let us now show that this original generator X^{orig} (recall, that this means the system part X^{orig}_{SS}) and the function β fall into the framework presented in Section 4.5. In the latter, we discuss decoupling and homogenisation in the setting of a general Lie group G and its associated Lie algebra $\mathfrak{g}$. Here, $\mathsf{G} = \mathsf{Sp}(2\cdot 2^m, \mathbb{R})$ and $\mathfrak{g} = \mathfrak{sp}(2\cdot 2^m, \mathbb{R})$. The pulses are chosen from the family of matrices $\{S_\beta\}_{\beta\in\tilde{\Gamma}}$ which is a subset of G by part (b) of Theorem 4.25. Furthermore, the family of matrices $\{S_\alpha\}_{\alpha\in\Gamma}$ forms a basis of $\mathfrak{g}$ by part (a) of Theorem 4.25. The adjoint action is given by

$$\mathsf{Ad}(S_\beta)S_\alpha := S_\beta^{-1} S_\alpha S_\beta = (-1)^{\langle\alpha,\beta\rangle} \quad \text{for all } \alpha\in\Gamma, \beta\in\tilde{\Gamma} \tag{4.134}$$

as proven in part (c) of Theorem 4.25, where

$$\langle\alpha,\beta\rangle = \Big(\sum_{j=0}^m a_j^T J_1 b_j\Big) \mod 2 \ .$$

Note that whether we include taking the modulo in the symplectic inner product does not make a difference for the phases in Eq. (4.134). By assumption X^{orig} has analytic time-dependence, i.e., using the basis $\{S_\alpha\}_{\alpha\in\Gamma}$ of $\mathfrak{sp}(2\cdot 2^m,\mathbb{R})$ from Theorem 4.25, we can write

$$X^{\mathrm{orig}}(t)=\sum_{\alpha\in\Gamma} b_\alpha(t)S_\alpha \quad \text{where} \quad b_\alpha(t)=\sum_{r=0}^{\infty} b_{\alpha,r}t^r \quad \text{where} \quad b_{\alpha,r}\in\mathbb{R}\ . \tag{4.135}$$

We note that here, the 'environment' $\mathcal{H}_E$ of the setup 4.11 is a technical quantity: it is one-dimensional – i.e., $\mathcal{H}_E=\mathbb{R}$ and $b_{\alpha,r}\in\mathbb{R}$ from Eq. (4.135) – and it should not be confused with the physical environment of n_E bosonic modes, which we omit in our considerations since we assume that system and environment evolve independently.

As a consequence, all assumptions 1-3 from the Setup 4.11 are satisfied, and we find that the toggling frame generator (more precisely, its system-only part) is given by

$$X^{\mathrm{tf}}(t)=\sum_{\alpha\in\Gamma} F_{\alpha,\beta}\left(t/T\right)S_\alpha B_\alpha(t)\ , \qquad \text{where} \qquad F_{\alpha,\beta}\left(t/T\right)=(-1)^{\sum_{j=1}^{K_t}\langle\alpha,\beta(\lambda_j)\rangle}$$

for $t\in[0,T]$ and for K_t from Eq. (4.24). This generator appears in the Dyson expansion of the toggling frame evolution

$$S^{\mathrm{tf}}(T)=\sum_{k=0}^{\infty}\sum_{\alpha_1,\ldots,\alpha_k\in\Gamma}\sum_{r_1,\ldots,r_k=0}^{\infty} T^{k+r_1+\ldots+r_k}\left(\prod_{l=1}^{k}S_{\alpha_l}b_{\alpha_l,r_l}\right)\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}\left(\{F_{\alpha_l,\beta(\lambda)}\}_{l=1}^{k}\right)\ .$$

Since all the assumptions of Setup 4.11 are satisfied, one can apply Corollary 4.15 for the index $\gamma:=\big((1,1),(0,0),\ldots,(0,0)\big)\in\Gamma$. It states that under the condition (4.132) there are scalars $c_1,c_2\in\mathbb{R}$ such that $\|S^{\mathrm{res}}(T)-c_1I_{2\cdot 2^m}-c_2S_\gamma\|=O(T^{N+1})$. Since $S_\gamma=-J_{2^m}$ by Eq. (4.112), this implies Nth order homogenisation of the considered pulse sequence. ■

Let us now relate multi-qubit DD schemes to bosonic homogenisation schemes.

Theorem 4.31 *(CV homogenisation from qubit DD schemes).* Let $m\in\mathbb{N}$ and $\beta:\Lambda\to(\mathbb{Z}_2^2)^{m+1}$. Assume that the pulse sequence

$$\left(\{t_\lambda\}_{\lambda\in\Lambda},\{\sigma_{\beta(\lambda)}\}_{\lambda\in\Lambda}\right) \tag{4.136}$$

on $m+1$ qubits – where σ_β is defined as in Eq. (4.55) – achieves Nth order dynamical decoupling in the sense of Theorem 4.14 from Section 4.5.

Then the symplectic pulse sequence

$$\left(\{t_\lambda\}_{\lambda\in\Lambda},\{S_{\beta'(\lambda)}\}_{\lambda\in\Lambda}\right) \tag{4.137}$$

on 2^m bosonic modes – where the function $\beta':\Lambda\to(\mathbb{Z}_2^2)^{m+1}$ is related to β by the substitution rule 4.29 – achieves Nth order CV homogenisation in the sense of Definition 4.9, i.e., a homogenised system evolution up to order N in T of an already decoupled system Hamiltonian with analytic time dependence. □

We note that the condition $\bigoplus_{l=1}^{k}\alpha_l\notin\left\{\big((0,0),\ldots,(0,0)\big),\big((1,1),(0,0),\ldots,(0,0)\big)\right\}$ is equivalent to

$$\prod_{l=1}^{k}S_{\alpha_l}\notin\{\pm I_{2\cdot 2^m},\pm J_{2^m}\}\ .$$

Proof. Let us first show that the given multi-qubit pulse sequence satisfies the assumptions of the setup in Section 4.5, i.e., the necessary properties are listed in Definition 4.11.

On the qubit level Nth order dynamical decoupling in the sense of Theorem 4.14 means that the function $\beta : \Lambda \to (\mathbb{Z}_2^2)^{m+1}$ satisfies

$$\begin{aligned} &\mathcal{F}^{(r_1,\dots,r_k)}_{(\alpha_1,\dots,\alpha_k)}\left(\{F_{\alpha_l,\beta(\lambda)}\}_{k=1}^{k}\right) = 0 \quad \text{for } k \in \mathbb{N}, r_1,\dots,r_k \in \mathbb{N}_0, \alpha_1,\dots,\alpha_k \in (\mathbb{Z}_2^2)^{m+1} \\ &\text{such that } k + \sum_{l=1}^{k} r_l \leq N \text{ and } \bigoplus_{l=1}^{k} \alpha_l \neq \big((0,0),\dots,(0,0)\big) \end{aligned} \tag{4.138}$$

Let us now consider the bosonic setting and the pulse sequence (4.137) obtained by the substitution rule $\sigma_\beta \mapsto S_{\beta'}$ from Definition 4.29, i.e., let a map $\beta : \Lambda \mapsto (\mathbb{Z}_2^2)^{m+1}$ be given. As already discussed, $\beta' \in \tilde{\Gamma}$ for all $\beta \in (\mathbb{Z}_2^2)^{m+1}$, i.e., for every $\lambda \in \Lambda$, we find $\beta'(\lambda) \in \tilde{\Gamma}$. Hence, as shown in Lemma 4.30, the bosonic pulse sequence (4.137) falls into the framework from Section 4.5. Furthermore, Lemma 4.30 gives a sufficient condition for Nth order homogenisation namely. We will hence prove that this Condition (4.132) is satisfied.

For the second step, let $N \in \mathbb{N}$ and let $k \in \mathbb{N},\ r_1,\dots,r_k \in \mathbb{N}_0,,\ \alpha_1,\dots,\alpha_k \in \Gamma$ be such that

$$k + \sum_{l=1}^{k} r_l \leq N \quad \text{and} \quad \bigoplus_{l=1}^{k} \alpha_l \notin \big\{\big((0,0),\dots,(0,0)\big), \big((1,1),(0,0),\dots,(0,0)\big)\big\} \tag{4.139}$$

are satisfied. For every $l = 1,\dots,k$, one may introduce the notation

$$\alpha_l = \big(a_0(l), a_1(l),\dots,a_m(l)\big) \quad , \quad \text{where } a_0(l) \coloneqq \big(\kappa(l),\mu(l)\big) \in \mathbb{Z}^2 \ ,$$

for the different components of the lth index α_l. Of special interest is its above-defined first component $\kappa(l)$ (we call the component with index 0 the first). We also define the new index $\widetilde{\alpha}_l$ by its relation to α_l, i.e., as

$$\widetilde{\alpha}_l \coloneqq \kappa(l)\big((1,1),(0,0),\dots,(0,0)\big) \oplus \alpha_l = \left[\left(\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} \oplus I_2 \oplus \cdots \oplus I_2\right)\alpha\right] \mod 2 \ . \tag{4.140}$$

This new index $\widetilde{\alpha}_l$ satisfies – using that $\oplus$ defines addition modulo two –

$$\bigoplus_{l=1}^{k} \alpha_l = \bigoplus_{l=1}^{k} \widetilde{\alpha}_l = \bigoplus_{l=1}^{k} \widetilde{\alpha}_l \oplus \big((\kappa,\kappa),(0,0),\dots,(0,0)\big) \tag{4.141}$$

where $\kappa \coloneqq \bigoplus_{l=1}^{k} \kappa(l)$. Inserting the assumption (4.139) on $\alpha_1,\dots,\alpha_k$ into Eq. (4.141) yields that in both cases, $\kappa = 0$ as well as $\kappa = 1$, we have

$$\bigoplus_{l=1}^{k} \widetilde{\alpha}_l \neq \big((0,0),\dots,(0,0)\big) \ . \tag{4.142}$$

The remainder of this proof is dedicated to showing that the scalar $\mathcal{F}^{(r_1,\dots,r_k)}_{(\alpha_1,\dots,\alpha_k)}\left(\{F_{\alpha_l,\beta'}\}_{k=1}^{k}\right)$ is equal to zero. In order to do so, let us compute the term

$$\langle \alpha, \beta'(\lambda)\rangle = \bigoplus_{j=0}^{m} a_j J_1 b'_j(\lambda) \quad , \qquad \text{for } \lambda \in \Lambda \ .$$

This term appears in the integrand of above scalar $\mathcal{F}$, more precisely as part of the exponent of the phase (-1) inside the functions $F_{\alpha,\beta'(\lambda)}$, defined in Eq. (4.49). Since we can always consider this quantity modulo 2 (as it appears in the exponent of a (-1) term), we compute

$$\begin{aligned}(a_0^T J_1 b_0') \mod 2 &= \left[a_0^T \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix} b_0\right] \mod 2 \\ &= \left[a_0^T \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}^T \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} b_0\right] \mod 2 \\ &= (\widetilde{a}_0^T J_1 b_0) \mod 2 \,.\end{aligned}$$

This implies that the first components of $\langle \alpha, \beta'(\lambda)\rangle$ and $\langle \widetilde{\alpha}, \beta(\lambda)\rangle$ agree (modulo 2). By definition, it is clear that the quantities $\beta'(\lambda) = (b_0'(\lambda), b_1(\lambda), \ldots, b_m(\lambda))$ from Eq. (4.128) and $\widetilde{\alpha} \coloneqq (\widetilde{a}_0, a_1, \ldots, a_m)$ from Eq. (4.140) can at most differ from $\beta(\lambda)$ and α, respectively, on their first component. As a consequence, we have proven that for every $\lambda \in \Lambda$

$$\langle \alpha, \beta'(\lambda)\rangle = a_0^T J_1 b_0'(\lambda) \oplus \sum_{j=1}^{m} a_j^T J_1 b_j(\lambda) = \langle \widetilde{\alpha}, \beta(\lambda)\rangle \quad \text{for all } \alpha \in (\mathbb{Z}_2^2)^{m+1} \,.$$

Recalling the definition of the functions F_α from Eq. (4.49), we can conclude that

$$F_{\alpha,\beta'(\lambda)} = F_{\widetilde{\alpha},\beta(\lambda)} \quad \text{for all } \alpha \in (\mathbb{Z}_2^2)^{m+1}, \lambda \in \Lambda \,. \tag{4.143}$$

Let us now exploit the assumption, that the pulse sequence (4.136) achieves Nth order dynamical decoupling in the sense of condition (4.138). We note that $k \in \mathbb{N}$ and $r_1, \ldots, r_k \in \mathbb{N}_0$ have been chosen such that $k + r_1 + \ldots + r_k \leq N$ by assumption (4.139) and that the indices $\widetilde{\alpha}_1, \ldots, \widetilde{\alpha}_k \in \Gamma \subset (\mathbb{Z}_2^2)^{m+1}$ are such that Eq. (4.142) is satisfied. Therefore, Nth order dynamical decoupling (4.138) of the multi-qubit sequence gives that

$$\mathcal{F}^{(r_1,\ldots,r_k)}_{(\widetilde{\alpha}_1,\ldots,\widetilde{\alpha}_k)}(\{F_{\widetilde{\alpha}_l,\beta(\lambda)}\}_{l=1}^k) = 0 \,.$$

Since Eq. (4.143) implies $F_{\widetilde{\alpha}_l,\beta(\lambda)} = F_{\alpha_l,\beta'(\lambda)}$ for every $l = 1, \ldots, k$, we find that

$$\mathcal{F}^{(r_1,\ldots,r_k)}_{(\alpha_1,\ldots,\alpha_k)}\left(\{F_{\alpha_l,\beta'(\lambda)}\}_{l=1}^k\right) = 0 \,.$$

In summary, we have shown that the symplectic pulse sequence (4.137) satisfies the Condition (4.132) and hence achieves Nth order homogenisation. ■

Theorem 4.31 shows that the substitution rule 4.29 translates an Nth order $m+1$-qubit DD scheme to an Nth order 2^m-mode bosonic homogenisation scheme. The pulse times are kept and the pulses are translated using the substitution rule 4.29. Note that the number of pulses may be even lower since all multi-qubit pulses of the form $(\sigma_z)_0$ are mapped to an identity, i.e., no pulse is applied at the corresponding time in the CV sequence (cf. Fig. 4.7 for an example).

4.8.3 Uhrig homogenisation schemes

Here, we present the central result of this chapter: the **novel homogenisation schemes**. The previous section showed how to translate a multi-qubit DD schemes of Nth order into a bosonic homogenisation scheme of the same order with the same (or a lower) number of pulses. Let us now apply this strategy from Definition 4.29 to the most efficient multi-qubit DD schemes, the NUDD schemes presented in Section 4.4.3.

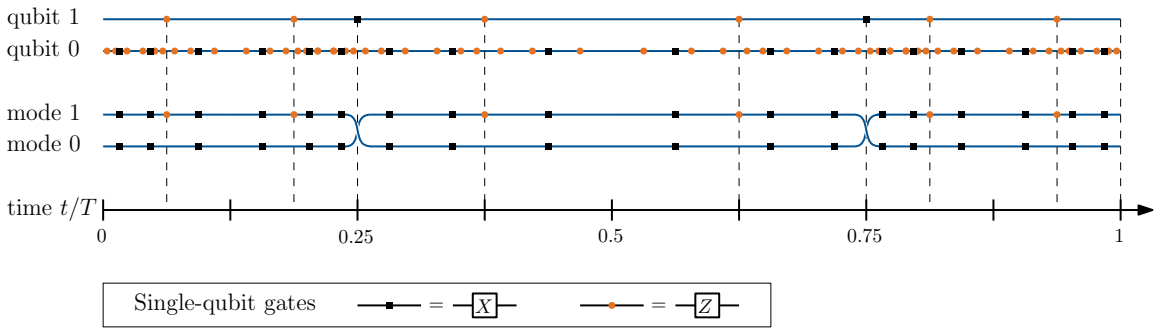


Figure 4.7: CV nested Uhrig homogenisation scheme for two modes and of order two. On top, the NUDD scheme is shown for two qubits and DD order $N = 2$ using $(N+1)^{2+2} - 1 = 80$ pulses each of which are either σ_x or σ_z Pauli gates on one of the two qubits. By applying the substitution rule from Definition 4.29, one obtains the deduced bosonic homogenisation scheme (on the bottom) of the same order $N = 2$ using $(N+1)^{2+1} - 1 = 26$ pulses. The blue lines symbolise free evolution of the system under the decoherence Hamiltonian. This evolution is interleaved with pulses, represented as orange or black icons. For the definition of these pulses, we refer to Fig. 4.5.

Theorem 4.32 *(CV nested Uhrig homogenisation schemes).* Let $m \in \mathbb{N}$, $N \in \mathbb{N}$ and consider the Nth order NUDD scheme for $m+1$ qubits ([86] and cf. Section 4.4.3). Then:

(i) The image of this pulse sequence under the substitution rule from Definition 4.29 is an Nth order homogenisation scheme for 2^m bosonic modes (in the sense of Definition 4.9).

(ii) The number of pulses in the CV scheme is $(N+1)^{2m+1}-1$ if N is even and $(N+1)^{2m+1}$ if N is odd.

(iii) All bosonic pulses are unitaries from Definition 4.27, i.e., they are passive Gaussian unitaries which are products of single-mode phase space rotations and two-mode SWAPs.

□

Proof. (i) The proof is a direct application of Theorem 4.31. In order to this latter, it is necessary to show that the Nth order NUDD scheme for $m+1$ qubits actually satisfies the assumptions of Theorem 4.31. More precisely, it is sufficient to show that the $m+1$-qubit NUDD scheme satisfies the assumptions of Theorem 4.14 from the general framework in Section 4.5. In Section 4.5.3 we showed that the NUDD scheme falls into the framework of DD for general Lie groups (cf. Definition 4.11). A proof of the other assumption of the decoupling criterion Theorem 4.14 is given by Liang and Imambekov [86] (cf. Lemma 4.16) As a consequence, one may apply Theorem 4.31.

(ii) The pulses of the NUDD scheme for N even are mapped to the bosonic pulses

$$(\sigma_x)_0 \mapsto U_{y_0} \quad , \quad (\sigma_x)_i \mapsto U_{x_i} \quad , \quad (\sigma_z)_0 \mapsto U_{I_{2\cdot 2^m}} \quad , \quad (\sigma_z)_i \mapsto U_{z_i}$$

for $i = 1, \ldots, m$ (recalling Eq. (4.129)). Since the Nth order DD pulse sequence consists of $(N+1)^{2m+2}-1$ pulses, among those N times $(N+1)^{2m+1}$ $(\sigma_z)_0$-pulses at the innermost

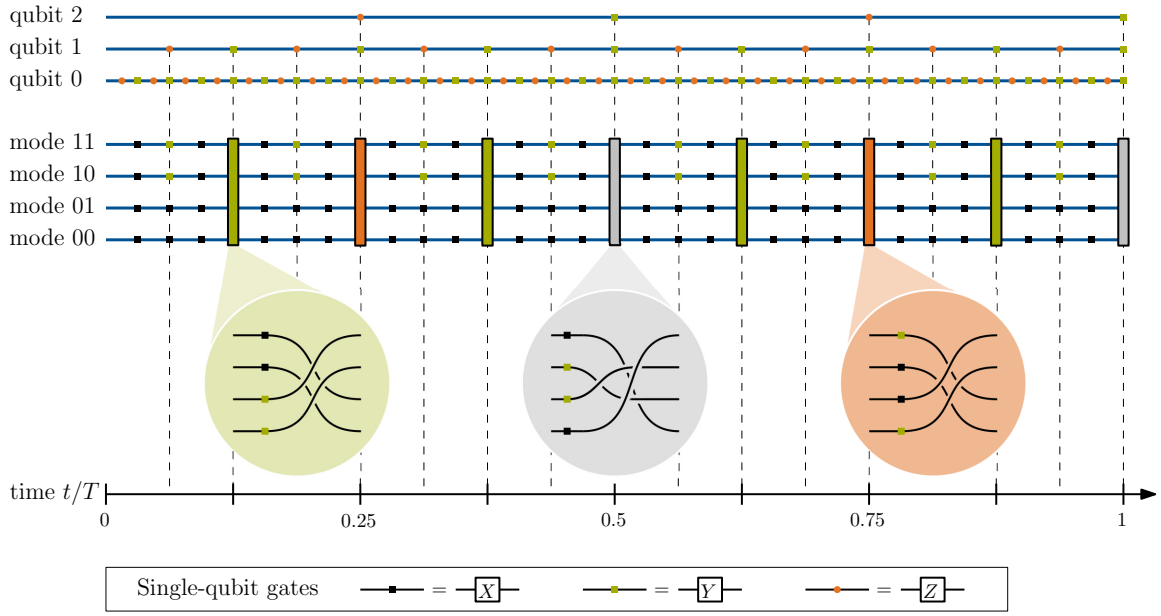


Figure 4.8: CV nested Uhrig homogenisation scheme (bottom) for four modes and order $N = 1$ and the corresponding first order three-qubit NUDD scheme (top). In both cases, the evolution under the (decoherence) Hamiltonian is interleaved by control pulses. The control pulses of the NUDD scheme (top) – which are products of single-qubit Paulis σ_x and σ_y – are translated to the homogenisation pulses by the substitution rule from Definition 4.29. Note that by definition of the NUDD sequence for odd decoupling orders, the pulses are more complicated than for even N. The single-mode pulses are described in Fig. 4.5.

level, the deduced bosonic scheme has $(N+1)^{2m+2}-1-N(N+1)^{2m+1} = (N+1)^{2m+1}-1$ non-trivial pulses. For N odd the pulses are given by

$$\prod_{i=0}^{k}(\sigma_y)_i \mapsto U_{y_0}\prod_{j=1}^{k}U_{y_j} \quad , \quad \prod_{i=0}^{k-1}(\sigma_y)_i(\sigma_x)_k \mapsto U_{y_0}\prod_{j=1}^{k-1}U_{y_j}U_{x_k}$$

and since again the inne the number of pulses from reduces from $(N+1)^{2m+2}$ in the multi-qubit setting to $(N+1)^{2m+1}$ bosonic pulses.

(iii) By Lemma 4.28 (part (b)), these pulses are passive Gaussian unitaries which are products of single-mode phase rotations U_{rot} and SWAP gates, equivalently products of single-mode phase rotations and transmittivity zero beam splitter gates.

■

Figs. 4.7, 4.8 and 4.9 show three examples of bosonic homogenisation schemes constructed from NUDD schemes, the first for the parameters $N = 2$ and $m = 1$, the second for $N = 1$ and $m = 2$ and the third for $N = 3$ and $m = 1$.

In order to achieve decoherence suppression and homogenisation simultaneously, let us combine the Uhrig decoherence suppression scheme from Theorem 4.20 and the Uhrig homogenisation scheme from Theorem 4.32. More precisely, for n_S bosonic system modes, we concatenate the former (of Nth order) at the inner level with the latter (also of Nth order) at the outer level: in between the application of any neighbouring pair of homogenisation pulses, a whole Nth order Uhrig decoherence suppression sequence is applied, i.e., with N (or $N+1$)

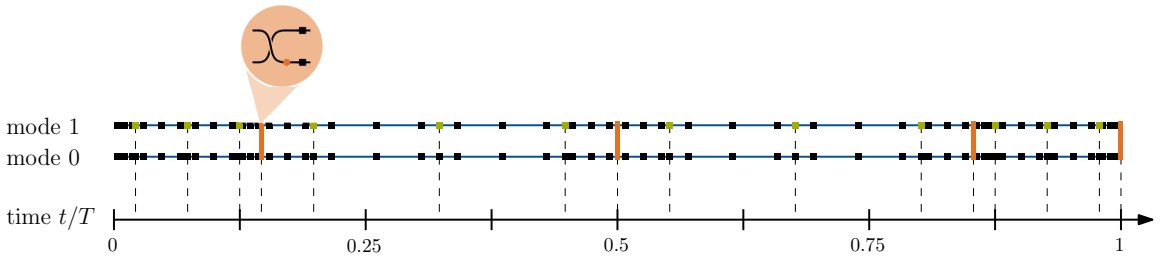


Figure 4.9: CV nested Uhrig homogenisation scheme for two modes and homogenisation order three. The evolution under the decoherence Hamiltonian (4.130) (depicted by blue lines) is interleaved with instantaneous application of control pulses (represented as green or black icons), the (single-mode) pulses are described in more detail in Fig. 4.5. The shown two-mode gate (orange) is given by the subsequent application of the basic single- and two-mode pulses from Fig. 4.5) as indicated by the inset above the circuit.

pulses at non-equidistant Uhrig times. The resulting schemes only uses

$$(N+1)(N+1)^{2\log_2(n_S)+1} = (N+1)^{2\log_2(n_S)+2} \approx O(N^{\log_2(n_S)})$$

pulses to achieve $\|S^{\text{res}}(T) - S_0 \oplus S_E\| = O(T^{N+1})$ for some environment term S_E and the homogenised system evolution S_0 at the symplectic level. At the unitary level, the latter corresponds to the unitary

$$U_{S_0} = e^{-i\omega/2 \sum_{i=1}^{n_S} Q_i^2 + P_i^2}$$

on the n_S system modes for some frequency $\omega \in \mathbb{R}$.

4.9 Further considerations

In this section, we present two further results on the constructed pulse sequences. In Section 4.9.1, we analyse the convergence of the considered Dyson series in terms of the time T after which one considers decoherence suppression. Furthermore, we investigate how the constructed pulse sequences perform when the decoherence Hamiltonian contains additional linear terms in Section 4.154.

4.9.1 Sufficient rates for decoherence suppression

So far, we did not specify how to exactly quantify a 'sufficiently small' time T. Since our analysis relies on the Dyson series of the toggling frame $S^{\text{tf}}(T)$, it is necessary that T is small enough such that this Dyson series converges.

We consider the toggling frame generator from Eq. (4.27) and investigate whether the generated Dyson series converges. Recalling the convergence condition from Eq. (3.19) (Dyson series from Section 3.3.1), this is the case if

$$\max_{t\in[0,T]} \|X^{\text{tf}}(t)\| = \max_{t\in[0,T]} \left\| \begin{pmatrix} X_{SS}(t) & \sigma(t)X_{SE}(t) \\ \sigma(t)X_{ES}(t) & X_{EE}(t) \end{pmatrix} \right\| < \infty \,. \tag{4.144}$$

We call $1/T$ the decoupling rate and T the decoupling cycle time. We note that the name cycle time originates from periodic DD where is specifies the time after which a decoupling

cycle has to be periodically repeated; here, it describes the time after which the whole pulse sequence has been applied.

Let us investigate the convergence criterion of Eq. (4.144) in more detail for the specific case of a time-independent symplectic generator

$$X^{\mathrm{orig}} = \begin{pmatrix} X_{SS} & X_{SE} \\ X_{ES} & X_{EE} \end{pmatrix} \in \mathfrak{sp}(2n_S + 2n_E, \mathbb{R}) \ .$$

We will give a bound on the necessary rate $1/T$ in terms of the energy scales set by the the system part $\|X_{SS}^{\mathrm{orig}}\|$, the system-environment interactions $\|X_{SE}^{\mathrm{orig}}\|$ and the environment part. Our analysis is conducted similarly to the one presented in the article [163] in which a system of a single-qubit with pure dephasing decoherence is considered. But note that we only examine and mimic the simplest form of the bounds presented in the above article. The more refined bounds from [163] seem much more difficult to translate to the bosonic setting.

Theorem 4.33 *(Sufficient rates for decoherence suppression).* For $n_S, n_E \in \mathbb{N}$, let

$$X^{\mathrm{orig}} = \begin{pmatrix} X_{SS} & X_{SE} \\ X_{ES} & X_{EE} \end{pmatrix} \in \mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$$

be given as well as the pulse sequence and consider the resulting symplectic evolution $S^{\mathrm{res}}(T)$ after application of a pulse sequence of U pulses (cf. Definition 4.18) specified by a function $\sigma : [0,1] \to \{-1,1\}$ which satisfies Eq. (4.72). If

$$\|X_{SS}\| + \|X_{SE}\| + \|X_{EE}\| \leq \frac{1}{T} \ , \tag{4.145}$$

then there are two matrices $S_{SS} \in \mathsf{Sp}(2n_S, \mathbb{R})$ and $S_{EE} \in \mathsf{Sp}(2n_E, \mathbb{R})$ such that

$$\|S^{\mathrm{res}}(T) - S_{SS} \oplus S_{EE}\| \leq \frac{e\sqrt{2}\left(\|X_{SS}\| + \|X_{SE}\| + \|X_{EE}\|\right)^{N+1}}{(N+1)!} T^{N+1} \ .$$

□

Proof. First recall that due to $X^{\mathrm{orig}} \in \mathfrak{sp}(2n_S + 2n_E, \mathbb{R})$ and with J from Eq. (4.21), we have that $\|X_{SE}\| = \|X_{ES}\|$ in any p-norm.

We will first show that

$$\|(S^{\mathrm{tf}}(T))_{SE}\| \leq \sum_{k=N+1}^{\infty} T^k \, \frac{\left(\|X_{SS}\| + \|X_{SE}\| + \|X_{EE}\|\right)^k}{k!} \ . \tag{4.146}$$

The toggling frame evolution $S^{\mathrm{tf}}(T)$ can be expanded in a Dyson series which is absolutely convergent if condition (4.144). This condition is equivalent to $\|X^{\mathrm{orig}}\| < \infty$ since

$$\|X^{\mathrm{tf}}(t)\| \coloneqq \left\| \begin{pmatrix} X_{SS} & \sigma(t) X_{SE} \\ \sigma(t) X_{ES} & X_{EE} \end{pmatrix} \right\| = \|X^{\mathrm{orig}}\| < \infty \ .$$

Recall the computation of the Dyson series from Eq. (4.50). Then the upper right block of $S^{\mathrm{tf}}(T)$ is given by the Dyson series

$$(S^{\mathrm{tf}}(T))_{SE} = \sum_{k=0}^{\infty} T^k \sum_{(A,B) \in \nu_k(S,E)} \left(\prod_{m=1}^{k} X_{A_m B_m} \right) \mathcal{F}^{(0,\ldots,0)}_{(\alpha_1,\ldots,\alpha_k)} \left(\{\sigma^{\alpha_m}\}_{m=1}^{k} \right) , \tag{4.147}$$

where the sequences from Eq. (4.66). In this calculation, we used the fact that if X^{orig} is not time-dependent, then $X_{AB,r} = 0$ in the analytic expansion (4.45) for all $r \geq 1$. By the assumption that σ satisfies Eq. (4.72), i.e., that

$$\mathcal{F}^{(0,\ldots,0)}_{(\alpha_1,\ldots,\alpha_k)}\left(\{\sigma^{\alpha_m}\}_{m=1}^k\right) = 0 \quad \begin{cases} \text{for all } k \in \mathbb{N},\, \alpha_1,\ldots,\alpha_k \in \{0,1\} \\ \text{such that } k \leq N \text{ and } \bigoplus_{m=1}^k \alpha_m = 1, \end{cases} \tag{4.148}$$

the first N terms in (4.147) vanish. For higher order terms with $k \geq N+1$, let us bound the norms of the product in (4.147). The sum

$$\sum_{(A,B)\in\nu_k(S,E)} \left\| \left(\prod_{m=1}^k X_{A_m B_m} \right) \right\| \leq \sum_{(A,B)\in\nu_k(S,E)} \prod_{m=1}^k \|X_{A_m B_m}\| \tag{4.149}$$

can be seen as the sum over certain paths of a binary tree of order s with vertices S and E. For a specific sequence $(A,B) \in \nu_k(S,E)$, the term $\prod_{m=1}^k \|X_{A_m B_m}\|$ is associated with one path from top to bottom with edges

$$((A_1,B_1),(B_1,B_2),\ldots(B_{k-1},B_k)) \ .$$

The summation over sequences $(A,B) \in \nu_k(S,E)$ assures that $A_1 = S$ and $B_k = S$, restricting the summation over graph paths to exactly those which end at a vertex E. Fig. 4.10 shows three examples of this identification. Very roughly estimated and using $\|X^{\mathrm{orig}}_{SE}\| = \|X^{\mathrm{orig}}_{ES}\|$, we thereby obtain

$$\begin{aligned} \sum_{(A,B)\in\nu_k(S,E)} \left\| \left(\prod_{m=1}^k X_{A_m B_m} \right) \right\| &\leq \sum_{\substack{m_1,m_2,m_3=1 \\ m_1+m_2+m_3=k \\ m_2 \text{ odd}}}^k k! \frac{\|X_{SS}\|^{m_1}\|X_{SE}\|^{m_3}\|X_{EE}\|^{m_3}}{m_1!m_2!m_3!} \\ &\leq (\|X_{SS}\| + \|X_{SE}\| + \|X_{EE}\|)^k \ . \end{aligned} \tag{4.150}$$

For $k \geq N+1$, another term in (4.147) gives

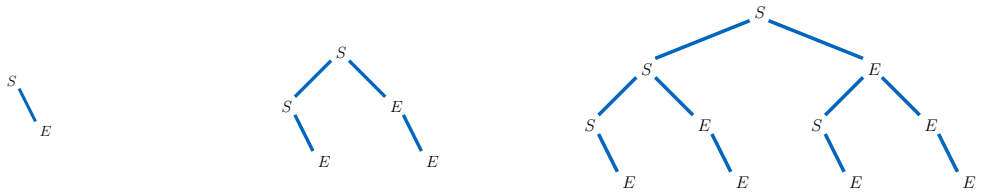
$$\begin{aligned} \left| \mathcal{F}^{(0,\ldots,0)}_{(\alpha_1,\ldots,\alpha_k)}\left(\{\sigma^{\alpha_m}\}_{m=1}^k\right) \right| &\leq \int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} |\sigma(\tau_1)^{\alpha_1}| \cdots |\sigma(\tau_k)^{\alpha_k}| d\tau_k \ldots d\tau_1 \\ &\leq \int_0^1 \int_0^{\tau_1} \cdots \int_0^{\tau_{k-1}} d\tau_k \ldots d\tau_2 d\tau_1 \leq \frac{1}{k!} \end{aligned} \tag{4.151}$$

using that $\sigma(\tau) \in [-1,1]$ for all $\tau \in [0,1]$. Inserting both estimates (4.150) and (4.151) into the Dyson expansion of $S^{\mathrm{tf}}(T)$ as well as using the fact that the first N terms vanish by (4.148), one can upper bound both off-diagonal parts of $S^{\mathrm{tf}}(T)$ by the same quantity, more precisely $\|(S^{\mathrm{tf}}(T))_{SE}\|$ and $\|(S^{\mathrm{tf}}(T))_{ES}\|$ are upper bounded by the right hand side of (4.146). Let us now use the assumption (4.145) which implies that

$$\|(S^{\mathrm{tf}}(T))_{SE}\| \leq (\|X_{SS}\| + \|X_{SE}\| + \|X_{EE}\|)^{N+1} T^{N+1} \sum_{k=N+1}^{\infty} \frac{1}{k!} \ . \tag{4.152}$$

The sum over k on the right hand side corresponds to th Nth order remainder term of the Taylor expansion of the exponential function around 0

$$e^1 = \sum_{k=0}^{N} \frac{1^k}{k!} + \sum_{k=N+1}^{\infty} \frac{1}{k!} \ .$$



(a) $k = 1$ (b) $k = 2$ (c) $k = 3$

Figure 4.10: Binary trees to compute the quantity on the right hand side of (4.149) in the following sense: A path with nodes $A_1, A_2, \ldots, A_{k+1} \in \{S, E\}^{k+1}$ gives a summand $\|X_{A_1A_2}\|\|X_{A_2A_3}\| \cdots \|X_{A_kA_{k+1}}\|$ such that the quantity on the right hand side of (4.149) is the sum of all these paths. Here the condition $(A, B) \in \nu_k(S, E)$ implies that all trees have root node S and outer nodes E. For $k = 1$ the right hand side of (4.149) is given by $\|X_{SE}\|$, and for $k = 2$ and $k = 3$, it is equal to $\|X_{SS}\|\|X_{SE}\| + \|X_{SE}\|\|X_{EE}\|$ and $\|X_{SS}\|^2\|X_{SE}\| + \|X_{SS}\|\|X_{SE}\|\|X_{EE}\| + \|X_{SE}\|^3 + \|X_{SE}\|\|X_{EE}\|^2$, respectively, where we used that $\|X_{SE}\| = \|X_{ES}\|$.

Let us use the Lagrange form of the remainder to estimate

$$\left| \sum_{k=N+1}^{\infty} \frac{1}{k!} \right| \leq \frac{e}{(N+1)!}.$$

As a consequence, Eq. (4.152) becomes

$$\|(S^{\mathrm{tf}}(T))_{SE}\| \leq (\|X_{SS}\| + \|X_{SE}\| + \|X_{EE}\|)^{N+1} T^{N+1} \frac{e}{(N+1)!} \,. \tag{4.153}$$

Note that the right hand side of (4.153) gives an upper bound on the two norms $\|(S^{\mathrm{tf}}(T))_{SE}\|$ and $\|(S^{\mathrm{tf}}(T))_{ES}\|$. Since, as usually $S^{\mathrm{tf}}(T) = S^{\mathrm{res}}(T)$, there exist matrices $S_{SS} \in \mathsf{Sp}(2n_S, \mathbb{R})$ and $S_{EE} \in \mathsf{Sp}(2n_E, \mathbb{R})$ such that

$$\|S^{\mathrm{res}}(T) - S_{SS} \oplus S_{EE}\|^2 \leq \|(S^{\mathrm{tf}}(T))_{SE}\|^2 + \|(S^{\mathrm{tf}}(T))_{ES}\|^2 = 2\|(S^{\mathrm{tf}}(T))_{SE}\|^2 \,,$$

which gives in combination with the upper bounds on $\|(S^{\mathrm{tf}}(T))_{SE}\|$ and $\|(S^{\mathrm{tf}}(T))_{ES}\|$ that

$$\|S^{\mathrm{res}}(T) - S_{SS} \oplus S_{EE}\| \leq \frac{e\sqrt{2}(\|X_{SS}\| + \|X_{SE}\| + \|X_{EE}\|)^{N+1}}{(N+1)!} T^{N+1} \,.$$

■

We note that this is a very simple bound on the necessary pulse rate. More refined bounds will require a more detailed analysis. In the above proof, the estimation which we made in the step from the first to second line in the calculation (4.150) is the roughest (in particular, it is not tight). If one wants to improve these bounds, a natural first step would be to improve this estimation.

4.9.2 Linear terms in the Hamiltonian

We note that the Hamiltonian $H^{\mathrm{orig}}(t)$ from Eq. (4.19) is not of the most general form. Although such quadratic terms in the mode operators are ubiquitous, other powers may appear.

A first and simple generalisation is to consider the presence of additional linear terms in the mode operators. In this section, we consider the effect of the constructed pulse sequences in the presence of such additional terms in the Hamiltonian which are linear and show that the decoupling and homogenisation properties of pulse sequences are not altered. More precisely, let the original Hamiltonian be of the form

$$H^{\text{orig}}(t) = \frac{1}{2}\sum_{j,k=1}^{2n} A_{jk}(t)R_jR_k + \sum_{j=1}^{2n} b_j(t)R_j \ , \tag{4.154}$$

where $A : [0,T] \mapsto \mathbb{R}^{2n\times 2n}$ is a symmetric matrix and $b : [0,T] \to \mathbb{R}^{2n}$ is an additional time-dependent vector. Instead of the quadratic Hamiltonians of the form of (4.19) considered so far, the Hamiltonian here includes additional time-dependent linear terms in the mode operators. The latter characterise displacements in the phase space (cf. Definition 2.25), e.g., they describe Gaussian displacement noise of the form of Definition 3.3. We will show that the presence of these linear terms has no influence on the decoupling and homogenisation properties that a pulse sequence has on the quadratic terms, i.e., a Hamiltonian of the form (4.19) has the same decoupling and homogenisation sequences as one of the form (4.154).

As described in Section 2.2.4, such a Hamiltonian generates a Gaussian unitary $U^{\text{orig}}(t)$ which is fully characterised by its effect on the first and second moments of a Gaussian state. Since all pulses appearing in $U^{\text{control}}(t)$ are likewise Gaussian unitaries, the resulting evolution $U^{\text{res}}(t)$ – generated by $H^{\text{orig}}(t) + H^{\text{control}}(t)$ – is still a Gaussian unitary. Let the initial Gaussian state $\rho \in \mathcal{D}(L^2(\mathbb{R}^n))$ on the $n = n_S + n_E$ system and environment modes have associated displacement vector $\overline{R}(\rho)$ and covariance matrix moments $V(\rho)$ from Definition 2.26. The resulting state

$$\rho^{\text{res}}(t) \coloneqq U^{\text{res}}(t)\rho\left(U^{\text{res}}(t)\right)^\dagger \ ,$$

after evolution under $U^{\text{res}}(t)$ is still a Gaussian state which is characterised by new first and second moments

$$\overline{R}^{\text{res}}(t) \coloneqq \overline{R}(\rho^{\text{res}}(T)) = S^{\text{res}}(T)\overline{R}(\rho) + d(T) \ , \tag{4.155}$$

$$V^{\text{res}}(t) \coloneqq V(\rho^{\text{res}}(T)) = S^{\text{res}}(T)V(\rho)(S^{\text{res}}(T))^T \ , \tag{4.156}$$

respectively, for some $d \in \mathbb{R}^{2n}$, cf. Eqs. (2.28)-(2.29) of Section 2.2.3.

Lemma 4.34 *(No dependence on linear terms).* Let a Hamiltonian $H^{\text{orig}}(t)$ be of the form Eq. (4.154) as well as a Gaussian state ρ be given and define $V^{\text{res}}(t)$ by Eq. (4.156). Then

$$\frac{d}{dt}V^{\text{res}}(t) = -JA^{\text{res}}(t)V(\rho) + V(\rho)A^{\text{res}}(t)J \ . \tag{4.157}$$

In particular, this equation shows no dependence on $b(t)$. □

Proof. A first observation shows that since every control pulse is a Gaussian unitary of the form $e^{-iH^{\text{control}}(t)}$, the control Hamiltonian is

$$H^{\text{control}}(t) = \frac{1}{2}\sum_{j,k=1}^{2n} A_{jk}^{\text{control}}(t)R_jR_k$$

for a time-dependent symmetric matrix $A^{\text{control}} : [0,T] \to \mathbb{R}^{2n\times 2n}$ that is characterised by the specific pulse times and operators and satisfies $A_{jk}^{\text{control}} = 0$ for $j = 2n_S + 1, \ldots, 2n$

or $k = 2n_S + 1, \ldots, 2n$ since the pulses only act non-trivial on the system. As a consequence, the resulting Hamiltonian which generates the resulting evolution is of the form

$$H^{\mathrm{res}}(t) \coloneqq H^{\mathrm{orig}}(t) + H^{\mathrm{control}}(t) = \frac{1}{2}\sum_{j,k=1}^{2n} A_{jk}^{\mathrm{res}}(t) R_j R_k + \sum_{j=1}^{2n} b_j(t) R_j$$

where $A^{\mathrm{res}}(t) \coloneqq A(t) + A^{\mathrm{control}}(t)$ for $t \in [0,T]$ is a time-dependent symmetric matrix. Note that the time-dependent vector in front of the linear term does not depend on the control Hamiltonian (i.e., the pulse sequence).

Consider the covariance matrix of a Gaussian state ρ defined in Eq. (2.24). Direct computation shows that its matrix elements can be expressed in the form

$$V_{jk}(\rho) = \mathrm{tr}\left(\{R_j, R_k\}\rho\right) - 2\,\mathrm{tr}(R_j\rho)\,\mathrm{tr}(R_k\rho) \quad \text{for } j,k = 1,\ldots,2n\ . \tag{4.158}$$

Let $\rho^{\mathrm{res}}(t)$ evolve under the resulting evolution such that it satisfies

$$\frac{d}{dt}\rho^{\mathrm{res}}(t) = i[H^{\mathrm{res}}(t), \rho^{\mathrm{res}}(t)]\ .$$

As a consequence, the derivative of its covariance matrix $V^{\mathrm{res}}(t)$ (using (4.158)) is given by

$$\begin{aligned}\frac{d}{dt}V_{jk}^{\mathrm{res}}(t) =& i\,\mathrm{tr}\left([\{R_j,R_k\}, H^{\mathrm{res}}(t)]\rho^{\mathrm{res}}(t)\right) + 2i\,\mathrm{tr}\left([R_j, H^{\mathrm{res}}(t)]\rho^{\mathrm{res}}(t)\right)\mathrm{tr}\left(R_k\rho^{\mathrm{res}}(t)\right)\\ &- 2i\,\mathrm{tr}\left(R_j\rho^{\mathrm{res}}(t)\right)\mathrm{tr}\left(\left[R_k, H^{\mathrm{res}}(t)\right]\rho^{\mathrm{res}}(t)\right)\ .\end{aligned} \tag{4.159}$$

To compute the result of the individual terms we use that the quadrature operators satisfy the commutation relations $[R_j, R_k] = iJ_{jk}$ which gives

$$\begin{aligned}[R_j, H^{\mathrm{res}}(t)] &= i\sum_{k,l=1}^{2n} J_{jk}A_{kl}^{\mathrm{res}}(t)R_l + i\sum_{k=1}^{2n} J_{jk}b_k(t)\ ,\\ [\{R_j,R_k\}, H^{\mathrm{res}}(t)] &= -i\sum_{l,m=1}^{2n} A_{lm}^{\mathrm{res}}(t)\left(\{R_l,R_j\}J_{mk} + \{R_l,R_k\}J_{mj}\right)\\ &\quad - 2i\sum_{l=1}^{2n} b_l(t)(R_jJ_{lk} + R_kJ_{lj})\end{aligned}$$

for $j,k = 1,\ldots,2n$. As a consequence, the first term in Eq. (4.159) is equal to

$$\begin{aligned}i\,\mathrm{tr}\left(\left[\{R_j,R_k\}, H^{\mathrm{res}}(t)\right]\rho^{\mathrm{res}}(t)\right) =& 2\sum_{l=1}^{2n} b_l(t)\left(J_{lj}\overline{R}_k^{\mathrm{res}}(t) + J_{lk}\overline{R}_j^{\mathrm{res}}(t)\right) +\\ &+ \sum_{l,m=1}^{2n} A_{lm}^{\mathrm{res}}(t)\left(V_{mk}^{\mathrm{res}}(t)J_{lj} + 2\overline{R}_m^{\mathrm{res}}(t)\overline{R}_k^{\mathrm{res}}(t)J_{lj}\right)\\ &+ \sum_{l,m=1}^{2n} A_{lm}^{\mathrm{res}}(t)\left(V_{mj}^{\mathrm{res}}(t)J_{lk} + 2\overline{R}_m^{\mathrm{res}}(t)\overline{R}_j^{\mathrm{res}}(t)J_{lk}\right)\end{aligned} \tag{4.160}$$

where $\overline{R}^{\mathrm{res}}(t)$ is defined in Eq. (4.155). For the second term of (4.159) let us compute

$$i\,\mathrm{tr}\left(\left[R_j, H^{\mathrm{res}}(t)\right]\rho^{\mathrm{res}}(t)\right) = \sum_{l,m=1}^{2n} A_{lm}^{\mathrm{res}}(t)J_{lj}\overline{R}_m^{\mathrm{res}}(t) + \sum_{l=1}^{2n} b_l(t)J_{lj}\ . \tag{4.161}$$

Note that the second term of Eq. (4.155) is equal to the third term with the roles of j and k interchanged. Setting everything together, i.e., inserting Eqs. (4.160) and (4.161) into Eq. (4.159), and using the symmetry of $A^{\mathrm{res}}(t)$ and $V^{\mathrm{res}}(t)$ as well as the antisymmetry of J one obtains that

$$\begin{aligned}\frac{d}{dt}V^{\mathrm{res}}_{jk}(t) &= \sum_{l,m=1}^{2n} A^{\mathrm{res}}_{lm}(t)\big(V_{mk}(t)J_{lj} + V_{mj}(t)J_{lk}\big) \\ &= \sum_{l,m=1}^{2n} \big(- J_{jl}A^{\mathrm{res}}_{lm}(t)V_{mk}(t) + V_{jm}A^{\mathrm{res}}_{ml}(t)J_{lk}\big)\ .\end{aligned}$$

This is equal to the desired Eq. (4.157). ■

Lemma 4.34 implies the following for the decoherence suppression and homogenisation properties of the pulse sequences considered throughout this chapter in the presence of linear decoherence terms. First, the linear decoherence terms cannot be reduced by the considered bosonic pulse sequences. Moreover, since the covariance matrix of the resulting state does not depend on the linear terms, these terms do not enter our analysis at all, i.e., the analysed matrix $S^{\mathrm{res}}(T)$ (the symplectic matrix associated with the resulting unitary $U^{\mathrm{res}}(T)$) has the same structure as for a quadratic decoherence Hamiltonian. As a consequence, pulse sequences that achieve Nth order decoherence suppression or homogenization for a quadratic original Hamiltonian in the sense of Definitions 4.8 and 4.9, respectively, will achieve the same in the presence of linear terms.

5 Asymmetric surface-Gottesman-Kitaev-Preskill code

This chapter includes results which have been published in the article *"Enhanced noise resilience of the surface-Gottesman-Kitaev-Preskill code via designed bias"* [76] by the author of this thesis together with her co-authors Lisa Hänggli and Robert König.

More precisely, this concerns the results in Sections 5.2.4, 5.4.2, 5.4.4, 5.5, 5.6 and central parts of 5.2.5. All authors contributed equally to the publication [76]. The central ideas were developed in joint discussions between the three co-authors. Margret Heinze computed the threshold values with the simulated data, cf. Sections 5.5.2 and Lisa Hänggli conducted the cutoff analysis, cf. Section 5.5.4. The central computations and proofs, i.e., those of the logical GKP error probabilities, were conducted in equal shares and jointly by Margret Heinze and Lisa Hänggli. The first draft for the simulation code was programmed by Robert König, the remainder was equally distributed among the three co-authors. Note that the results in Section 5.2.1, from Theorem 5.9 and the comparison to the biased noise model considered by Tuckett et al. [158] in Section 5.4.3 are new and have not been published before.

The idea

This chapter treats qubit-into-CV quantum error correcting codes and questions on their noise tolerance. The central research problem is: can one use an engineered noise bias to enhance the noise tolerance of a concatenated DV-into-CV encoding? More precisely, we consider the concatenation of the surface code and the Gottesman-Kitaev-Preskill (GKP) code [71] – two codes which are promising DV-into-DV and DV-into-CV encodings, respectively – and investigate whether an additional encoding of the GKP code (producing the noise bias) can augment the error thresholds of the concatenated code. The basis for this question is laid by two observations: first, applying a single-mode squeezing unitary to each mode of the GKP code effectively transforms the logical GKP noise from symmetric noise (independent equally likely X and Z-noise) to biased noise. On the other hand, asymmetry in the noise of the surface code enhances its noise resilience compared to symmetric (i.i.d.) noise [158, 160]. Here, we combine these two ideas: We introduce asymmetry in the GKP code by an adapted encoding (squeezing the GKP lattice in phase space) and numerically investigate whether this enhances the noise resilience of the concatenated surface-GKP code. Note that the answer is not clear a priori since there are limiting results on Gaussian error correction suggesting that Gaussian encoding cannot enhance the noise resilience of CV codes towards Gaussian errors. However, we find that the asymmetric surface-GKP codes achieve higher noise thresholds compared to the standard surface-GKP encoding, when assuming the idealised setup of error-free syndrome measurements.

Organisation of this chapter

- Section 5.1 describes the Gottesman-Kitaev-Preskill (GKP) codes [71] with a focus on symmetric (i.e., square lattice) GKP codes on a single bosonic mode.

- Section 5.2 introduces asymmetric (i.e., rectangular lattice) GKP codes and discusses symmetric and asymmetric physical displacement noise as well as the logical error distribution on the GKP-qubit. This section furthermore proves a key theorem that motivates the main result of this chapter: symmetric physical noise can be transformed into biased noise at the level of the logical GKP-qubit when the latter is encoded using a rectangular lattice. We alos compute the respective logical GKP-qubit error probabilities.

- Section 5.3 reviews some prior results which are directly relevant to our analysis, e.g., on error thresholds of the concatenated surface-GKP and toric-GKP codes and on the enhanced noise resilience of the surface code against biased noise.

- Section 5.4 introduces *the modified asymmetric surface-GKP code* as well as some analytic results on the noise bias of the encoded GKP-qubits.

- In Section 5.5, we present the simulation methods that we use to numerically compute the error threshold of this code for different asymmetry parameters.

- The results of these simulations, i.e., the enhanced threshold estimates achieved by an increased asymmetry ratio, are given in Section 5.6.

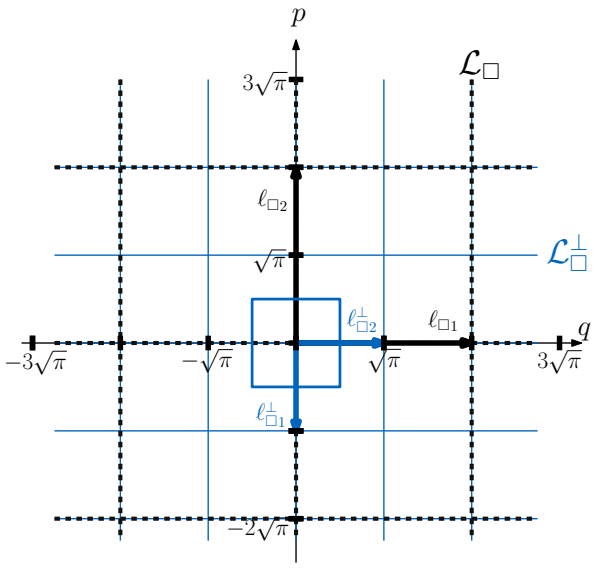


Figure 5.1: Example of a symplectically integral lattice in the two-dimensional phase space: The square lattice $\mathcal{L}_\square$ (dotted black lines) is generated by the two vectors $\ell_{\square 1} = 2\sqrt{\pi}e_1$ and $\ell_{\square 2} = 2\sqrt{\pi}e_2$ (black arrows) with respect to the standard basis e_1, e_2 of $\mathbb{R}^2$ and the corresponding dual lattice $\mathcal{L}_\square^\perp$ (solid blue lines) is generated by $\ell_{\square 1}^\perp = -\sqrt{\pi}e_2$ and $\ell_{\square 2}^\perp = \sqrt{\pi}e_1$. The area around the origin enclosed by the blue square is the Voronoi cell of the dual lattice, see Definition 5.3.

5.1 The Gottesman-Kitaev-Preskill (GKP) code

In 2001, Gottesman, Kitaev and Preskill introduced a class of stabiliser codes that encode qubits into bosonic modes [71] and that we work with throughout this chapter.

In Section 5.1.1, we define general GKP codes which encode a finite-dimensional quantum system in n harmonic oscillator modes. Section 5.1.2 presents syndrome measurements and error correction, especially against probabilistic displacement noise, in this setup. Sections 5.1.3 and 5.1.4 focus in more detail on the square lattice GKP-qudits and qubits, respectively.

5.1.1 The GKP code: basic definition

Symplectically integral lattices on the phase space

Central to the definition of GKP codes are the following lattices on the phase space.

Definition 5.1 *(Symplectically integral lattice).* For $n \in \mathbb{N}$, let $J \in \mathbb{R}^{2n\times 2n}$ be the matrix defining the symplectic form (cf. Eq. (2.19)). Let $\{\ell_i\}_{i=1}^{2n} \subset \mathbb{R}^{2n}$ and $\{\ell_i^\perp\}_{i=1}^{2n} \subset \mathbb{R}^{2n}$ be such that

$$\ell_i^T J \ell_j \in 2\pi\mathbb{Z} \ , \qquad \text{for all } i,j = 1,\dots,2n \ , \tag{5.1}$$

$$(\ell_i^\perp)^T J \ell_j = 2\pi\delta_{ij} \ , \qquad \text{for all } i,j = 1,\dots,2n \ , \tag{5.2}$$

where $\delta_{i,j}$ is the Kronecker delta. A *symplectically integral lattice* is defined as

$$\mathcal{L} := \left\{ \sum_{k=1}^{2n} m_k \ell_k \;\middle|\; m_k \in \mathbb{Z} \text{ for every } k \right\} \ ,$$

where $\{\ell_i\}_{i=1}^{2n} \subset \mathbb{R}^{2n}$ satisfy Eq. (5.1). The *dual lattice* is defined as

$$\mathcal{L}^\perp := \left\{ \sum_{k=1}^{2n} m_k \ell_k^\perp \;\middle|\; m_k \in \mathbb{Z} \text{ for every } k \right\} \ .$$

We say that the vectors $\{\ell_i\}_{i=1}^{2n}$ span or generate the lattice $\mathcal{L}$ and that the dual lattice $\mathcal{L}^\perp$ is generated by the $2n$ vectors $\{\ell_i^\perp\}_{i=1}^{2n}$. Note that every symplectically integral lattice forms a subgroup of the additive group $(\mathbb{R}^{2n}, +)$. Instead of defining the lattice by its generating vectors it is sometimes useful to arrange them into the columns of a generating matrix $G := \begin{pmatrix} \ell_1 & \ell_2 & \cdots & \ell_{2n} \end{pmatrix}$, see e.g. the original GKP paper [71]. Fig. 5.1 shows an example of a two-dimensional symplectically integral lattice: the *square lattice* in two-dimensional phase space. Note that the corresponding GKP code is discussed in Section 5.1.4 in more detail.

Stabilisers and code space

Let us consider a physical system of $n \in \mathbb{N}$ bosonic modes. This quantum system is described by the Hilbert space $\mathcal{H}_S = \left(L^2(\mathbb{R}^n)\right)$ with associated phase space $\mathbb{R}^{2n}$. The GKP code is a stabiliser code (cf. Definition 3.18) defined on this system.

Definition 5.2 *(GKP code [71])*. Let $\mathcal{L} \subset \mathbb{R}^{2n}$ be a symplectically integral lattice spanned by vectors $\{\ell_i\}_{i=1}^{2n} \in \mathbb{R}^{2n}$. The *GKP stabiliser generators* are $\{D(\ell_i)\}_{i=1}^{2n}$, cf. Eq. (3.9) from Section 3.1.2, i.e., the stabiliser group

$$\mathcal{S}(\mathcal{L}) \coloneqq \langle D(\ell_1), \ldots, D(\ell_{2n}) \rangle = \{D(\xi) \mid \xi \in \mathcal{L}\} \tag{5.3}$$

consists of Weyl displacement operators (2.22) along the symplectically integral lattice $\mathcal{L}$. The stabiliser code associated with the stabiliser group $\mathcal{S}(\mathcal{L})$ is called the *GKP code* $\mathsf{GKP}(\mathcal{L})$.

We note that by condition (5.1), the stabiliser elements commute. The stabiliser generators are displacements along the lattice's spanning vectors $D(\ell_i)$ for $i = 1, \ldots, 2n$. Similar to stabiliser codes in the finite-dimensional setting (Definition 3.19), the GKP code space is the joint $+1$ eigenspace of the stabiliser generators (5.3). More precisely, elements of the GKP code space are those elements of the n-mode Hilbert space which are invariant under transformations described by Weyl displacements $D(\xi)$ along the lattice elements $\xi \in \mathcal{L}$. We denote the GKP code space by

$$\mathsf{GKP}(\mathcal{L}) \subseteq L^2(\mathbb{R}^n)$$

to indicate its dependence on the symplectically integral lattice $\mathcal{L}$. The encoding channel of a GKP code is denoted by

$$\mathcal{E}_{\mathcal{L}} : \mathcal{D}(\mathbb{C}^d) \to \mathcal{D}(L^2(\mathbb{R}^n)) \ . \tag{5.4}$$

The phase space $\mathbb{R}^{2n}/\mathcal{L}$ associated with the code space $\mathsf{GKP}(\mathcal{L})$ corresponds to a torus in $2n$ dimensions. Recall that the displacements give a representation $(\alpha, \xi) \mapsto e^{-i\alpha} D(\xi)$ of the Heisenberg-Weyl group H_n from Eq. (3.10) on $\mathcal{S}'(\mathbb{R}^n)$ – which restricts to an irreducible unitary representation on $L^2(\mathbb{R}^n)$ – and the GKP stabilisers form a discrete subgroup of H_n. The associated Hilbert space $\mathsf{GKP}(\mathcal{L})$ carries an irreducible representation of a discrete Heisenberg-Weyl group, the one generated by 'logical' displacements, as we explain in the next paragraph.

Logical operators

The logical operators are the elements of the centraliser of the stabiliser group inside the Heisenberg Weyl group, i.e., the displacement operators which leave the code space invariant. Again up to global phases, this centraliser is given by

$$\begin{aligned} Z\big(\mathcal{S}(\mathcal{L})\big) &\coloneqq \big\{D(\nu) \mid \nu \in \mathbb{R}^{2n} \text{ such that } SD(\nu) = D(\nu)S \text{ for all } S \in \mathcal{S}(\mathcal{L})\big\} \\ &= \big\{D(\nu) \mid \nu \in \mathbb{R}^{2n} \text{ such that } D(\xi)D(\nu) = D(\nu)D(\xi) \text{ for all } \xi \in \mathcal{L}\big\} \ . \end{aligned}$$

Using the definition of the dual lattice (5.2), it is easy to see that

$$Z\big(\mathcal{S}(\mathcal{L})\big) = \{D(\nu) \mid \nu \in \mathcal{L}^\perp\} = \langle D(\ell_1^\perp), \ldots, D(\ell_{2n}^\perp) \rangle \ .$$

Hence the logical operations are Weyl displacements $D(\nu)$ along elements $\nu \in \mathcal{L}^\perp$ of the dual lattice. Note that two such displacements $D(\nu)$ and $D(\nu')$ have the same logical effect – they act identically on the code space – if and only if they satisfy (up to a global phase) $D(\nu)S = D(\nu')$ for some stabiliser $S \in \mathcal{S}(\mathcal{L})$. By definition of the GKP stabiliser group (5.3), this is the case if and only if $\nu - \nu' \in \mathcal{L}$. As a consequence, a complete set of inequivalent logical operations is characterised by elements of $\mathcal{L}^\perp/\mathcal{L}$. Its elements are cosets $[\xi]$ of $\mathcal{L}$ where $\xi \in \mathcal{L}^\perp$ is a representative of the coset $[\xi] = \xi + \mathcal{L}$. The set

$$\Big\{D(\xi) \mid [\xi] \in \mathcal{L}^\perp/\mathcal{L}\Big\} \ , \tag{5.5}$$

forms a complete family of inequivalent logical operations.

One can use this to compute the dimension of the code space $\mathsf{GKP}(\mathcal{L})$. By (5.5), the number of logical or encoded (generalised) Pauli operators is equal to the number of elements in the coset space $\mathcal{L}^\perp/\mathcal{L}$. The latter corresponds to the ratio of the unit cell volume of $\mathcal{L}$ divided by the unit cell volume of $\mathcal{L}^\perp$. Let us compute this volume ratio: Recall that the generating vectors of $\mathcal{L}$ must satisfy Eq. (5.1) which can be rewritten as $\ell_i^T J \ell_j = 2\pi A_{ij}$ where A is an antisymmetric matrix with integral entries. In terms of the matrix A Eq. (5.2) – defining the dual lattice vectors – translates to

$$\ell_i = \sum_j A_{ij} \ell_j^\perp \ .$$

Then the volume ratio considered above – the volume of the unit cell of $\mathcal{L}$ divided by the volume of the unit cell of $\mathcal{L}^\perp$ – is given by the determinant of A. Since on $\mathbb{C}^d$ there are d^2 (generalised) Pauli operators, the dimension of the code space $\mathsf{GKP}(\mathcal{L})$ is equal to the square root of the number $\det(A)$ computed above:

$$\dim\left(\mathsf{GKP}(\mathcal{L})\right) = \sqrt{\det(A)} \ . \tag{5.6}$$

5.1.2 GKP recovery for displacement errors

Syndrome measurements

As usual for stabiliser codes, the syndrome measurements correspond to measuring the eigenvalues of a complete set of $2n$ independent stabiliser generators. For the code $\mathsf{GKP}(\mathcal{L})$, these are displacements $D(\ell_1), \ldots, D(\ell_{2n})$ along basis vectors of the symplectically integral lattice $\mathcal{L}$. To further understand the corresponding measurement outcome – the syndrome $s = (s_1, \ldots, s_{2n})$ – it is convenient to introduce the concept of a closest lattice point, and the lattice modulo operator.

Definition 5.3 *(Closest lattice point and Voronoi cell).* Let $\mathcal{L} \subset \mathbb{R}^{2n}$ be a symplectically integral lattice. For $x \in \mathbb{R}^{2n}$, the *closest lattice point* in $\mathcal{L}$ to x is

$$Q_{\mathcal{L}}(x) \coloneqq \mathsf{argmin}_{\xi \in \mathcal{L}} \|x - \xi\| \ . \tag{5.7}$$

Here $\|\cdot\|$ is the Euclidean norm and ties are broken in a systematic manner, i.e., such that $Q_{\mathcal{L}}(x)_j \geq x_j$ for all components $j = 1, \ldots, 2n$.

The *Voronoi cell* $\mathcal{V}_{\mathcal{L}}$ of the lattice $\mathcal{L}$ is defined as the set of points $x \in \mathbb{R}^{2n}$ to which the origin is the closest lattice point, i.e., as

$$\mathcal{V}_{\mathcal{L}} \coloneqq \{x \in \mathbb{R}^{2n} \mid Q_{\mathcal{L}}(x) = 0\} \ .$$

The *modulo lattice operation* $(\cdot \mod \mathcal{L})$ is defined as the map

$$\mathbb{R}^{2n} \to \mathcal{V}_{\mathcal{L}} \quad , \quad x \mapsto (x \mod \mathcal{L}) \coloneqq x - Q_{\mathcal{L}}(x) \ .$$

Let us return to the GKP code $\mathsf{GKP}(\mathcal{L})$. At the level of the phase space, the GKP syndrome measurement corresponds to measuring the displacement error modulo the dual lattice. More precisely, if an initial ideal GKP code state is corrupted by a displacement $D(\nu)$ for $\nu \in \mathbb{R}^{2n}$

then the syndrome associated with the error ν is

$$s(\nu) = \nu \mod \mathcal{L}^\perp = \nu - Q_{\mathcal{L}^\perp}(\nu) \ . \tag{5.8}$$

For every $\nu \in \mathbb{R}^{2n}$, the syndrome $s(\nu)$ is an element of the dual Voronoi cell $\mathcal{V}_{\mathcal{L}^\perp}$.

Decoding

Generally a *decoding strategy* or *decoder* is defined by a map

$$c : \mathcal{V}_{\mathcal{L}^\perp} \to \mathbb{R}^{2n} \qquad , \qquad s \mapsto c(s) \ ,$$

which associates a correction vector $c(s) \in \mathbb{R}^{2n}$ to every syndrome $s \in \mathcal{V}_{\mathcal{L}^\perp}$. The GKP recovery procedure corresponds to a phase space displacement $D(c(s))$ of the corrupted state by the correction vector $c(s) \in \mathbb{R}^{2n}$. To analyse the validity and the success of a decoder, assume an error $D(\nu)$ has corrupted an initial ideally encoded GKP state. We use the following terminology:

(i) The decoder is *valid* for an error $\nu \in \mathbb{R}^{2n}$ if the correction displacement maps the corrupted state back to the code space $\mathsf{GKP}(\mathcal{L})$. The corrected state is in the code space if it relates to the original state by a phase space displacement by a vector of the dual lattice. Phrased differently, a decoder is valid for an error $\nu \in \mathbb{R}^{2n}$ if the overall displacement applied to the encoded state, i.e., the product of the error and the correction $D(c(s))D(\nu)$ corresponding to an element of the dual lattice $\mathcal{L}^\perp$, i.e., if

$$\nu + c(s(\nu)) \in \mathcal{L}^\perp \ . \tag{5.9}$$

(ii) If the overall displacement applied to the encoded state is an element of the lattice $\mathcal{L}$, then the error and correction has no logical effect on the encoded state. Hence the decoder corrects an error $\nu \in \mathbb{R}^{2n}$ *successfully* if the overall displacement amounts to a shift along the lattice $\mathcal{L}$. This is the case if

$$\nu + c(s(\nu)) \in \mathcal{L} \ . \tag{5.10}$$

More generally, an error $\nu \in \mathbb{R}^{2n}$ results in the logical action of the operator $\overline{L}$ associated with the coset $[\xi_{\overline{L}}] \in \mathcal{L}^\perp/\mathcal{L}$ if it satisfies

$$\nu + c(s(\nu)) \in [\xi_{\overline{L}}] \ . \tag{5.11}$$

Note that here one writes $[\xi_{\overline{L}}] = \xi_{\overline{L}} + \mathcal{L}$ to denote the coset (element of (5.5)) of the phase space element $\xi_{\overline{L}} \in \mathbb{R}^{2n}$ such that the corresponding displacement $D(\xi_{\overline{L}})$ acts as a logical L-operator on the encoded GKP state. The recovery channel corresponding to a decoder c is denoted by $\mathcal{R}_{\mathcal{L},c}$. Let us furthermore denote by $\mathcal{E}_{\mathcal{L}}^\dagger$ the inverse encoding map, which is defined on states supported on the code space $\mathsf{GKP}(\mathcal{L})$ and satisfies $\mathcal{E}_{\mathcal{L}}^\dagger \circ \mathcal{E}_{\mathcal{L}} = I_{\mathcal{D}(\mathbb{C}^d)}$.

Closest lattice point decoding

The usual decoding strategy for the GKP code is *closest lattice point decoding*. Here one associates the correction vector

$$c(s) = -s$$

to the syndrome $s \in \mathcal{V}_{\mathcal{L}^\perp}$. This corresponds to a shift to the lattice point in $\mathcal{L}^\perp$ which is closest to the error displacement ν since

$$c(s(\nu)) + \nu = -s(\nu) + \nu = -\left(\nu - Q_{\mathcal{L}^\perp}(\nu)\right) + \nu = Q_{\mathcal{L}^\perp}(\nu) \ , \tag{5.12}$$

where $Q_{\mathcal{L}^\perp}(\nu)$ is defined in Eq. (5.7). This decoder is valid – i.e., it satisfies Eq. (5.9) for all $\nu \in \mathbb{R}^{2n}$ – since $Q_{\mathcal{L}^\perp}(\nu) \in \mathcal{L}^\perp$ by definition of $Q_{\mathcal{L}^\perp}$. Furthermore, it is successful for the displacement error $D(\nu)$ if and only if it satisfies (5.10) for $\nu \in \mathbb{R}^{2n}$. By Eq. (5.12), this is equivalent to the condition that $Q_{\mathcal{L}^\perp}(\nu)$ belongs to the trivial coset $[0] \in \mathcal{L}^\perp/\mathcal{L}$.

Suppose a displacement error $D(\nu)$ corrupts an ideal GKP state, followed by the GKP syndrome measurement and closest lattice point correction. The residual logical operator generally depends on the coset to which $Q_{\mathcal{L}^\perp}(\nu)$ belongs: If the displacement error ν satisfies $Q_{\mathcal{L}^\perp}(\nu) \in [\xi_{\overline{L}}]$, then a residual logical operator $\overline{L}$ occurs. Here $[\xi_{\overline{L}}] \in \mathcal{L}^\perp/\mathcal{L}$ denotes the coset associated with logical operator $\overline{L}$. Since

$$Q_{\mathcal{L}^\perp}(\nu) \in [\xi_{\overline{L}}] \quad \Leftrightarrow \quad \nu - \nu \mod \mathcal{L}^\perp = \xi_{\overline{L}} + \xi \quad \text{for some } \xi \in \mathcal{L} \ ,$$

this can be summarised as follows: The combined channel – displacement noise $D(\nu)$ and GKP recovery on GKP states – results in the logical action

$$\overline{L} \qquad \text{if} \qquad \nu = \xi_{\overline{L}} + \xi + \eta \quad \text{for some } \xi \in \mathcal{L}, \eta \in \mathcal{V}_{\mathcal{L}^\perp} \ . \tag{5.13}$$

Let us denote the recovery channel associated with the closest lattice point decoding by $\mathcal{R}_{\mathcal{L}}$.

5.1.3 Square lattice GKP-qudit

Let us consider the special case of a single bosonic mode, i.e., $n = 1$. Then a symplectically integral lattice $\mathcal{L} \subset \mathbb{R}^2$ is generated by two vectors ℓ_1, ℓ_2 and Eq. (5.1) simplifies to the condition

$$\ell_1^T J \ell_2 = \pm 2\pi d \tag{5.14}$$

for some $d \in \mathbb{N}$. By Eq. (5.6), the dimension of the code space $\mathsf{GKP}(\mathcal{L})$ is given by the number d: As a consequence, the code $\mathsf{GKP}(\mathcal{L})$ encodes a *qudit* into one bosonic mode.

Definition 5.4 *(Square lattice GKP-qudit).* The *square lattice GKP-qudit* code $\mathsf{GKP}(\mathcal{L})$ is defined by the lattice $\mathcal{L} \subset \mathbb{R}^2$ spanned by the two vectors

$$\ell_1 = \begin{pmatrix} \sqrt{2\pi d} \\ 0 \end{pmatrix} \quad , \quad \ell_2 = \begin{pmatrix} 0 \\ \sqrt{2\pi d} \end{pmatrix} \ ,$$

for some $d \in \mathbb{N}$ such that $d \geq 2$.

It is easy to see that the two vectors (ℓ_1, ℓ_2) satisfy Eq. (5.14), i.e., the lattice is symplectically integral and this code actually corresponds to a GKP-qudit code.

To investigate such codes in more detail, let us discuss a construction of the corresponding code words and stabiliser operators by Bouzouina and de Bièvre [24]. In 1996, the authors studied certain area preserving maps on the single bosonic mode Hilbert space which is associated with a torus phase space. Note that this foundation was established before Gottesman, Kitaev and Preskill discovered its use in the context of quantum error correction later in 2001 [71]: in the context of GKP codes, the following construction is related to square GKP

lattices (described in the section at hand) and rectangular GKP lattices (cf. Section 5.2.2). We additionally note that Bouzouina's and de Bièvre's work builds on previous ideas of [78] and [14].

When describing the action of logical operators it is useful to fix a basis of $\mathsf{GKP}(\mathcal{L})$. Therefore let $\psi \in \mathsf{GKP}(\mathcal{L}) \subset \mathcal{S}'$, i.e., $D(\ell)\psi = \psi$ for all $\ell \in \mathcal{L}$. From the eigenvalue equation for ℓ_2

$$D(\ell_2)\psi \coloneqq e^{i\sqrt{2\pi d}Q}\psi = \psi \ ,$$

we find that ψ in position space (i.e., Q eigenbasis) must satisfy $\psi(x)(1 - e^{i\sqrt{2\pi d}\,x}) = 0$ for all $x \in \mathbb{R}$. As a consequence $\psi(x)$ can only be non-zero for $x = m\sqrt{2\pi/d}$ where $m \in \mathbb{Z}$, which implies that

$$\psi(x) = \sum_{m\in\mathbb{Z}} c_m \delta\left(x - m\sqrt{2\pi/d}\right) \ , \tag{5.15}$$

for some coefficients $c_m \in \mathbb{C}$ for $m \in \mathbb{N}$. Using the other spanning vector of $\mathcal{L}$ and the corresponding eigenvalue equation

$$D(\ell_1)\psi = e^{-i\sqrt{2\pi d}P}\psi = \psi$$

we find that these coefficients must satisfy

$$c_m = c_{m+d} \qquad \text{for all } m \in \mathbb{N} \ . \tag{5.16}$$

By Eqs. (5.15) and (5.16), the dimension of $\mathsf{GKP}(\mathcal{L})$ is d. Defining

$$e_j(x) = \sum_{m\in\mathbb{Z}} \delta\big(x - m\sqrt{2\pi d} - j\sqrt{2\pi/d}\big) \qquad \text{for } j = 0, \ldots, d-1 \ , \tag{5.17}$$

then gives a basis $\{e_j\}_{j=0}^{d-1}$ of $\mathsf{GKP}(\mathcal{L})$.

Since the logical operators are associated with the set of cosets $\mathcal{L}^\perp/\mathcal{L}$, let us consider $D(\xi)$ for $\xi \in \mathcal{L}^\perp$. The spanning vectors of the symplectically dual lattice are given by

$$\ell_1^\perp \coloneqq \begin{pmatrix} 0 \\ -\sqrt{2\pi/d} \end{pmatrix} \qquad , \qquad \ell_2^\perp \coloneqq \begin{pmatrix} \sqrt{2\pi/d} \\ 0 \end{pmatrix} \ .$$

Displacements along the dual lattice satisfy

$$D(m_1\ell_1^\perp + m_2\ell_2^\perp) = e^i D(\ell_1^\perp)^{m_1} D(\ell_2^\perp)^{m_2} \ ,$$

for $m_1, m_2 \in \mathbb{Z}$. The action of logical error operators on these basis vectors can be computed to be

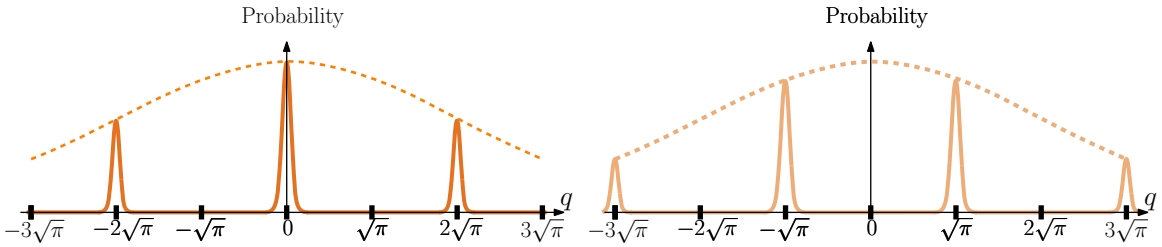
$$\begin{aligned} D(\ell_1^\perp)e_j &= e^{-i2\pi j/d} e_j \ , \\ D(\ell_2^\perp)e_j &= e_{(j+1) \mod d} \ , \end{aligned}$$

for all $j = 0, \ldots, d-1$. That is, $D(-\ell_1^\perp)$ and $D(\ell_2^\perp)$ are the (generalised) logical Pauli-$\overline{Z}$ and $\overline{X}$ operators, respectively, of a d-dimensional system. They generate what is sometimes referred to as a finite-dimensional Weyl system.

Of course, the elements $\psi \in \mathsf{GKP}(\mathcal{L}) \subset \mathcal{S}'$ are non-normalisable. When one wants to consider physically realisable – normalisable – states one usually considers normalisable approximations of the element $\psi \in \mathcal{S}'$ [71] such as depicted in Fig. 5.2. For a squeezing parameter $\epsilon > 0$ one sets

$$e_j^{(\epsilon)}(x) = \frac{1}{2\pi^2} \sum_{m\in\mathbb{Z}} e^{-\frac{\epsilon^2}{2}(m\sqrt{2\pi d})^2} e^{-\frac{1}{2\epsilon^2}(x - m\sqrt{2\pi d} - j\sqrt{2\pi/d})^2} \ , \tag{5.18}$$

for $j = 0, \ldots, d-1$ which approximates $\lim_{\epsilon\to\infty} e_j^{(\epsilon)} = e_j$ in the limit $\epsilon \to \infty$ of infinite squeezing.



(a) Approximation of e_0 (b) Approximation of e_1

Figure 5.2: Probability distribution in position space of finitely squeezed approximations of the logical states e_0 and the e_1 from Eq. (5.17) for $d = 2$ (the square lattice GKP-qubit), i.e., $e_j^{(\epsilon)}$ for $j = 0, 1$ from Eq. (5.18).

5.1.4 Square lattice GKP-qubit

The square lattice GKP-qubit code encodes a single qubit into a single bosonic mode with phase space $\mathbb{R}^2$. The associated symplectically integral lattice $\mathcal{L}_\square$ is generated by the vectors

$$\ell_{\square 1} := \begin{pmatrix} 2\sqrt{\pi} \\ 0 \end{pmatrix} \quad , \quad \ell_{\square 2} := \begin{pmatrix} 0 \\ 2\sqrt{\pi} \end{pmatrix} , \tag{5.19}$$

i.e., it corresponds to the lattice associated with the GKP-qudit from Definition 5.4 for $d = 2$. This is the simplest and most prominent example of the GKP code. To clarify our notation, we will use an index $\square$ to label the respective code space, logical operators etc. From now on we will call $\mathsf{GKP}(\mathcal{L}_\square)$ the *square lattice GKP code*.

The dual lattice $\mathcal{L}_\square^\perp$ is generated by the two vectors

$$\ell_{\square 1}^\perp := \begin{pmatrix} 0 \\ -\sqrt{\pi} \end{pmatrix} \quad , \quad \ell_{\square 2}^\perp := \begin{pmatrix} \sqrt{\pi} \\ 0 \end{pmatrix} .$$

The lattice, its dual and the Voronoi cell are given by the following sets:

$$\mathcal{L}_\square := \left\{ \begin{pmatrix} 2\sqrt{\pi}m_1 & 2\sqrt{\pi}m_2 \end{pmatrix}^T \mid m_1, m_2 \in \mathbb{Z} \right\} ,$$
$$\mathcal{L}_\square^\perp := \left\{ \begin{pmatrix} \sqrt{\pi}m_1 & \sqrt{\pi}m_2 \end{pmatrix}^T \mid m_1, m_2 \in \mathbb{Z} \right\} , \tag{5.20}$$
$$\mathcal{V}_\square^\perp := \mathcal{V}_{\mathcal{L}_\square^\perp} = \left\{ \begin{pmatrix} \sqrt{\pi}\lambda_1 & \sqrt{\pi}\lambda_2 \end{pmatrix}^T \mid \lambda_1, \lambda_2 \in \left[-\tfrac{1}{2}, \tfrac{1}{2} \right] \right\} . \tag{5.21}$$

Logical states and operators

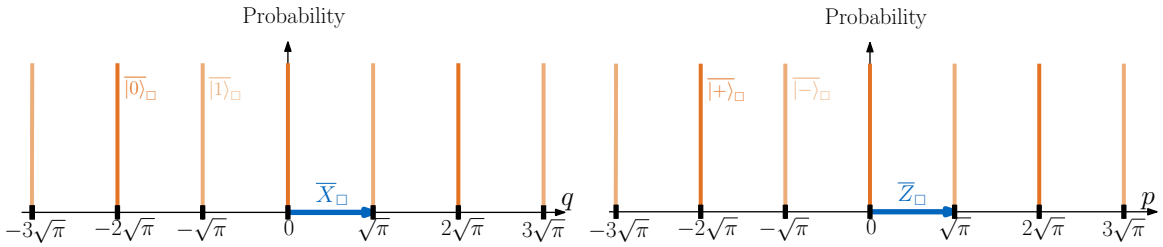
As discussed in the previous section, the two-dimensional code space $\mathsf{GKP}(\mathcal{L}_\square)$ has the basis

$$|\overline{0}\rangle_\square := \sum_{k\in\mathbb{Z}} \delta(q - 2k\sqrt{\pi})|q\rangle = \sum_{k\in\mathbb{Z}} |q = 2k\sqrt{\pi}\rangle , \tag{5.22}$$
$$|\overline{1}\rangle_\square := \sum_{k\in\mathbb{Z}} \delta(q - (2k+1)\sqrt{\pi})|q\rangle = \sum_{k\in\mathbb{Z}} |q = (2k+1)\sqrt{\pi}\rangle . \tag{5.23}$$

In the position basis, these code state elements are infinite sums of δ-peaks around even and odd multiples of $\sqrt{\pi}$, respectively.

The logical X and Z operations are given by

$$\overline{X}_\square := D\left(\ell_{\square 2}^\perp\right) = e^{-i\sqrt{\pi}P} \quad , \quad \overline{Z}_\square := D\left(-\ell_{\square 1}^\perp\right) = e^{i\sqrt{\pi}Q} .$$



(a) Logical states in position basis (b) Logical states in momentum basis

Figure 5.3: Logical states of $\mathsf{GKP}(\mathcal{L}_\square)$ in position/momentum basis. In the position basis, the $|\overline{0}\rangle_\square$-state (respectively, the $|\overline{1}\rangle_\square$-state) is given by the superposition infinitely sharp δ-peaks around even (respectively, odd) multiples of $\sqrt{\pi}$. In the momentum basis, the same holds for the state $|\overline{+}\rangle_\square$ (respectively, $|\overline{-}\rangle_\square$). The logical $\overline{X}_\square$ corresponds to a q-shift of $\sqrt{\pi}$, mapping the logical $|\overline{0}\rangle_\square$ to the logical $|\overline{1}\rangle_\square$ and vice versa. Similarly, the logical $\overline{Z}_\square$ shifts p by an amount of $\sqrt{\pi}$ and maps $|\overline{+}\rangle_\square \mapsto |\overline{-}\rangle_\square$ and vice versa.

Note that the logical operators are not self-inverse, $\overline{X}_\square^{-1} \neq \overline{X}_\square$ and $\overline{Z}_\square^{-1} \neq \overline{Z}_\square$, but that their logical action on the code words is, i.e., they satisfy $\overline{X}_\square^{-1}\psi = \overline{X}_\square\psi$ and $\overline{Z}_\square^{-1}\psi = \overline{Z}_\square\psi$ for all $\psi \in \mathsf{GKP}(\mathcal{L}_\square)$. The basis of $\mathsf{GKP}(\mathcal{L}_\square)$ in which $\overline{Z}_\square$ is diagonal is given by the logical $+$ and $-$ states

$$|\overline{+}\rangle_\square \coloneqq \sum_{k\in\mathbb{Z}} \delta(p - 2k\sqrt{\pi})|p\rangle = \sum_{k\in\mathbb{Z}} |p = 2k\sqrt{\pi}\rangle \ ,$$
$$|\overline{-}\rangle_\square \coloneqq \sum_{k\in\mathbb{Z}} \delta(p - (2k+1)\sqrt{\pi})|p\rangle = \sum_{k\in\mathbb{Z}} |p = (2k+1)\sqrt{\pi}\rangle \ .$$

For the square lattice GKP code, Fig. 5.1 shows the lattice $\mathcal{L}_\square$ as well as its dual $\mathcal{L}_\square^\perp$. In Fig. 5.3, the encoded logical states as well as the logical operations are depicted in the position or momentum basis.

Logical Clifford gates

Consider $n \in \mathbb{N}$ square lattice GKP-qubits, i.e., let n copies of the square lattice $\mathcal{L}_\square$ be given. In order to perform universal quantum computation, one must be able to implement a universal gate set on the encoded qubits. Here we discuss how the encoded gates look like on the bosonic level. By Theorem 2.34, the *Clifford group* on n qubits is a part of a universal gate set. It consists of CNOT gates between pairs of qubit, as well as Hadamard and phase gates on every single qubit. These gates transform the logical X- and Z-operators as described by Eqs. (2.57), (2.52) and (2.55), respectively.

Recall the examples of Gaussian unitary gates from Section 2.2.4. First, the unitary

$$\overline{H} \coloneqq e^{i\frac{\pi}{4}(Q^2+P^2)} = U_{\mathrm{rot}}(-\pi/2)$$

was shown to have the associated symplectic matrix $S_{\mathrm{rot}}(-\pi/2)$ from Eq. (2.42) and to transform the quadratures as $\overline{H}Q\overline{H}^\dagger = P$ and $\overline{H}P\overline{H}^\dagger = -Q$. The unitary $\overline{H}$ furthermore satisfies

$$\overline{H}\overline{X}_\square\overline{H}^\dagger = \overline{H}e^{-i\sqrt{\pi}P}\overline{H}^\dagger = e^{i\sqrt{\pi}Q} = \overline{Z}_\square$$
$$\overline{H}\overline{Z}_\square\overline{H}^\dagger = \overline{H}e^{i\sqrt{\pi}Q}\overline{H}^\dagger = e^{-i\sqrt{\pi}P} = \overline{X}_\square$$

which corresponds to the action a *Hadamard* gate must have on the logical $\overline{X}_\square$- and $\overline{Z}_\square$-operators as computed in Eq. (2.52). As discussed in Section 2.2.4, $\overline{H}$ is a passive Gaussian unitary.

Second, recall the active Gaussian unitary

$$\overline{S} \coloneqq e^{\frac{i}{2}Q^2} = U_{\mathrm{act,Q}}(-1)$$

from Eq. (2.44) which hence satisfies $\overline{S}Q\overline{S}^\dagger = Q$ and $\overline{S}P\overline{S}^\dagger = P - Q$. As a consequence and using the Baker-Campbell-Hausdorff formula, we find that

$$\begin{aligned}
\overline{S}\overline{X}_\square\overline{S}^\dagger &= \overline{S}e^{-i\sqrt{\pi}P}\overline{S}^\dagger = e^{-i\sqrt{\pi}(P-Q)} = e^{-i\sqrt{\pi}P}e^{i\sqrt{\pi}Q}e^{i\frac{\pi}{2}} = i\overline{X}_\square\overline{Z}_\square \ , \\
\overline{S}\overline{Z}_\square\overline{S}^\dagger &= \overline{S}e^{i\sqrt{\pi}Q}\overline{S}^\dagger = e^{i\sqrt{\pi}Q} = \overline{Z}_\square \ ,
\end{aligned}$$

hold. This corresponds to the action (2.55) of the single-qubit phase gate $M_{\pi/4} = \mathrm{diag}(1, e^{i\pi/4})$ on the logical X and Z operators.

Third, for the CNOT gate, consider the two-mode unitary

$$\overline{\mathsf{CNOT}} \coloneqq e^{iQ_1P_2} \ . \tag{5.24}$$

By Lemma 2.31, one can compute its associated symplectic matrix: if a unitary is of the form $U = e^{-\frac{i}{2}R^TAR}$, then it transforms the quadratures as $UR_jU^\dagger = \sum_k S_{jk}R_k$ where $S \coloneqq e^{-JA}$. For the CNOT unitary $U = \overline{\mathsf{CNOT}} \coloneqq e^{iQ_1P_2}$, this results in

$$A = -\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \quad , \quad S = e^{-JA} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \ ,$$

i.e., it transforms the mode operators as

$$\begin{aligned}
\overline{\mathsf{CNOT}}Q_1\overline{\mathsf{CNOT}}^\dagger &= Q_1 \quad , & \overline{\mathsf{CNOT}}P_1\overline{\mathsf{CNOT}}^\dagger &= P_1 - P_2 \ , \\
\overline{\mathsf{CNOT}}Q_2\overline{\mathsf{CNOT}}^\dagger &= Q_1 + Q_2 \quad , & \overline{\mathsf{CNOT}}P_2\overline{\mathsf{CNOT}}^\dagger &= P_2 \ .
\end{aligned}$$

Therefore, the logical $\overline{X}_\square$ and $\overline{Z}_\square$ are transformed as

$$\begin{aligned}
\overline{\mathsf{CNOT}}\,\overline{X}_{\square 1}\overline{\mathsf{CNOT}}^\dagger &= \overline{\mathsf{CNOT}}e^{-i\sqrt{\pi}P_1}\overline{\mathsf{CNOT}}^\dagger = e^{-i\sqrt{\pi}(P_1-P_2)} = \overline{X}_{\square 1}\overline{X}_{\square 2}{}^{-1} \\
\overline{\mathsf{CNOT}}\,\overline{Z}_{\square 1}\overline{\mathsf{CNOT}}^\dagger &= \overline{\mathsf{CNOT}}e^{i\sqrt{\pi}Q_1}\overline{\mathsf{CNOT}}^\dagger = e^{-i\sqrt{\pi}Q_1} = \overline{Z}_{\square 1} \\
\overline{\mathsf{CNOT}}\,\overline{X}_{\square 2}\overline{\mathsf{CNOT}}^\dagger &= \overline{\mathsf{CNOT}}e^{-i\sqrt{\pi}P_2}\overline{\mathsf{CNOT}}^\dagger = e^{-i\sqrt{\pi}P_2} = \overline{X}_{\square 2} \\
\overline{\mathsf{CNOT}}\,\overline{Z}_{\square 2}\overline{\mathsf{CNOT}}^\dagger &= \overline{\mathsf{CNOT}}e^{i\sqrt{\pi}Q_2}\overline{\mathsf{CNOT}}^\dagger = e^{-i\sqrt{\pi}(Q_1+Q_2)} = \overline{Z}_{\square 1}\overline{Z}_{\square 2} \ .
\end{aligned}$$

This action corresponds to the one given by the CNOT gate (cf. Eqs.(2.57)) where the first GKP-qubit is the control qubit and the second GKP-qubit is the target qubit.

For universal quantum computation in the circuit model, Clifford gates are not sufficient. The last missing gate is for example the $(\pi/8)$-gate, cf. Eq. (2.56). We note that its physical realisation in the GKP code does not correspond to a Gaussian unitary. But it can be constructed by preparing Hadamard eigenstates, transforming them with an inverse phase gate and then using a gate teleportation circuit to teleport the $(\pi/8)$-gate to the GKP-qubit. For a detailed description of this procedure we refer to the original article by Gottesman, Kitaev and Preskill [71]. In a recent paper [15], Baragiola et al. showed that distillable magic states can be produced by applying GKP error correction to Gaussian input states such as the vacuum. This result proves that universality can be achieved by using only Gaussian operations without employing any further non-Gaussian resources than the non-Gaussian logical GKP states.

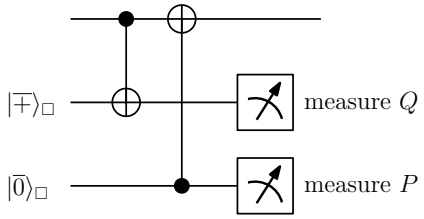


Figure 5.4: Circuit of the square lattice GKP-qubit code syndrome measurements. On the corrupted GKP state (first mode, also called data mode) one executes two encoded CNOT gates which relate it to two auxiliary modes where the latter are initialised in the encoded GKP states $|\overline{+}\rangle_\square$ and $|\overline{0}\rangle_\square$. First, the data mode is used as the control and the $|\overline{+}\rangle_\square$-mode as the target of the CNOT gate. Second, the data mode is used as target and the $|\overline{0}\rangle_\square$-mode as control. The position of the second mode is measured and the momentum of the third which yield the Q-shifts and the P-shifts on the data mode.

Syndrome measurement and error correction

The stabiliser generators are

$$D\left(\ell_{\square 1}\right) \coloneqq e^{-2i\sqrt{\pi}P} \qquad , \qquad D\left(\ell_{\square 2}\right) \coloneqq e^{2i\sqrt{\pi}Q} \ .$$

Measuring the two stabiliser generators returns the error syndrome

$$s(\nu) = \nu \mod \mathcal{L}_\square^\perp = \nu - Q_{\mathcal{L}^\perp}(\nu) \in \mathcal{V}_\square^\perp$$

for an error $\nu \in \mathbb{R}^2$, cf. Eq. (5.8). Inserting the dual square lattice $\mathcal{L}_\square^\perp$ from Eq. (5.20) code this simplifies to measuring

$$\left(Q \mod \sqrt{\pi}\right) \qquad \text{and} \qquad \left(P \mod \sqrt{\pi}\right) \ .$$

The stabiliser measurements can be implemented using a circuit which feeds the errors to an auxiliary GKP-qubit and then destructively measures the GKP-qubit, following the general procedure suggested by Steane [149]. Such a circuit uses the logical CNOT gate from Eq. (5.24) and is depicted in Fig. 5.4.

The closest lattice point decoder corresponds to the correction $c(s) = -s$, cf. Eq. (5.12). For the square lattice GKP code $\mathsf{GKP}(\mathcal{L}_\square)$ it associates the displacement

$$D\big(\nu + c(s(\nu))\big) = D\left(Q_{\mathcal{L}_\square^\perp}(\nu)\right)$$

to the error $\nu \in \mathbb{R}^2$. Fig. 5.5(b) shows the error correction procedure for a specific example of a displacement error.

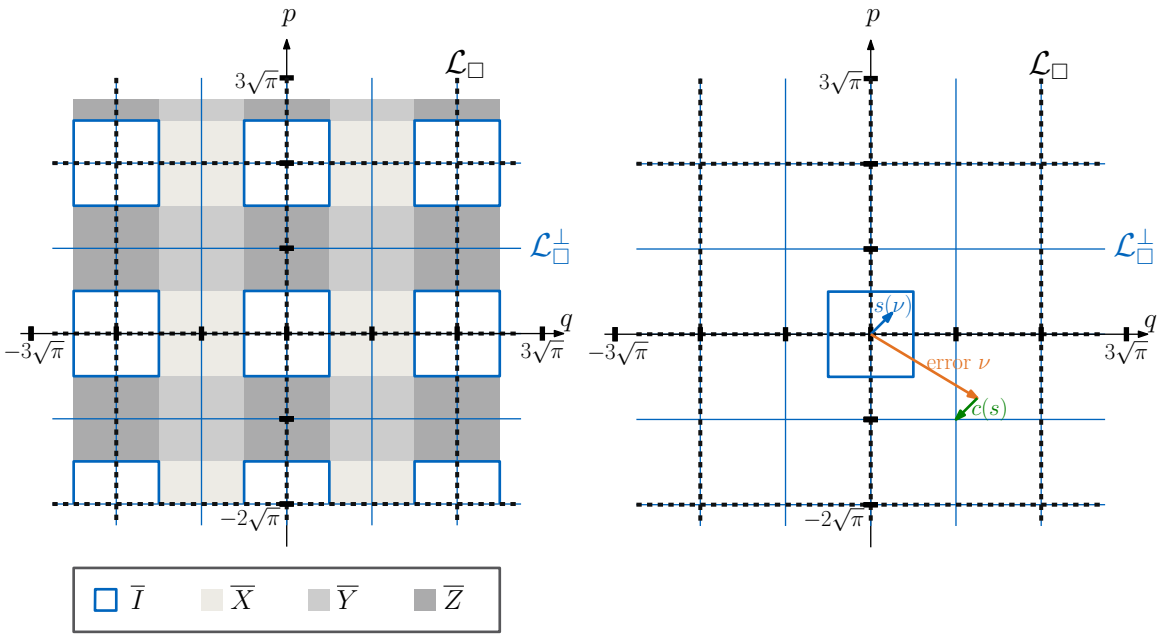
The correction is successful if $Q_{\mathcal{L}_\square^\perp}(\nu) \in \mathcal{L}_\square$. More precisely, by Eq. (5.13), after GKP error correction, the encoded GKP-qubit undergoes the action of a logical

$$\overline{L} \quad \text{if} \quad Q_{\mathcal{L}_\square^\perp}(\nu) \in [1_{L\in\{X,Y\}}\ell_{\square 2}^\perp - 1_{L\in\{Y,Z\}}\ell_{\square 1}^\perp]$$

Here $[0]$, $[\ell_{\square 2}^\perp]$, $[\ell_{\square 2}^\perp - \ell_{\square 1}^\perp]$ and $[-\ell_{\square 1}^\perp]$ are the four cosets of $\mathcal{L}_\square^\perp/\mathcal{L}_\square$ associated with the four logical operations $\overline{I}_\square$, $\overline{X}_\square$, $\overline{Y}_\square$, and $\overline{Z}_\square$, respectively. Using the definitions of $\mathcal{L}_\square$, $\mathcal{L}_\square^\perp$ and $\mathcal{V}_{\mathcal{L}_\square^\perp}$, the above conditions are equivalent to

$$\overline{L} \quad \text{if} \quad \nu = \begin{pmatrix} \sqrt{\pi}(2m_1 + 1_{L\in\{X,Y\}} + \eta_1) \\ \sqrt{\pi}(2m_2 + 1_{L\in\{Y,Z\}} + \eta_2) \end{pmatrix} \ ,$$

for $m_1, m_2 \in \mathbb{Z}$ and $\eta_1, \eta_2 \in [-\sqrt{\pi}/2, \sqrt{\pi}/2]$. These regions are illustrated in Fig. 5.5(a).



(a) Logical errors of phase space regions

(b) Example of GKP error correction

Figure 5.5: $\mathsf{GKP}(\mathcal{L}_\square)$ under the closest lattice point decoder. (a) If the displacement error $\nu \in \mathbb{R}^2$ lies inside the phase space region associated with $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$ (colouring explained below), then the GKP error correction results in the logical $\overline{L}$. (b) In this example of error ν (orange), syndrome $s(\nu) = \nu - Q_{\mathcal{L}^\perp}(\nu)$ (blue) and correction $c(s) = -s$ (green), the error correction produces a logical $\overline{Y}$ since $Q_{\mathcal{L}^\perp}(\nu) = \ell^\perp_{\square 2} + \ell^\perp_{\square 1} \in [\ell^\perp_{\square 2} - \ell^\perp_{\square 1}]$.

5.2 Symmetry and asymmetry in GKP-qubit codes on a single mode

Throughout this section, we consider GKP codes which encode a single qubit into one bosonic mode and derive some of their properties which are central to our construction of *asymmetric surface-GKP codes* (Section 5.4). In Section 5.2.1, start by showing that symplectic transformations define equivalence classes on the set of GKP codes an that for single-qubit codes there is only a single one. In Section 5.2.2, we define asymmetric GKP codes and in Section 5.2.3, we present the physical noise model of probabilistic displacement noise which we consider throughout. Section 5.2.4 computes the logical noise channel at the level of the encoded GKP-qubit in terms of the underlying GKP lattice and the physical noise model. In Section 5.2.5, we show that an asymmetric GKP lattice renders an initially symmetric Gaussian displacement noise biased towards one phase-space direction and we compute how the probabilities in the logical noise channels of this asymmetric GKP-qubit relates to the one of the symmetric GKP code. At the end, we also comment on the amount of squeezing introduced by these asymmetric codes.

In the following, let $\mathcal{L} \subset \mathbb{R}^2$ be a symplectically integral lattice which is generated by two vectors ℓ_1, ℓ_2 that satisfy

$$\ell_1^T J \ell_2 = 4\pi \ . \tag{5.25}$$

The associated GKP code $\mathsf{GKP}(\mathcal{L})$ encodes a single qubit into one oscillator mode.

5.2.1 Unitarily transformed encoding of the square lattice GKP code

For $S \in \mathsf{Sp}(2,\mathbb{R})$ and a symplectically integral lattice $\mathcal{L} \subset \mathbb{R}^2$ denote by

$$\rho \mapsto U_S \mathcal{E}_{\mathcal{L}}(\rho) U_S^\dagger \quad \text{for } \rho \in \mathcal{D}(\mathbb{C}^2) \tag{5.26}$$

a unitarily transformed GKP-encoding. Recall the square lattice GKP code $\mathsf{GKP}(\mathcal{L}_\square)$ introduced in Section 5.1.4. First we show that every GKP code can be constructed from the code $\mathsf{GKP}(\mathcal{L}_\square)$ by such a unitarily transformed encoding.

Lemma 5.5. Let $\mathcal{L} \subset \mathbb{R}^2$ be a symplectically integral lattice (cf. Definition 5.1) on the phase space $\mathbb{R}^2$ and let $\mathsf{GKP}(\mathcal{L})$ be its associated GKP code. Then:

(i) For any symplectic matrix $S \in \mathsf{Sp}(2,\mathbb{R})$, $\mathsf{GKP}(S\mathcal{L})$ defines a new GKP code and

$$\mathsf{GKP}(S\mathcal{L}) = U_S \mathsf{GKP}(\mathcal{L}) \ . \tag{5.27}$$

(ii) Suppose that the generating vectors of $\mathcal{L}$ satisfy Eq. (5.25). Then there is a symplectic matrix $S \in \mathsf{Sp}(2,\mathbb{R})$ such that

$$\mathsf{GKP}(\mathcal{L}) = U_S \mathsf{GKP}(\mathcal{L}_\square)$$

where $\mathcal{L}_\square$ is the square lattice GKP code introduced in Section 5.1.4.

□

Proof. (i) Suppose that the vectors ℓ_1 and ℓ_2 generate the lattice $\mathcal{L}$. Let us first show that for any $S \in \mathsf{Sp}(2,\mathbb{R})$ the set

$$S\mathcal{L} \coloneqq \{S\ell_1 m_1 + S\ell_2 m_2 \mid m_1, m_2 \in \mathbb{Z}\}$$

is again a symplectically integral lattice according to Definition 5.1. Since $\mathcal{L}$ is a symplectically integral lattice (cf. Definition 5.1), its generating vectors satisfy

$$\ell_i^T J \ell_j \in 2\pi\mathbb{Z} \qquad \text{for } i,j = 1,2 \tag{5.28}$$

which corresponds to Eq. (5.1). We note that Eq. (5.28) is invariant under symplectic transformations since

$$(S\ell_i)^T J (S\ell_j) = \ell_i^T \underbrace{S^T J S}_{=J} \ell_j = \ell_i^T J \ell_j \quad \text{for } i,j = 1,2,\ S \in \mathsf{Sp}(2,\mathbb{R}) \ .$$

Therefore, the vectors $S\ell_1$ and $S\ell_2$ satisfy $(S\ell_i)^T J (S\ell_j) \in 2\pi\mathbb{Z}$ as well and generate a lattice $S\mathcal{L}$ which is symplectically integral. Furthermore, $\mathsf{GKP}(S\mathcal{L})$ encodes the same number of degrees of freedom as $\mathsf{GKP}(\mathcal{L})$: if Eq. (5.25) is satisfied then this is $d = 2$, a qubit.

To prove Eq. (5.27), it is sufficient to show that elements in $U_S\mathsf{GKP}(\mathcal{L})$ are stabilised by elements of the stabiliser group associated with the GKP code with lattice $S\mathcal{L}$. Let $\psi \in \mathsf{GKP}(\mathcal{L})$ and $\xi \in S\mathcal{L}$. Note that then $S^{-1}\xi \in \mathcal{L}$, i.e., $D(S^{-1}\xi) \in \mathcal{S}(\mathcal{L})$. The computation

$$D(\xi) U_S \psi = U_S \underbrace{U_S^\dagger D(\xi) U_S}_{=D(S^{-1})} \psi = U_S \underbrace{D(S^{-1}\xi)\psi}_{=\psi} = U_S \psi \ ,$$

where we used Eq. (2.34) and that $D(S^{-1}\xi) \in \mathcal{S}(\mathcal{L})$, shows that $D(\xi)$ for $\xi \in S\mathcal{L}$ stabilises states of the form $U_S\psi$ where $\psi \in \mathsf{GKP}(\mathcal{L})$. Since GKP codes are stabiliser codes, we have thus shown that $U_S\mathsf{GKP}(\mathcal{L})$ defines a new GKP code and that the lattice associated with this new code (and its stabilisers) is $S\mathcal{L}$.

(ii) By assumption, the two generating vectors ℓ_1, ℓ_2 of $\mathcal{L}$ satisfy $\ell_1^T J \ell_2 = 4\pi$. Define the matrix

$$S \coloneqq \frac{1}{2\sqrt{\pi}} \begin{pmatrix} \ell_1 & \ell_2 \end{pmatrix} .$$

Its columns are the normalised generating vectors of the lattice $\mathcal{L}$. Since it satisfies

$$S^T J S = \frac{\ell_1^T J \ell_2}{4\pi} J = J$$

it is a symplectic matrix. Direct computation shows that

$$S\ell_{\square 1} = \frac{1}{2\sqrt{\pi}} \begin{pmatrix} \ell_1 & \ell_2 \end{pmatrix} \begin{pmatrix} 2\sqrt{\pi} \\ 0 \end{pmatrix} = \ell_1 ,$$
$$S\ell_{\square 2} = \frac{1}{2\sqrt{\pi}} \begin{pmatrix} \ell_1 & \ell_2 \end{pmatrix} \begin{pmatrix} 0 \\ 2\sqrt{\pi} \end{pmatrix} = \ell_2 ,$$

where $\ell_{\square 1}$ and $\ell_{\square 2}$ are the two vectors generating the square lattice $\mathcal{L}_\square$ from Eq. (5.19). Thus the vectors $\ell_1 = S\ell_{\square 1}$ and $\ell_2 = S\ell_{\square 2}$ generate the symplectically integral lattice $\mathcal{L} = S\mathcal{L}_\square$. The first part (i) of this Lemma now implies that $\mathsf{GKP}(\mathcal{L}) = \mathsf{GKP}(S\mathcal{L}_\square) = U_S\mathsf{GKP}(\mathcal{L}_\square)$.

■

Lemma 5.5 implies that symplectic transformations $S \in \mathsf{Sp}(2, \mathbb{R})$ define equivalence classes on the set of GKP codes encoding a single qubit on the single-mode phase space $\mathbb{R}^2$. Furthermore, for GKP codes that satisfy Eq. (5.25), part (ii) shows that there is only a single such equivalence class. More precisely, all GKP-qubit codes on a single bosonic mode can be constructed from the square lattice GKP code $\mathsf{GKP}(\mathcal{L}_\square)$. By Lemma 5.5, the GKP code with lattice $\mathcal{L} = S\mathcal{L}_\square$ for $S \in \mathsf{Sp}(2, \mathbb{R})$ can be obtained from the square lattice GKP code by a unitarily transformed encoding (5.26) of the form

$$\mathcal{E}_{S\mathcal{L}_\square}(\rho) = U_S \mathcal{E}_{\mathcal{L}_\square}(\rho) U_S^\dagger \quad \text{for } \rho \in \mathcal{D}(\mathbb{C}^2) .$$

5.2.2 Asymmetric (rectangular lattice) GKP codes

Of a special interest for the *modified asymmetric surface-GKP code* will be specific lattices $\mathcal{L} = S\mathcal{L}_\square$ and their respective GKP codes: asymmetric, called *rectangular*, and the (asymmetric) hexagonal GKP codes, presented below in this section.

Rectangular lattice GKP codes

Let $r > 0$ and define the two vectors

$$\ell_{r1} \coloneqq \begin{pmatrix} 2\sqrt{\pi r} \\ 0 \end{pmatrix} \quad , \quad \ell_{r2} \coloneqq \begin{pmatrix} 0 \\ 2\sqrt{\pi/r} \end{pmatrix} . \tag{5.29}$$

One can easily see that they generate a symplectically integral lattice.

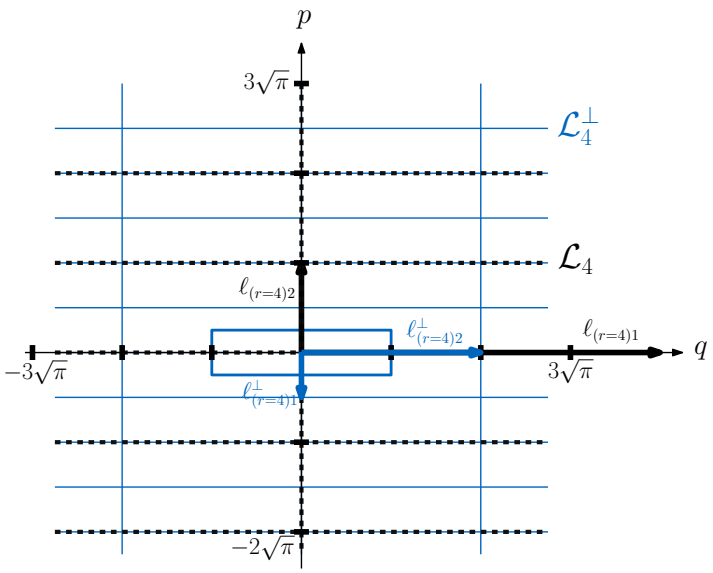


Figure 5.6: Rectangular lattice in phase space for asymmetry ratio $r = 4$: the lattice $\mathcal{L}_4$ (black dashed lines) is spanned by the two vectors $\ell_{(r=4)1}$ and $\ell_{(r=4)2}$ from Eqs. (5.29), the dual lattice $\mathcal{L}_4^\perp$ (blue solid lines) by $\ell^\perp_{(r=4)1}$ and $\ell^\perp_{(r=4)2}$ from Eqs. (5.32) and the Voronoi cell $\mathcal{V}_{\mathcal{L}_4^\perp}$ from Eq. (5.33) is depicted by blue circumference.

Definition 5.6 *(Rectangular GKP-qubit).* The *rectangular lattice GKP-qubit code* or simply the *rectangular GKP code* is defined as $\mathsf{GKP}(\mathcal{L}_r)$ where

$$\mathcal{L}_r := \left\{ \begin{pmatrix} 2\sqrt{\pi r}\, m_1 & 2\sqrt{\pi/r}\, m_2 \end{pmatrix}^T \,\middle|\, m_1, m_2 \in \mathbb{Z} \right\} \tag{5.30}$$

is the symplectically integral lattice generated by ℓ_{r1} and ℓ_{r2} from Eq. (5.29).

We call the parameter $r > 0$ the *asymmetry ratio* associated with the code $\mathsf{GKP}(\mathcal{L}_r)$ since it corresponds to the ratio between the lengths of the two generating vectors ℓ_{1r} and ℓ_{2r}. For $r = 1$, the code $\mathsf{GKP}(\mathcal{L}_1)$ corresponds to the square lattice GKP code. The latter is also called the *symmetric* GKP code, whereas for $r \neq 1$ we call $\mathsf{GKP}(\mathcal{L}_r)$ an *asymmetric* GKP code.

We note that it is straightforward to check that the two vectors from Eq. (5.29) satisfy Eq. (5.1) and thereby actually generate a symplectically integral lattice. These two vectors can be obtained from the square lattice GKP vectors $\ell_{\square 1}$ and $\ell_{\square 2}$ (cf. Eq. (5.19)) by transforming the latter as $\ell_{r1} = S_r \ell_{\square 1}$ and $\ell_{r2} = S_r \ell_{\square 2}$ where the symplectic matrix

$$S_r := \begin{pmatrix} \sqrt{r} & 0 \\ 0 & 1/\sqrt{r} \end{pmatrix} = S_{\mathrm{sq}}(\ln(r)) \tag{5.31}$$

is a single-mode squeezing matrix as considered in Section 2.2.4 (cf. Eq. (2.46) for the symplectic matrix and (2.45) for its unitary). The matrix squeezes the q-coordinate in phase space by the factor $\sqrt{r}$ and the p-coordinate by the inverse factor $1/\sqrt{r}$. Following Lemma 5.5, this squeezing matrix S_r relates the rectangular lattice to the square lattice GKP codes via $S_r \mathcal{L}_\square = \mathcal{L}_r$. We call $\mathsf{GKP}(\mathcal{L}_r)$ the *asymmetric GKP code with (asymmetry) ratio* r. We note that we choose the name asymmetric instead of the (potentially more) intuitive term 'squeezed GKP code' to avoid confusion: in the context of GKP codes, the term 'squeezed'

is commonly used to refer to the finitely squeezed normalisable states in contrast to the non-normalisable infinitely squeezed logical GKP states (cf. Section 5.1.3 and Fig. 5.2).

The dual lattice of $\mathcal{L}_r$ is generated by the two vectors

$$\ell_{r1}^{\perp} = \begin{pmatrix} 0 \\ -\sqrt{\pi/r} \end{pmatrix} \quad , \quad \ell_{r2}^{\perp} = \begin{pmatrix} -\sqrt{\pi r} \\ 0 \end{pmatrix} \ . \tag{5.32}$$

The dual rectangular $\mathcal{L}_r^{\perp}$ lattice and its Voronoi cell are given by the sets

$$\begin{aligned} \mathcal{L}_r^{\perp} &\coloneqq \left\{ \begin{pmatrix} \sqrt{\pi r}\, m_1 & \sqrt{\frac{\pi}{r}}\, m_2 \end{pmatrix}^T \mid m_1, m_2 \in \mathbb{Z} \right\} \ , \\ \mathcal{V}_r^{\perp} = \mathcal{V}_{\mathcal{L}_r^{\perp}} &\coloneqq \left\{ \begin{pmatrix} \sqrt{\pi r}\, \lambda_1 & \sqrt{\frac{\pi}{r}}\, \lambda_2 \end{pmatrix}^T \,\middle|\, \lambda_1, \lambda_2 \in \left[-\tfrac{1}{2}, \tfrac{1}{2}\right] \right\} \ . \end{aligned} \tag{5.33}$$

Fig. 5.6 shows the lattice $\mathcal{L}_4$ as well as its dual $\mathcal{L}_4^{\perp}$. The two basis states of $\mathsf{GKP}(\mathcal{L}_r)$

$$\begin{aligned} |\overline{0}\rangle_r &\coloneqq U_{S_r} |\overline{0}\rangle_{\square} = \sum_{k\in\mathbb{Z}} \delta(q - 2k\sqrt{\pi r}) |q\rangle = \sum_{k\in\mathbb{Z}} |q = 2k\sqrt{\pi r}\rangle \ , \\ |\overline{1}\rangle_r &\coloneqq U_{S_r} |\overline{1}\rangle_{\square} = \sum_{k\in\mathbb{Z}} \delta\left(q - (2k+1)\sqrt{\pi/r}\right) |q\rangle = \sum_{k\in\mathbb{Z}} \left| q = (2k+1)\sqrt{\pi/r} \right\rangle \end{aligned}$$

are computed from the square lattice GKP basis states (5.22) and (5.23). They are the ± 1 eigenstates of the logical GKP-Z operator which we call $\overline{Z}_r$ and corresponds to the phase-space displacement along $-\sqrt{\pi/r}$ in P-direction. Similarly, the logical GKP-X is a phase space Q-shift of $\sqrt{\pi r}$.

Hexagonal GKP code

Another prominent example of a GKP code is the *hexagonal* GKP code $\mathsf{GKP}(\mathcal{L}_{\hexagon})$. For

$$c \coloneqq \left(\frac{2}{\sqrt{3}} \right)^{\frac{1}{2}} \ , \tag{5.34}$$

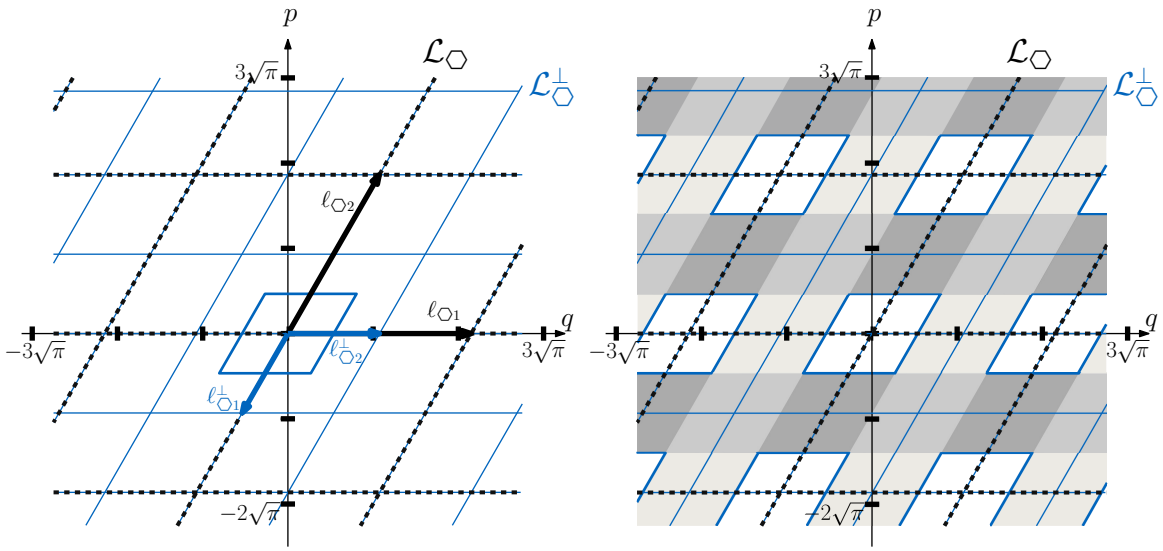
the code $\mathsf{GKP}(\mathcal{L}_{\hexagon})$ is constructed from the hexagonal symplectically integral lattice

$$\mathcal{L}_{\hexagon} \coloneqq \left\{ \sqrt{\pi} c \begin{pmatrix} 2m_1 + m_2 & \sqrt{3}\, m_2 \end{pmatrix}^T \,\middle|\, m_1, m_2 \in \mathbb{Z} \right\} \tag{5.35}$$

which is related to the square lattice via $\mathcal{L}_{\hexagon} = S_{\hexagon} \mathcal{L}_{\square}$, where

$$S_{\hexagon} \coloneqq c \begin{pmatrix} 1 & 1/2 \\ 0 & \sqrt{3}/2 \end{pmatrix}$$

is the associated symplectic matrix. Fig. 5.7 shows the hexagonal lattice, its dual lattice as well as their generating vectors. Note that since the two generating vectors of $\mathcal{L}_{\hexagon}$ have the same length and enclose an angle of $\pi/3$, they form a hexagonal lattice. In the original GKP article [71], Gottesman, Kitaev and Preskill argue that this code can tolerate (slightly) larger shift errors than the square lattice GKP code: both GKP codes enclose an a phase space region of equal size inside their Voronoi cells, but the shortest lattice vector of $\mathcal{L}_{\hexagon}$ has length $2\sqrt{\pi} c$ which is slightly larger than the length of both vectors generating square lattice GKP code $\|\ell_{\square 1}\| = 2\sqrt{\pi}$ since $c \approx 1.07457$.



(a) Hexagonal lattice and its dual

(b) Logical errors of phase space regions

Figure 5.7: Hexagonal GKP code in phase space: (a) the lattice $\mathcal{L}_{\hexagon}$ (black dashed lines) from Eq. (5.35) with its two spanning vectors $\ell_{\hexagon 1}$ and $\ell_{\hexagon 2}$ and similarly the dual lattice $\mathcal{L}_{\hexagon}^{\perp}$ (blue solid lines) with its spanning vectors $\ell_{\hexagon 1}^{\perp}$ and $\ell_{\hexagon 2}^{\perp}$ are depicted. (b) The GKP error correction procedure produces a logical operation depending on the phase space region, the displacement error belongs to (cf. Fig. 5.5a for an explanation of the four differently coloured regions).

Asymmetric hexagonal-lattice GKP codes

One can construct asymmetric hexagonal GKP codes by unitarily encoding the hexagonal code with an additional squeezing for $r > 0$ with lattice

$$\mathcal{L}_{\hexagon,r} \coloneqq S_r S_{\hexagon} \mathcal{L}_{\square} \ . \tag{5.36}$$

Note that $\mathsf{GKP}(\mathcal{L}_{\hexagon,r})$ is a GKP code since the product of two symplectic matrices is again a symplectic matrix. The code $\mathsf{GKP}(\mathcal{L}_{\hexagon,r})$ is called the *asymmetric hexagonal GKP code* and has lattice, dual lattice and Voronoi cell

$$\mathcal{L}_{\hexagon,r} = \left\{ c \begin{pmatrix} 2\sqrt{\pi r}\, m_1 + \sqrt{\pi/r}\, m_2 & \sqrt{3\pi/r}\, m_2 \end{pmatrix}^T \;\middle|\; m_1, m_2 \in \mathbb{Z} \right\} \ , \tag{5.37}$$

$$\mathcal{L}_{\hexagon,r}^{\perp} = \left\{ \frac{c}{2} \begin{pmatrix} 2\sqrt{\pi r}\, m_1 + \sqrt{\pi/r}\, m_2 & \sqrt{3\pi/r}\, m_2 \end{pmatrix}^T \;\middle|\; m_1, m_2 \in \mathbb{Z} \right\} \ , \tag{5.38}$$

$$\mathcal{V}_{\hexagon,r}^{\perp} \coloneqq \left\{ \frac{c}{2} \begin{pmatrix} 2\sqrt{\pi r}\, \lambda_1 + \sqrt{\pi/r}\, \lambda_2 & \sqrt{3\pi/r} \lambda_2 \end{pmatrix}^T \;\middle|\; \lambda_1, \lambda_2 \in \left[-\tfrac{1}{2}, \tfrac{1}{2}\right] \right\} \ , \tag{5.39}$$

respectively, where c is the constant from Eq. (5.34). Fig. 5.8 shows the asymmetric hexagonal lattice as well as its dual for asymmetry ratio $r = 4$.

5.2.3 Symmetric and asymmetric physical noise channels

We consider the performance of GKP codes under probabilistic displacement noise. Recall from Section 3.1.3 that such noise is described by the quantum channel

$$\mathcal{N}_{f_Z}(\rho) = \int_{\mathbb{R}^2} f_Z(\xi) D(\xi) \rho D(\xi)^\dagger d^2\xi \ ,$$

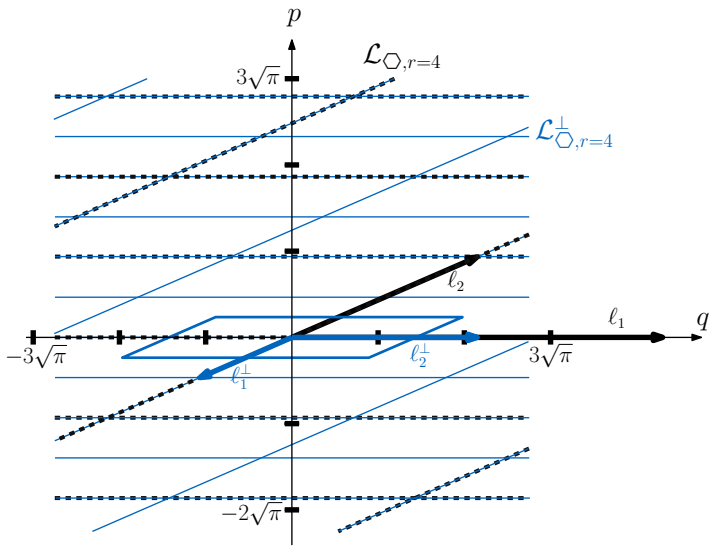


Figure 5.8: Asymmetric hexagonal lattice for $r = 4$ in phase space: the symplectically integral lattice $\mathcal{L}_{\hexagon,r=4}$ (black dashed lines) from Eq. (5.36) is generated by the vectors in black and its dual lattice $\mathcal{L}^{\perp}_{\hexagon,r=4}$ from Eq. (5.38) (blue solid lines) by the vectors in blue. The Voronoi cell of the dual lattice from Eq. (5.39) is depicted by blue circumference around the origin.

for $\rho \in \mathcal{D}(L^2(\mathbb{R}^2))$, where $f_Z : \mathbb{R}^2 \to \mathbb{R}$ is a probability density function associated to the random variable Z. Here we specify $Z \sim \mathsf{N}(0, \Sigma)$ to be centred normal distribution for a positive semidefinite covariance matrix $\Sigma \in \mathbb{R}^{2\times 2}$: the noise is given by

$$\mathcal{N}_\Sigma(\rho) = \int_{\mathbb{R}^2} f_\Sigma(\xi) D(\xi) \rho D(\xi)^\dagger d^2\xi \quad \text{where} \quad f_\Sigma(\xi) = \frac{1}{2\pi\sqrt{\det \Sigma}} e^{-\frac{1}{2}\xi^T \Sigma^{-1} \xi} , \tag{5.40}$$

i.e., $f_\Sigma : \mathbb{R}^2 \to \mathbb{R}$ is a centred normal probability density function in two dimensions.

Let us consider the special case of *symmetric noise* where the covariance matrix Σ is proportional to the identity, i.e., $\Sigma = \sigma^2 I_2$ for some $\sigma > 0$. Then

$$f_{\sigma^2 I}(\xi) = \frac{1}{2\pi\sigma} e^{-\frac{1}{2\sigma^2}\xi^T \xi} = \frac{1}{2\pi\sigma} e^{-\frac{1}{2\sigma^2}(\xi_1^2 + \xi_2^2)} ,$$

is a two-dimensional centred normal distribution which has equal marginal distributions in both variables ξ_1 and ξ_2 and is fully characterised by the variance σ^2. The associated noise channel

$$\mathcal{N}_{\sigma^2 I_2}(\rho) \coloneqq \int_{\mathbb{R}^2} f_{\sigma^2 I}(\xi) D(\xi) \rho D(\xi)^\dagger d^2\xi , \tag{5.41}$$

describes what we call *symmetric displacement noise*. All other noise channels of the form of (5.40) model *asymmetric displacement noise*. Of special interest for the work presented in this chapter are the asymmetric noise channels

$$\mathcal{N}_{\Sigma_r}(\rho) \coloneqq \int_{\mathbb{R}^2} f_{\Sigma_r}(\xi) D(\xi) \rho D(\xi)^\dagger d^2\xi ,$$

where $r > 0$ and $\Sigma_r \coloneqq \sigma^2 (S_{1/r})^2$ for the squeezing matrix S_r from Eq. (5.31).

5.2.4 Logical noise channels

When the GKP code is concatenated with another code (encoding a single logical qubit into many GKP-qubits), its noisy qubits are forwarded to this outer code. For a performance

analysis of the concatenated code, it is essential to compute the error probability of these noisy GKP-qubits, i.e., the probability that after GKP error correction, the GKP-qubit has suffered a logical $\overline{X}$, $\overline{Y}$ or $\overline{Z}$-error. In this section, we derive formulas and results on these logical error probabilities for different GKP codes under probabilistic displacement noise.

Logical noise without conditioning on the GKP syndromes

Consider a valid decoder $c : s \mapsto c(s)$ for the code $\mathsf{GKP}(\mathcal{L})$. Recall the considerations in Section 5.1.2: A single error $D(\nu)$ which causes the syndrome $s(\nu)$ results in the logical operator $\overline{L}$ if condition (5.11) is satisfied. Assuming that the GKP-qubit undergoes displacement noise $\mathcal{N}_\Sigma$ from Eq. (5.40), such a logical operator $\overline{L}$ occurs with probability

$$p_{\overline{L}} \coloneqq \int_{\{\nu\in\mathbb{R}^2 \,|\, c(s(\nu))+\nu\in[\xi_{\overline{L}}]\}} f_\Sigma(\nu) d^2\nu \ , \tag{5.42}$$

where one averages over displacements ν which occur with probability $f_\Sigma(\nu)d^2\nu$. Since we consider GKP codes which encode a single qubit, i.e, for which Eq. (5.25) holds, the logical operations $\overline{L}$ are single-qubit Pauli operators $\overline{I}$, $\overline{X}$, $\overline{Y}$, and $\overline{Z}$.

The *recovery channel*

$$\mathcal{R}_{\mathcal{L},c} : \mathcal{D}(L^2(\mathbb{R})) \to \mathcal{D}(\mathsf{GKP}(\mathcal{L}))$$

is the CPTP map associated with the recovery procedure, i.e., the syndrome measurement of the code $\mathsf{GKP}(\mathcal{L})$ followed by the correction $c : \mathcal{V}_{\mathcal{L}^\perp} \to \mathbb{R}^2$. An initially encoded GKP state $\rho \in \mathcal{D}(\mathsf{GKP}(\mathcal{L}))$ is transformed to

$$\mathcal{R}_{\mathcal{L},c} \circ \mathcal{N}_\Sigma(\rho) = \int_{\mathbb{R}^2} f_\Sigma(\nu) D\big(c(s(\nu))\big) D(\nu) \rho D(\nu)^\dagger D\big(c(s(\nu))\big)^\dagger d^2\nu \ ,$$

by the post-GKP error correction state – after probabilistic displacement noise $\mathcal{N}_\Sigma$ and GKP error correction $\mathcal{R}_{\mathcal{L},c}$. Recall the GKP encoding channel from Eq. (5.4). The combined channel

$$\mathcal{E}_{\mathcal{L}}^\dagger \circ \mathcal{R}_{\mathcal{L},c} \circ \mathcal{N}_\Sigma \circ \mathcal{E}_{\mathcal{L}} \ , \tag{5.43}$$

i.e., the GKP encoding followed by probabilistic displacement noise, the recovery and inverse encoding, acts on the (logical) single-qubit states as a logical single-qubit Pauli channel (cf. Eq. (3.5) from Section 3.1.2) of the form

$$\overline{\mathcal{N}}_{(p_{\overline{I}},p_{\overline{X}},p_{\overline{Y}},p_{\overline{Z}})}(\rho) = p_{\overline{I}}\rho + p_{\overline{X}}\overline{X}\rho\overline{X}^\dagger + p_{\overline{Y}}\overline{Y}\rho\overline{Y}^\dagger + p_{\overline{Z}}\overline{Z}\rho\overline{Z}^\dagger \ . \tag{5.44}$$

Here the error probabilities $p_{\overline{L}}$ are given by Eq. (5.42) and we call the channel from Eq. (5.43) the *logical noise channel* associated with the GKP code $\mathsf{GKP}(\mathcal{L})$, the noise $\mathcal{N}_\Sigma$ and the decoder c.

Logical noise with conditioning on the GKP syndromes

The recovery channel $\mathcal{R}_{\mathcal{L}}$ cannot only be regarded as a quantum channel but also as a quantum instrument (cf. Definition 2.16) where one considers the post-correction quantum state as well as the syndrome outcome. It might be useful to consider the logical noise channel for a given syndrome, i.e., to condition it on a given syndrome outcome $s_0 \in \mathcal{V}_{\mathcal{L}^\perp}$. Assume again that a probabilistic displacement noise $\mathcal{N}_\Sigma$ corrupts an ideal GKP state and that the syndrome measurement results in outcome $s_0 \in \mathcal{V}_{\mathcal{L}^\perp}$. After GKP error correction, we are

interested in the probability of a logical operator $\overline{L}$ under the condition that the syndrome s_0 was measured. Let us denote this probability by $p_{\overline{L}}^{s_0}$ and the probability of a syndrome s_0 by $p_s(s_0)$.

Lemma 5.7. The *logical noise channel conditioned on the syndrome* $s_0 \in \mathcal{V}_{\mathcal{L}^\perp}$ is given by the single-qubit Pauli channel

$$\overline{\mathcal{N}}_{\left(p_{\overline{I}}^{s_0}, p_{\overline{X}}^{s_0}, p_{\overline{Y}}^{s_0}, p_{\overline{Z}}^{s_0}\right)}(\rho) = p_{\overline{I}}^{s_0}\rho + p_{\overline{X}}^{s_0}\overline{X}\rho\overline{X}^\dagger + p_{\overline{Y}}^{s_0}\overline{Y}\rho\overline{Y}^\dagger + p_{\overline{Z}}^{s_0}\overline{Z}\rho\overline{Z}^\dagger \ . \tag{5.45}$$

The respective probabilities are

$$p_{\overline{L}}^{s_0} \coloneqq \frac{\sum_{\xi\in\mathcal{L}} f_\Sigma(\xi_{\overline{L}} + \xi - c(s_0))}{\sum_{\eta\in\mathcal{L}^\perp} f_\Sigma(\eta + s_0)} \tag{5.46}$$

for $L \in \{I, X, Y, Z\}$. □

Proof. By definition, the quantity $p_{\overline{L}}^{s_0}$ is the probability that a displacement $\nu \in \mathbb{R}^2$ which is randomly chosen according to the probability density function f_Σ, satisfies $c(s(\nu)) + \nu \in [\xi_{\overline{L}}]$ under the condition that $s(\nu) = s_0$, i.e., it is given by

$$p_{\overline{L}}^{s_0} \coloneqq \Pr\left[c(s(\nu)) + \nu \in [\xi_{\overline{L}}] \;\middle|\; s(\nu) = s_0\right]$$

where $\Pr[\cdot]$ denotes the probability over $\nu \in \mathbb{R}^2$ randomly chosen according to the distribution f_Σ. By Bayes' rule, this is equal to the quotient

$$p_{\overline{L}}^{s_0} = \frac{\Pr\left[c(s(\nu)) + \nu \in [\xi_{\overline{L}}] \text{ and } s(\nu) = s_0\right]}{\Pr[s(\nu) = s_0]}$$

of the probability that both $c(s(\nu))+\nu \in [\xi_{\overline{L}}]$ and $s(\nu) = s_0$ hold and the probability $p_s(s_0) \coloneqq \Pr[s(\nu) = s_0]$ of the syndrome s_0. Computing the nominator and the denominator yields

$$\begin{aligned} \Pr\left[c(s(\nu)) + \nu \in [\xi_{\overline{L}}] \text{ and } s(\nu) = s_0\right] &\coloneqq \int_{\{\nu\in\mathbb{R}^2 \,|\, c(s_0)+\nu=\xi_{\overline{L}}+\xi,\ \xi\in\mathcal{L}\}} f_\Sigma(\nu)d^2\nu \\ &= \sum_{\xi\in\mathcal{L}} f_\Sigma(\xi_{\overline{L}} + \xi - c(s_0)) \end{aligned} \tag{5.47}$$

$$\begin{aligned} p_s(s_0) &= \int_{\{\nu\in\mathbb{R}^2 \,|\, s(\nu)=s_0\}} f_\Sigma(\nu)d^2\nu \\ &= \sum_{\eta\in\mathcal{L}^\perp} f_\Sigma(\eta + s_0) \ , \end{aligned} \tag{5.48}$$

respectively. Here we used that $s(\nu) = \nu \mod \mathcal{L}^\perp$. ■

Logical noise channels for closest lattice point correction

In the special case of closest lattice point correction $c(s) \coloneqq -s$ associated with the GKP lattice $\mathcal{L}$ (cf. Section 5.1.1), we omit the index c in the recovery channel and simply write $\mathcal{R}_\mathcal{L}$. An error $\nu \in \mathbb{R}^2$ is corrected by a displacement $D(c(s(\nu))) = D(Q_{\mathcal{L}^\perp}(\nu))$. It is useful to consider the Voronoi cell of the dual lattice $\mathcal{L}^\perp$, generated by $\ell_1^\perp$ and $\ell_2^\perp$, i.e.,

$$\mathcal{V}_{\mathcal{L}^\perp} \coloneqq \left\{\lambda_1\ell_1^\perp + \lambda_2\ell_2^\perp \mid \lambda_1, \lambda_2 \in \left[-\tfrac{1}{2}, \tfrac{1}{2}\right)\right\} \ .$$

Note that ignoring the syndrome, as in the previous section, can be interpreted as averaging over the syndrome outcomes.

Lemma 5.8. For the closest lattice point decoder the probabilities from Eq. (5.44) of the logical noise channel are given by

$$p_{\overline{L}} = \sum_{\xi\in\mathcal{L}} \int_{\mathcal{V}_{\mathcal{L}^\perp}} f_\Sigma(\xi_{\overline{L}} + \xi + \nu) d^2\nu \quad \text{for} \quad \overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\} \ . \tag{5.49}$$

Moreover, the probabilities from Eq. (5.45) for the logical noise channel conditioned on the syndrome s_0 are given by

$$p_{\overline{L}}^{s_0} = \frac{\sum_{\xi\in\mathcal{L}} f_\Sigma(\xi_{\overline{L}} + \xi + s_0)}{\sum_{\eta\in\mathcal{L}^\perp} f_\Sigma(\eta + s_0)} \quad \text{for} \quad \overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\} \ . \tag{5.50}$$

□

Proof. For the unconditional probability $p_{\overline{L}}$, let us compute the set of $\nu \in \mathbb{R}^2$ that satisfy condition $Q_{\mathcal{L}^\perp}(\nu) \in [\xi_{\overline{L}}]$. We have

$$\{\nu \in \mathbb{R}^2 \,|\, c(s(\nu)) + \nu \in [\xi_{\overline{L}}]\} = \{\nu \in \mathbb{R}^2 \,|\, \nu = \xi_{\overline{L}} + \xi + w,\ \xi \in \mathcal{L},\ w \in \mathcal{V}_{\mathcal{L}^\perp}\} \ .$$

The probability of a logical operator $\overline{L}$ occurring – averaged over the displacements ν which occur according to f_Σ – is

$$p_{\overline{L}} \coloneqq \int_{\{\nu\in\mathbb{R}^2 \,|\, Q_{\mathcal{L}^\perp}(\nu)\in[\xi_{\overline{L}}]\}} f_\Sigma(\nu) d^2\nu = \sum_{\xi\in\mathcal{L}} \int_{\mathcal{V}^\perp} f_\Sigma(\xi_{\overline{L}} + \xi + \nu) d^2\nu \ .$$

Let us now consider the probabilities for the logical noise channel conditioned on the syndromes, i.e., $p_{\overline{L}}^{s_0}$ from Eq. (5.46). Under the assumption of closest lattice point correction $c(s) = -s$, the numerator of $p_{\overline{L}}^{s_0}$ – cf. Eq. (5.47) – directly translates to

$$\sum_{\xi\in\mathcal{L}} f_\Sigma(\xi_{\overline{L}} + \xi - c(s_0)) = \sum_{\xi\in\mathcal{L}} f_\Sigma(\xi_{\overline{L}} + \xi + s_0)$$

and the denominator stays unchanged, as given in Eq. (5.48). ■

5.2.5 Biasing logical noise by unitarily transformed GKP-encoding

Let us formally prove how to construct biased logical noise from symmetric physical noise by a suitable unitary encoding.

Theorem 5.9 *(Biased noise by encoding).* Let $S \in \mathsf{Sp}(2,\mathbb{R})$ be a symplectic matrix and consider $\mathsf{GKP}(\mathcal{L}_\square)$. Then for $\rho \in \mathcal{D}(\mathsf{GKP}(\mathcal{L}_\square))$ one has

$$\left(\mathcal{R}_{S\mathcal{L}_\square} \circ \mathcal{N}_{\sigma^2 I}\right)(U_S \rho U_S^\dagger) = U_S \left(\mathcal{R}_{\mathcal{L}_\square} \circ \mathcal{N}_{\sigma^2 S^{-1}(S^{-1})^T}(\rho)\right) U_S^\dagger \ . \tag{5.51}$$

where $\mathcal{N}_\Sigma$ denotes the displacement noise channel for $\Sigma \in \mathbb{R}^{2\times 2}$, cf. Eq. (5.40). □

Proof. Let $\rho \in \mathcal{D}(\mathsf{GKP}(\mathcal{L}_\square))$. By Lemma 5.5 (ii), then $U_S \rho U_S^\dagger \in \mathcal{D}(\mathsf{GKP}(S\mathcal{L}_\square))$ defines an encoded state on the asymmetric GKP code with lattice $\mathcal{L} \coloneqq S\mathcal{L}_\square$. Suppose it undergoes

symmetric displacement noise (5.41). Using $U_S^\dagger D(\xi) U_S = D(S^{-1}\xi)$ from Eq. (2.34), the corrupted state is

$$\begin{aligned}\mathcal{N}_{\sigma^2 I}(U_S \rho U_S^\dagger) &= \int_{\mathbb{R}^2} f_{\sigma^2 I}(\xi) D(\xi) \big(U_S \rho U_S^\dagger\big) D(\xi)^\dagger d^2\xi \\ &= \int_{\mathbb{R}^2} f_{\sigma^2 I}(\xi) U_S U_S^\dagger D(\xi) U_S \rho U_S^\dagger D(\xi)^\dagger U_S U_S^\dagger d^2\xi \\ &= U_S \left(\int_{\mathbb{R}^2} f_{\sigma^2 I}(\xi) D(S^{-1}\xi) \rho D(S^{-1}\xi)^\dagger d^2\xi \right) U_S^\dagger \\ &= U_S \left(\int_{\mathbb{R}^2} f_{\sigma^2 I}(S\eta) D(\eta) \rho D(\eta)^\dagger d^2\eta \right) U_S^\dagger \, .\end{aligned} \tag{5.52}$$

The last step is obtained by the substitution $\eta = S^{-1}\xi$ and using the fact that $\det(S) = 1$ since S is symplectic. In terms of the new variable η, the transformed probability density function is

$$f_{\sigma^2 I}(\xi) d^2\xi = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2\sigma^2}(S\eta)^T S\eta} d^2\xi = f_\Sigma(\eta) d^2\eta$$

where $\Sigma^{-1} = S^T S/\sigma^2$ i.e., $\Sigma \coloneqq \sigma^2 S^{-1}(S^{-1})^T$. Inserting this into the corrupted state from Eq. (5.52) gives

$$\begin{aligned}\mathcal{N}_{\sigma^2 I}(U_S \rho U_S^\dagger) &= U_S \left(\int_{\mathbb{R}^2} f_{\sigma^2 S^{-1}(S^{-1})^T}(\eta) D(\eta) \rho D(\eta)^\dagger d^2\eta \right) U_S^\dagger \\ &= U_S \mathcal{N}_{\sigma^2 S^{-1}(S^{-1})^T}(\rho) U_S^\dagger \, .\end{aligned} \tag{5.53}$$

Suppose the GKP closest lattice point error correction procedure associated with the asymmetric code $\mathsf{GKP}(S\mathcal{L}_\square)$ is applied, the recovery channel is $\mathcal{R}_{\mathsf{GKP}(S\mathcal{L}_\square)}$. An error $\nu \in \mathbb{R}^2$ is corrected by the displacement $D\big(c(s(\nu))\big)$ where

$$c(s(\nu)) = -\nu \mod (S\mathcal{L}_\square^\perp) = Q_{S\mathcal{L}_\square^\perp}(\nu) - \nu \, .$$

We again use Eq. (2.34) to compute

$$U_S^\dagger D\big(c(s(S\eta))\big) U_S = D\big(S^{-1}(Q_{S\mathcal{L}_\square^\perp}(S\eta) - S\eta)\big) = D\big(Q_{\mathcal{L}_\square^\perp}(\eta) - \eta\big) \, , \tag{5.54}$$

for $\eta \in \mathbb{R}^2$. Then, the corrupted state from Eq. (5.53) is mapped to the corrected state

$$\begin{aligned}\big(\mathcal{R}_{\mathsf{GKP}(S\mathcal{L}_\square)} \circ \mathcal{N}_{\sigma^2 I}\big)(U_S \rho U_S^\dagger) &= \int_{\mathbb{R}^2} f_{\sigma^2 I}(\xi) D\big(c(s(\xi))\big) D(\xi)\, U_S\, \rho\, U_S^\dagger D(\xi)^\dagger D\big(c(s(\xi))\big)^\dagger d^2\xi \\ &= \int_{\mathbb{R}^2} f_{\sigma^2 S^{-1}(S^{-1})^T}(\eta) U_S \underbrace{U_S^\dagger D\big(c(s(S\eta)\big) U_S}_{\text{Eq. (5.54)}} D(\eta) \rho D(\eta)^\dagger \underbrace{U_S^\dagger D\big(c(s(S\eta)\big)^\dagger U_S}_{\text{Eq. (5.54)}} U_S^\dagger d^2\eta \\ &= U_S \left(\int_{\mathbb{R}^2} f_{\sigma^2 S^{-1}(S^{-1})^T}(\eta) D(\eta - Q_{\mathcal{L}_\square^\perp}(\eta)) D(\eta) \rho D(\eta)^\dagger D\big(Q_{\mathcal{L}_\square^\perp}(\eta) - \eta\big)^\dagger \right) U_S^\dagger \, ,\end{aligned}$$

Note that we can identify $\eta \mapsto \eta - Q_{\mathcal{L}_\square^\perp}(\eta)$ with the correction map $\eta \mapsto c(s(\eta))$ associated with the closest lattice point decoder for the square lattice GKP code $\mathsf{GKP}(\mathcal{L}_\square)$. Hence

$$\big(\mathcal{R}_{S\mathcal{L}_\square} \circ \mathcal{N}_{\sigma^2 I}\big)(U_S \rho U_S^\dagger) = U_S \Big(\mathcal{R}_{\mathcal{L}_\square} \circ \mathcal{N}_{\sigma^2 S^{-1}(S^{-1})^T}(\rho)\Big) U_S^\dagger \, .$$

The inverse unitary encoding $U_S^\dagger(\cdot)U_S$ maps the corrected state on $\mathsf{GKP}(S\mathcal{L}_\square)$ back to the associated logical square lattice GKP state in $\mathsf{GKP}(\mathcal{L}_\square)$. In summary, we have thus shown Eq. (5.51). ■

Theorem 5.9 implies that the post-error-correction states of both codes – $\mathsf{GKP}(S\mathcal{L}_\square)$ under symmetric noise $\mathcal{N}_{\sigma^2 I}$ and $\mathsf{GKP}(\mathcal{L}_\square)$ under noise $\mathcal{N}_{\sigma^2 S^{-1}(S^{-1})^T}$ – are identical. Then of course the resulting logical noise channels of both codes are equal, i.e., in both cases it is given by a probabilistic single-qubit Pauli noise channel $\overline{\mathcal{N}}_{(p_{\overline{I}},p_{\overline{X}},p_{\overline{Y}},p_{\overline{Z}})}$ (cf. Eq. (3.5)), where the respective logical error probabilities for $\overline{X}$-, $\overline{Y}$- and $\overline{Z}$-errors are identical as well.

Rectangular lattice GKP code under symmetric noise

Recall the symplectic matrix S_r from Eq. (5.31) for $r > 0$. It is symmetric and it associated unitary U_{S_r} is an active squeezing Gaussian unitary. By $\mathcal{L}_r \coloneqq S_r\mathcal{L}_\square$ it defines the rectangular lattice GKP code $\mathsf{GKP}(\mathcal{L}_r)$ from Definition 5.6. Hence Theorem 5.9 can be applied to the symplectic matrix S_r from Eq. (5.31).

Corollary 5.10 *(to Theorem 5.9 for $S = S_r$).* For $r > 0$, consider the symplectic matrix S_r from Eq. (5.31). Then one finds that

$$U_{S_r}^\dagger \left(\mathcal{R}_{\mathcal{L}_r} \circ \mathcal{N}_{\sigma^2 I}\right)\left(U_{S_r}\rho U_{S_r}^\dagger\right) U_{S_r} = \left(\mathcal{R}_{\mathcal{L}_\square} \circ \mathcal{N}_{\sigma^2 (S_{1/r})^2}\right)(\rho) \tag{5.55}$$

for all $\rho \in \mathcal{D}(\mathsf{GKP}(\mathcal{L}_\square))$ and $\sigma > 0$. □

We use the covariance matrix

$$\Sigma_r \coloneqq {S_r}^{-1}({S_r}^{-1})^T = \sigma^2 (S_{1/r})^2 = \begin{pmatrix} \sigma^2/r^2 & 0 \\ 0 & \sigma^2 r \end{pmatrix} , \tag{5.56}$$

to write the physical noise channel on the right hand side of Eq. (5.55) as $\mathcal{N}_{\sigma^2 (S_{1/r})^2} = \mathcal{N}_{\Sigma_r}$.

We have thus shown that the logical noise channel $\overline{\mathcal{N}}$ of the rectangular lattice code $\mathsf{GKP}(\mathcal{L}_r)$ (cf. Definition 5.30) under symmetric physical noise $\mathcal{N}_{\sigma^2 I}$ (cf. (5.41)) is equal to the logical noise channel of the square lattice code $\mathsf{GKP}(\mathcal{L}_\square)$ under the asymmetric physical noise channel $\mathcal{N}_{\Sigma_r} = \mathcal{N}_{\sigma^2 (S_{1/r})^2}$. For both of these GKP codes, we obtain the following identical probabilities of the logical noise channel.

Theorem 5.11 *(Symmetric and asymmetric GKP codes).* For $r > 0$, let S_r be the symplectic matrix from Eq. (5.31) and consider the codes $\mathsf{GKP}(\mathcal{L}_\square)$ and $\mathsf{GKP}(\mathcal{L}_r)$ where $\mathcal{L}_r \coloneqq S_r\mathcal{L}_\square$. Let furthermore $\sigma \in \mathbb{R}$ and recall the displacement noise channels $\mathcal{N}_\Sigma$ from Section 3.1.3.

Then the logical error channels of $\mathsf{GKP}(\mathcal{L}_r)$ under symmetric noise $\mathcal{N}_{\sigma^2 I}$ and of $\mathsf{GKP}(\mathcal{L}_\square)$ under asymmetric noise $\mathcal{N}_{\Sigma_r}$ for $\Sigma_r = \sigma^2 (S_{1/r})^2$ are equivalent. More precisely,

$$\mathcal{E}_{\mathcal{L}_r}^\dagger \circ \mathcal{R}_{\mathcal{L}_r} \circ \mathcal{N}_{\sigma^2 I} \circ \mathcal{E}_{\mathcal{L}_r} = \overline{\mathcal{N}}_\pi = \mathcal{E}_{\mathcal{L}_\square}^\dagger \circ \mathcal{R}_{\mathcal{L}_\square} \circ \mathcal{N}_{\Sigma_r} \circ \mathcal{E}_{\mathcal{L}_\square} \tag{5.57}$$

is equal to a qubit noise channel of independent X and Z-noise (cf. Eq. (3.9) from Section 3.1.2). Its associated probabilities $\pi = (p_{\overline{I}}, p_{\overline{X}}, p_{\overline{Y}}, p_{\overline{Z}})$ are given by

$$\begin{array}{rcccl} p_{\overline{I}} &=& (1-q_{\overline{X}}) &\cdot& (1-q_{\overline{Z}}) \\ p_{\overline{X}} &=& q_{\overline{X}} &\cdot& (1-q_{\overline{Z}}) \\ p_{\overline{Y}} &=& q_{\overline{X}} &\cdot& q_{\overline{Z}} \\ p_{\overline{Z}} &=& (1-q_{\overline{X}}) &\cdot& q_{\overline{Z}} \end{array} \tag{5.58}$$

where

$$1 - q_{\overline{X}} \coloneqq \frac{1}{2} \sum_{m \in \mathbb{Z}} \operatorname{erf}\left(\sqrt{\frac{2\pi r}{\sigma^2}}\left(m + \frac{1}{4}\right)\right) - \operatorname{erf}\left(\sqrt{\frac{2\pi r}{\sigma^2}}\left(m - \frac{1}{4}\right)\right) , \tag{5.59}$$

$$1 - q_{\overline{Z}} \coloneqq \frac{1}{2} \sum_{m \in \mathbb{Z}} \operatorname{erf}\left(\sqrt{\frac{2\pi}{\sigma^2 r}}\left(m + \frac{1}{4}\right)\right) - \operatorname{erf}\left(\sqrt{\frac{2\pi}{\sigma^2 r}}\left(m - \frac{1}{4}\right)\right) , \tag{5.60}$$

and where the Gauss error function is defined as

$$\operatorname{erf}(x) \coloneqq \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \qquad \text{for } x \in \mathbb{R} . \tag{5.61}$$

If one considers the rectangular lattice GKP code under symmetric noise, then the logical error channel conditioned on the syndrome $s_0 = \begin{pmatrix} x & z \end{pmatrix}^T \in \mathcal{V}_r^\perp$ is that of a qubit-noise channel of independent X- and Z-noise $\overline{\mathcal{N}}_{\left(p_{\overline{I}}^{s_0}, p_{\overline{X}}^{s_0}, p_{\overline{Y}}^{s_0}, p_{\overline{Z}}^{s_0}\right)}$ with associated probabilities

$$\begin{array}{rcccl} p_{\overline{I}}^{s_0} & = & (1 - q_{\overline{X}}^{s_0}) & \cdot & (1 - q_{\overline{Z}}^{s_0}) \\ p_{\overline{X}}^{s_0} & = & q_{\overline{X}}^{s_0} & \cdot & (1 - q_{\overline{Z}}^{s_0}) \\ p_{\overline{Y}}^{s_0} & = & q_{\overline{X}}^{s_0} & \cdot & q_{\overline{Z}}^{s_0} \\ p_{\overline{Z}}^{s_0} & = & (1 - q_{\overline{X}}^{s_0}) & \cdot & q_{\overline{Z}}^{s_0} \end{array} \tag{5.62}$$

where $q_{\overline{X}}^{s_0}$ and $q_{\overline{Z}}^{s_0}$ are given by

$$1 - q_{\overline{X}}^{s_0} = \frac{\sum_{m_1 \in \mathbb{Z}} e^{-\frac{1}{2\sigma^2}(2\sqrt{\pi r} m_1 + x)^2}}{\sum_{m_2 \in \mathbb{Z}} e^{-\frac{1}{2\sigma^2}(\sqrt{\pi r} m_2 + x)^2}} , \quad 1 - q_{\overline{Z}}^{s_0} = \frac{\sum_{m_1 \in \mathbb{Z}} e^{-\frac{1}{2\sigma^2}(\frac{1}{2}\sqrt{\frac{\pi}{r}} m_1 + z)^2}}{\sum_{m_2 \in \mathbb{Z}} e^{-\frac{1}{2\sigma^2}(\sqrt{\pi / r} m_2 + z)^2}} , \tag{5.63}$$

respectively. □

Proof. Corollary 5.10 to Theorem 5.9 shows that the right hand side of (5.57) is equal to the left hand side.

It is therefore sufficient to compute the error probabilities for the logical noise channel of the right hand side. Consider the logical noise channel without syndrome information for the square lattice GKP code $\mathsf{GKP}(\mathcal{L}_\square)$ under asymmetric noise of the form $\mathcal{N}_{\Sigma_r} = \mathcal{N}_{\sigma^2 (S_{1/r})^2}$.

As shown in Eq. (5.49) – using the closest lattice point decoder – these probabilities are

$$p_{\overline{L}} = \sum_{\xi \in \mathcal{L}_\square} \int_{\mathcal{V}_\square^\perp} f_{\Sigma_r}(\xi_{\overline{L}} + \xi + \nu) d^2\nu \tag{5.64}$$

for $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$. Since Σ_r is diagonal, the probability density function f_{Σ_r} is the product of its marginals, i.e., for $\xi = \begin{pmatrix} \xi_1 & \xi_2 \end{pmatrix}^T$ it can be written as

$$f_{\Sigma_r}(\xi) = g_{\sigma^2/r}(\xi_1) g_{\sigma^2 r}(\xi_2) \quad \text{where} \quad g_\mu(x) \coloneqq \frac{1}{\sqrt{2\pi\mu^2}} e^{-\frac{1}{2\mu^2} x^2} \text{ for } x \in \mathbb{R} . \tag{5.65}$$

The two marginals $g_{\sigma^2/r}$ and $g_{\sigma^2 r}$ are centred normal distributions with variance σ^2/r and $\sigma^2 r$, respectively.

In $\mathsf{GKP}(\mathcal{L}_\square)$, one can write the vector $\xi_{\overline{L}}$ as a function depending on $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$ as

$$\xi_{\overline{L}} = \sqrt{\pi} \begin{pmatrix} 1_{\overline{L}\in\{\overline{X},\overline{Y}\}} \\ 1_{\overline{L}\in\{\overline{Y},\overline{Z}\}} \end{pmatrix} \tag{5.66}$$

where $1_{\overline{L}\in\{\overline{X},\overline{Y}\}}$ and $1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}$ denote the indicator function on the sets $\{\overline{X},\overline{Y}\}$ and $\{\overline{Y},\overline{Z}\}$, respectively. Using the vector (5.66), the error probabilities $p_{\overline{L}}$ from Eq. (5.64) become

$$p_{\overline{L}} = \sum_{\xi\in\mathcal{L}_\square} \int_{\mathcal{V}_\square^\perp} f_{\Sigma_r} \begin{pmatrix} \xi_1 + \nu_1 + 1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\sqrt{\pi} \\ \xi_2 + \nu_2 + 1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\sqrt{\pi} \end{pmatrix} d^2\nu \tag{5.67}$$

Let us insert the function f_{Σ_r} from Eq. (5.65) and the sets $\mathcal{L}_\square$ as well as $\mathcal{V}_\square^\perp$ from Eqs. (5.20) and (5.21), respectively, into Eq. (5.67):

$$\begin{aligned} p_{\overline{L}} &= \left(\sum_{m_1\in\mathbb{Z}} \int_{-\frac{\sqrt{\pi}}{2}}^{\frac{\sqrt{\pi}}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}\left(\sqrt{r}\nu_1 + (2m_1 + 1_{\overline{L}\in\{\overline{X},\overline{Y}\}})\sqrt{\pi r}\right)^2} d\nu_1 \right) \\ &\quad \cdot \left(\sum_{m_2\in\mathbb{Z}} \int_{-\frac{\sqrt{\pi}}{2}}^{\frac{\sqrt{\pi}}{2}} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}\left(\frac{\nu_2}{\sqrt{r}} + (2m_2 + 1_{\overline{L}\in\{\overline{Y},\overline{Z}\}})\sqrt{\frac{\pi}{r}}\right)^2} d\nu_2 \right) \\ &= \left(\sum_{m_1\in\mathbb{Z}} \int_{-\sqrt{\frac{\pi r}{4}}}^{\sqrt{\frac{\pi r}{4}}} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}\left(\nu_1 + (2m_1 + 1_{\overline{L}\in\{\overline{X},\overline{Y}\}})\sqrt{\pi r}\right)^2} d\nu_1 \right) \\ &\quad \cdot \left(\sum_{m_2\in\mathbb{Z}} \int_{-\sqrt{\frac{\pi}{4r}}}^{\sqrt{\frac{\pi}{4r}}} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}\left(\nu_2 + (2m_2 + 1_{\overline{L}\in\{\overline{Y},\overline{Z}\}})\sqrt{\frac{\pi}{r}}\right)^2} d\nu_2 \right) \\ &= \int_{-\sqrt{\frac{\pi r}{4}}}^{\sqrt{\frac{\pi r}{4}}} e_\sigma\left(4r, x + \sqrt{\pi r}\, 1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\right) dx \int_{-\sqrt{\frac{\pi}{4r}}}^{\sqrt{\frac{\pi}{4r}}} e_\sigma\left(\tfrac{r}{4}, z + \sqrt{\tfrac{\pi}{r}}\, 1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\right) dz \ . \end{aligned} \tag{5.68}$$

In the calculation, we used a substitution and defined the function

$$e_\sigma(u,v) \coloneqq \frac{1}{\sqrt{2\pi\sigma^2}} \sum_{m\in\mathbb{Z}} e^{-\frac{1}{2\sigma^2}(m\sqrt{\pi u} + v)^2} \ . \tag{5.69}$$

We note that the integration in the last line of Eq. (5.68) is over the Voronoi cell of the rectangular lattice code $\mathsf{GKP}(\mathcal{L}_r)$ such that

$$p_{\overline{L}} = \int_{\mathcal{V}_r^\perp} e_\sigma\left(4r, \nu_1 + \sqrt{\pi r}\, 1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\right)\ e_\sigma\left(\tfrac{r}{4}, \nu_2 + \sqrt{\tfrac{\pi}{r}}\, 1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\right) d^2\nu \ . \tag{5.70}$$

Let us further compute the two factors on the right hand side of Eq. (5.68). Using again a variable substitution to solve the integrals and the definition of the Gauss error function erf from Eq. (5.61) these factors are

$$\begin{aligned} &\int_{-\sqrt{\frac{\pi r}{4}}}^{\sqrt{\frac{\pi r}{4}}} e_\sigma\big(4r, \nu_1 + 1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\sqrt{\pi r}\big) d\nu_1 = \frac{1}{\sqrt{\pi}} \sum_{k\in\mathbb{Z}} \int_{\sqrt{\frac{2\pi r}{\sigma^2}}\left(k - \frac{1}{4} + \frac{1}{2}1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\right)}^{\sqrt{\frac{2\pi r}{\sigma^2}}\left(k + \frac{1}{4} + \frac{1}{2}1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\right)} e^{-t^2} dt \\ &= \tfrac{1}{2} \sum_{k\in\mathbb{Z}} \operatorname{erf}\left(\sqrt{\tfrac{2\pi r}{\sigma^2}}\big(k + \tfrac{1}{4} + \tfrac{1}{2}1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\big)\right) - \operatorname{erf}\left(\sqrt{\tfrac{2\pi r}{\sigma^2}}\big(k - \tfrac{1}{4} + \tfrac{1}{2}1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\big)\right) , \end{aligned} \tag{5.71}$$

and

$$\begin{aligned}\int_{-\sqrt{\frac{\pi}{4r}}}^{\sqrt{\frac{\pi}{4r}}} e_\sigma\big(\tfrac{r}{4},\nu_2+1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\sqrt{\tfrac{\pi}{r}}\big)d\nu_2 &= \frac{1}{\sqrt{\pi}}\sum_{k\in\mathbb{Z}}\int_{\sqrt{\frac{2\pi}{\sigma^2 r}}\left(k-\frac{1}{4}+\frac{1}{2}1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\right)}^{\sqrt{\frac{2\pi}{\sigma^2 r}}\left(k+\frac{1}{4}+\frac{1}{2}1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\right)} e^{-t^2}dt \\ &= \tfrac{1}{2}\sum_{k\in\mathbb{Z}}\operatorname{erf}\left(\sqrt{\tfrac{2\pi}{\sigma^2 r}}\big(k+\tfrac{1}{4}+\tfrac{1}{2}1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\big)\right)-\operatorname{erf}\left(\sqrt{\tfrac{2\pi}{\sigma^2 r}}\big(k-\tfrac{1}{4}+\tfrac{1}{2}1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\big)\right). \end{aligned} \tag{5.72}$$

These quantities furthermore satisfy

$$\begin{aligned}\int_{-\sqrt{\frac{\pi r}{4}}}^{\sqrt{\frac{\pi r}{4}}} e_\sigma\left(4r,\nu_1+\sqrt{\pi r}\right)d\nu_1 &= 1-\int_{-\sqrt{\frac{\pi r}{4}}}^{\sqrt{\frac{\pi r}{4}}} e_\sigma\left(4r,\nu_1\right)d\nu_1\ , \\ \int_{-\sqrt{\frac{\pi}{4r}}}^{\sqrt{\frac{\pi}{4r}}} e_\sigma\left(\tfrac{4}{r},\nu_2+\sqrt{\tfrac{\pi}{r}}\right)d\nu_2 &= 1-\int_{-\sqrt{\frac{\pi}{4r}}}^{\sqrt{\frac{\pi}{4r}}} e_\sigma\left(\tfrac{4}{r},\nu_2\right)d\nu_1\ .\end{aligned}$$

Inserting the quantities from Eqs. (5.71) and (5.72) into the probabilities $p_{\overline{L}}$ from (5.68) shows that the logical noise channel is given by Eq. (5.58) where

$$q_{\overline{X}} \coloneqq \frac{1}{2}\sum_{m\in\mathbb{Z}}\operatorname{erf}\left(\sqrt{\frac{2\pi r}{\sigma^2}}\left(m+\frac{3}{4}\right)\right)-\operatorname{erf}\left(\sqrt{\frac{2\pi r}{\sigma^2}}\left(m+\frac{1}{4}\right)\right)\ , \tag{5.73}$$

$$q_{\overline{Z}} \coloneqq \frac{1}{2}\sum_{m\in\mathbb{Z}}\operatorname{erf}\left(\sqrt{\frac{2\pi}{\sigma^2 r}}\left(m+\frac{3}{4}\right)\right)-\operatorname{erf}\left(\sqrt{\frac{2\pi}{\sigma^2 r}}\left(m+\frac{1}{4}\right)\right)\ . \tag{5.74}$$

This proves that $1-q_{\overline{X}}$ and $1-q_{\overline{Z}}$ are given by Eq. (5.59) and Eq. (5.60), respectively.

Let us now recall the logical noise channel when conditioned on the syndrome measurement outcome, cf. Eq. (5.45). Denote by $s_0=\begin{pmatrix}x & z\end{pmatrix}^T\in\mathcal{V}_r^\perp$ the syndrome of the rectangular lattice code $\mathsf{GKP}(\mathcal{L}_r)$ under symmetric noise $\mathcal{N}_{\sigma^2 I}$. Then using the general formula from Eq. (5.48) such a syndrome occurs with probability

$$\begin{aligned}p_s(s_0) &= \frac{1}{2\pi\sigma^2}\sum_{m_1,m_2\in\mathbb{Z}} e^{-\frac{1}{2\sigma^2}(\sqrt{\pi r}m_1+x)^2}e^{-\frac{1}{2\sigma^2}(\sqrt{\frac{\pi}{r}}m_2+z)^2} \\ &= e_\sigma(r,x)e_\sigma(1/r,z)\ .\end{aligned} \tag{5.75}$$

Using Eq. (5.50), $p_{\overline{L}}$ and $p_{\overline{L}}^{s_0}p_s(s_0)$ are related to each other via integration over the syndromes $s_0\in\mathcal{V}_r^\perp$. One can directly read off the value of

$$p_{\overline{L}}^{s_0}p_s(s_0) = e_\sigma\left(4r,x+\sqrt{\pi r}\,1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\right)\ e_\sigma\left(\tfrac{4}{r},z+\sqrt{\tfrac{\pi}{r}}\,1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\right)$$

from Eq. (5.70). As a consequence, the probabilities of the logical noise channel are

$$p_{\overline{L}}^{s_0} \coloneqq \frac{e_\sigma\left(4r,x+\sqrt{\pi r}\,1_{\overline{L}\in\{\overline{X},\overline{Y}\}}\right)\ e_\sigma\left(\frac{4}{r},z+\sqrt{\frac{\pi}{r}}\,1_{\overline{L}\in\{\overline{Y},\overline{Z}\}}\right)}{e_\sigma(r,x)e_\sigma(1/r,z)}\ . \tag{5.76}$$

Direct computation shows that for all $x\in\mathbb{R}$ the relations

$$e_\sigma(r,x)=e_\sigma(4r,x+\sqrt{\pi r})+e_\sigma(4r,x)\ ,\ e_\sigma\left(\tfrac{1}{r},z\right)=e_\sigma\big(\tfrac{4}{r},z+\sqrt{\tfrac{\pi}{r}}\big)+e_\sigma\left(\tfrac{4}{r},z\right) \tag{5.77}$$

hold. Combining Eq. (5.76) with Eqs. (5.77) implies that the probabilities (from Eq. (5.62)) satisfy

$$q_{\overline{X}}^{s_0} \coloneqq 1-\frac{e_\sigma\left(4r,x\right)}{e_\sigma(r,x)}\qquad,\qquad q_{\overline{Z}}^{s_0} \coloneqq 1-\frac{e_\sigma\left(4/r,x\right)}{e_\sigma(1/r,x)}\ ,$$

for $s_0 = \begin{pmatrix} x & z \end{pmatrix}^T$. This proves the second part of the claim, i.e., Eq. (5.63) when one inserts the function e_σ from Eq. (5.69).

For the square lattice code $\mathsf{GKP}(\mathcal{L}_\square)$ under asymmetric noise $\mathcal{N}_{\sigma^2(S_{1/r})^2}$, the syndrome $s_0^\square = \begin{pmatrix} x_\square & z_\square \end{pmatrix}^T$ occurs with probability

$$p_s^\square(s_0^\square) \coloneqq \sum_{\eta \in \mathcal{L}_\square^\perp} f_{\Sigma_r}(\eta + s_0^\square) = \frac{1}{2\pi\sigma^2} \sum_{k_1,k_2 \in \mathbb{Z}} e^{-\frac{1}{2\sigma^2}(\sqrt{\pi r}k_1 + \sqrt{r}x_\square)^2} e^{-\frac{1}{2\sigma^2}\left(\sqrt{\frac{\pi}{r}}k_2 + \frac{z_\square}{\sqrt{r}}\right)^2} .$$

This is equal to the probability

$$p_s(s_0) = \sum_{\eta \in \mathcal{L}_r^\perp} f_{\sigma^2 I}(\eta + s_0) = \sum_{\eta \in (S_r\mathcal{L}_\square)^\perp} f_{\sigma^2 I}(\eta + s_0) = \sum_{\eta \in \mathcal{L}_\square^\perp} f_{\sigma^2 I}(S_r\eta + S_r s_0^\square)$$

of the syndrome s_0 in the rectangular lattice code $\mathsf{GKP}(\mathcal{L}_r)$ under symmetric noise $\mathcal{N}_{\sigma^2 I}$ (using Eq. (5.75)) where s_0 and $s_0^\square$ are related via $s_0 = S_r s_0^\square$. We have thus shown that both considered logical error channels also have equal probabilities when conditioned on the syndromes if the respective syndromes are related to each other via the same symplectic matrix S_r that relates the GKP-lattices. ■

Theorem 5.11 shows that the rectangular lattice code $\mathsf{GKP}(\mathcal{L}_r)$ for $r > 0$ under symmetric displacement noise with corresponding covariance matrix $\Sigma_\square = \sigma^2 I_2$ is equivalent to the square GKP code under asymmetric noise with covariance matrix (cf. (5.56))

$$\Sigma_r = \begin{pmatrix} \sigma^2/r^2 & 0 \\ 0 & \sigma r^2 \end{pmatrix} .$$

The term "equivalent" here means that they have the same logical noise channel after GKP error correction. This holds for both cases: first without using the GKP syndrome information and second with conditioning on the syndromes. Fig: 5.9 exemplarily illustrates this result for the case of $r = 4$.

Logical noise channel of the asymmetric hexagonal-lattice GKP code

For the numerical analysis in Section 5.5, we also consider non-rectangular lattices, namely the asymmetric hexagonal lattice lattice $\mathcal{L}_{\hexagon,r}$ from Eq. (5.36). Let us compute the probabilities of the residual logical noise channel of the associated GKP code $\mathsf{GKP}(\mathcal{L}_{\hexagon,r})$.

Lemma 5.12. For $r > 0$, let the asymmetric hexagonal GKP code with lattice $\mathcal{L}_{\hexagon,r}$ from Eq. (5.36) be given as well as the symmetric displacement noise channel $\mathcal{N}_{\sigma^2 I}$ from Eq. (5.41).

Then – without conditioning on the GKP error syndromes – the residual noise channel $\overline{\mathcal{N}}_\pi$ of the GKP-qubit is that of stochastic Pauli noise (cf. Eq. (3.5)) with the associated probabilities $\pi = (p_{\overline{I}}, p_{\overline{X}}, p_{\overline{Y}}, p_{\overline{Z}})$ where

$$\begin{aligned} p_{\overline{L}} \coloneqq & \int_{-\frac{c}{2}\sqrt{\frac{3\pi}{4r}}}^{\frac{c}{2}\sqrt{\frac{3\pi}{4r}}} \int_{-\frac{c}{2}\sqrt{\pi r}+\frac{\nu_2}{\sqrt{3}}}^{\frac{c}{2}\sqrt{\pi r}+\frac{\nu_2}{\sqrt{3}}} \cdot \\ & \cdot e_\sigma^{\hexagon}\left(4r, \nu_1 + c\sqrt{\pi r}1_{\overline{L}\in\{\overline{X},\overline{Y}\}} + \sqrt{\tfrac{c^2\pi}{4r}}1_{\overline{L}\in\{\overline{Z},\overline{Y}\}}, \nu_2 + \sqrt{\tfrac{3c^2\pi}{4r}}1_{\overline{L}\in\{\overline{Z},\overline{Y}\}}\right) d\nu_1 d\nu_2 \end{aligned} \tag{5.78}$$

for $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$ and $c^2 \coloneqq 2/\sqrt{3}$ from Eq. (5.34). Here, for $\sigma, u > 0$ and $x, z \in \mathbb{R}$, we

defined

$$e_\sigma^{\hexagon}(u,x,z) \coloneqq \frac{1}{2\pi\sigma^2} \sum_{m_1,m_2\in\mathbb{Z}} e^{-\frac{1}{2\sigma^2}\left[\left(c\sqrt{\pi u}m_1+\frac{c}{2}\sqrt{\frac{\pi}{u}}m_2+x\right)^2+\left(\frac{c}{2}\sqrt{\frac{3\pi}{u}}m_2+z\right)^2\right]} . \tag{5.79}$$

When conditioning on a syndrome outcome $s_0 = \begin{pmatrix} x & z \end{pmatrix}^T$, the logical noise channel $\overline{\mathcal{N}}_\pi$ is that of stochastic Pauli noise, cf. Eq. (5.62), where $\pi = (p_{\overline{I}}^{s_0}, p_{\overline{X}}^{s_0}, p_{\overline{Y}}^{s_0}, p_{\overline{Z}}^{s_0})$ with probabilities

$$p_{\overline{L}}^{s_0} \coloneqq \frac{e_\sigma^{\hexagon}\left(4r, \nu_1 + c\sqrt{\pi r}1_{\overline{L}\in\{\overline{X},\overline{Y}\}} + \sqrt{\frac{c^2\pi}{4r}}1_{\overline{L}\in\{\overline{Z},\overline{Y}\}}, \nu_2 + \sqrt{\frac{3c^2\pi}{4r}}1_{\overline{L}\in\{\overline{Z},\overline{Y}\}}\right)}{e_\sigma^{\hexagon}(r,x,z)} \tag{5.80}$$

for $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$ where again $e_\sigma^{\hexagon}$ as defined in Eq. (5.79). □

Proof. Similar to the proof of Theorem 5.11, we will use the general formulas for the probabilities $p_{\overline{L}}$ and $p_{\overline{L}}^{s_0}$ from Eqs. (5.49) and (5.50), respectively, and insert the vectors

$$\xi_{\overline{L}} = \begin{pmatrix} 2\sqrt{\pi r}1_{\overline{L}\in\{\overline{X},\overline{Y}\}} + \sqrt{\frac{\pi}{r}}1_{\overline{L}\in\{\overline{Z},\overline{Y}\}} \\ \sqrt{\frac{3\pi}{r}}1_{\overline{L}\in\{\overline{X},\overline{Z}\}} \end{pmatrix}$$

for $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$ of the asymmetric hexagonal GKP code. First, without conditioning on the syndromes, the error probabilities from Eq. (5.49) become

$$\begin{aligned} p_{\overline{L}} &= \int_{\mathcal{V}_{\hexagon,r}^\perp} \sum_{\xi\in\mathcal{L}_{\hexagon,r}} f_{\sigma^2 I}(\nu+\xi+\xi_{\overline{L}})d^2\nu \\ &= \frac{1}{2\pi\sigma^2}\int_{\mathcal{V}_{\hexagon,r}^\perp} \sum_{m_1,m_2} e^{-\frac{1}{2\sigma^2}\left[c\sqrt{\pi r}(2m_1+1_{\overline{L}\in\{\overline{X},\overline{Y}\}})+c\sqrt{\pi/r}(2m_2+1_{\overline{L}\in\{\overline{Z},\overline{Y}\}})+\nu_1\right]^2} \\ &\qquad\qquad \cdot e^{-\frac{1}{2\sigma^2}\left[c\sqrt{3\pi/(2r)}\left(2m_2+1_{\overline{L}\in\{\overline{Z},\overline{Y}\}}\right)+\nu_2\right]^2} d^2\nu \\ &= \int_{\mathcal{V}_{\hexagon,r}^\perp} e_\sigma^{\hexagon}\left(4r, \nu_1 + c\sqrt{\pi r}1_{\overline{L}\in\{\overline{X},\overline{Y}\}} + \sqrt{\frac{c^2\pi}{4r}}1_{\overline{L}\in\{\overline{Z},\overline{Y}\}}, \nu_2 + \sqrt{\frac{3c^2\pi}{4r}}1_{\overline{L}\in\{\overline{Z},\overline{Y}\}}\right)d^2\nu \end{aligned}$$

where the quantity $e_\sigma^{\hexagon}$ is defined in Eq. (5.79). If one inserts the definition of the dual Voronoi cell $\mathcal{V}_{\hexagon,r}^\perp$ from Eq. (5.39), this corresponds to the desired Eq. (5.78).

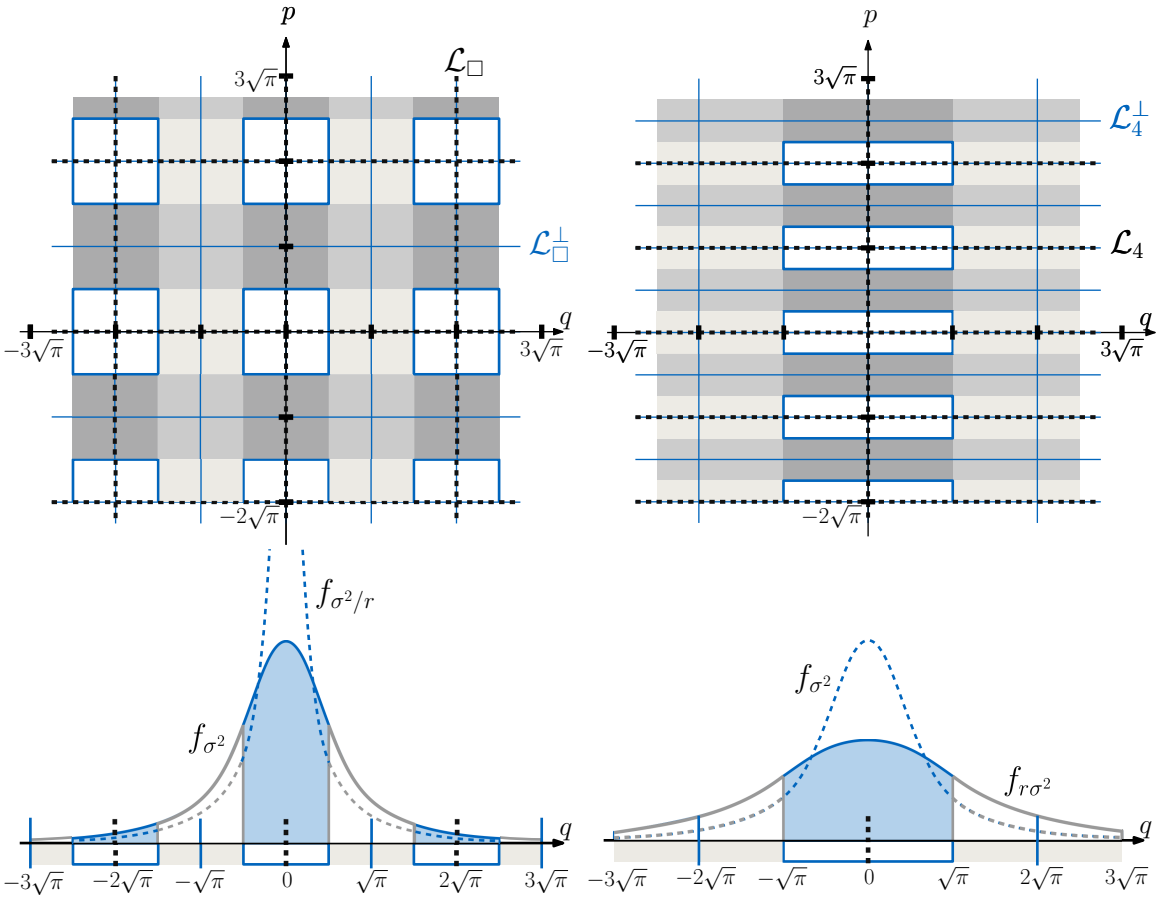
Similarly, one may directly compute the probability $p_s(s_0)$ of a syndrome $s_0 \coloneqq \begin{pmatrix} x & z \end{pmatrix}^T$ from Eq. (5.48) to be

$$p_s(s_0) \coloneqq \sum_{\eta\in\mathcal{L}_{\hexagon,r}^\perp} f_{\sigma^2 I}(s_0+\eta) = e_\sigma^{\hexagon}(r,x,z) . \tag{5.81}$$

In order to compute the residual error probabilities $p_{\overline{L}}^{s_0}$ from Eq. (5.50) – when conditioning on the syndrome s_0, one may use the relation

$$p_{\overline{L}}^{s_0} = (p_s(s_0))^{-1} \sum_{\xi\in\mathcal{L}_{\hexagon,r}} f_{\sigma^2 I}(s_0+\xi+\xi_{\overline{L}}) .$$

The nominator is given by the above computed term for $p_{\overline{L}}$ (without the integration over the dual Voronoi cell) and the denominator by Eq. (5.81), such that combining both terms yields the desired Eq. (5.80). ■



(a) Ratio $r = 1$ (b) Ratio $r = 4$

Figure 5.9: Comparison of the square lattice GKP code $\mathsf{GKP}(\mathcal{L}_\square) = \mathsf{GKP}(\mathcal{L}_1)$ and the rectangular lattice GKP code $\mathsf{GKP}(\mathcal{L}_r)$ for asymmetry ratio $r = 4$ on the left (a) and right (b), respectively. The two top figures depict the defining symplectically integral lattices (in dashed black lines) and their duals (blue solid lines) as well as the phase space regions associated with the logical operations I, X, Y and Z (cf. Fig 5.5). For rectangular lattices GKP codes under symmetric noise $\mathcal{N}_{\sigma^2 I}$ from Eq. (5.41), the GKP-error correction can be conducted in the Q-direction and the P-direction separately. The two bottom figures depict error distribution and correction in the Q-direction for the two different asymmetries ratios $r = 1$ and $r = 4$, respectively: the Gaussian probability density function $f_{\sigma^2}(q) = (2\pi\sigma)^{-1/2} e^{-q^2/(2\sigma^2)}$ is the marginal in q-direction of the symmetric noise probability density function. The light blue shaded area – the integration of $f_{\sigma^2}(q)$ over the set $\bigcup_{m\in\mathbb{Z}} \left[\frac{\sqrt{r\pi}}{2}(4m-1), \frac{\sqrt{r\pi}}{2}(4m+1)\right)$ – corresponds to the probability that the q-direction of a randomly chosen shift $\nu = (q\ p)^T$ lies in the blue edged phase space regions $\mathcal{V}_r^\perp + \mathcal{L}_r$ of the phase space. It gives the probability of the closest-lattice point correction in the q-direction to be successful, i.e., the value $1 - q_{\overline{X}}$ from Eq. (5.59) for the two different ratios $r = 1$ and $r = 4$. Theorem 5.11 implies that the value $1 - q_{\overline{X}}$ for $r = 1$ and function f_{σ^2} (in (a)) is equal to $1 - q_{\overline{X}}$ for $r = 4$ and the squeezed function $f_{r\sigma^2}$ (in (b)); vice versa the integrals under the dotted cures also coincide: $1 - q_{\overline{X}}$ for $r = 1$ and the (inversely) squeezed function $f_{\sigma^2/r}$ (dotted line in (a)) corresponds to $1 - q_{\overline{X}}$ for $r = 4$ and f_{σ^2} (dotted line in (b)).

Note that in contrast to the rectangular lattice code under symmetric noise, the (asymmetric) hexagonal-lattice GKP code under symmetric noise results in a residual noise channel which is no longer given by independent X and Z-noise.

Squeezing for asymmetric GKP codes

As discussed above, the rectangular lattice code $\mathsf{GKP}(\mathcal{L}_r)$ is obtained from the square GKP code by a unitary Gaussian encoding U_{S_r}. The unitary U_{S_r} is associated with the symplectic matrix S_r from Eq. (5.31), a single-mode squeezing matrix. An operationally relevant task is to quantify the amount of squeezing which is necessary to produce a GKP state with asymmetry ratio $r \neq 1$. Let us introduce the *squeezing factor* s_Q of the variance in the Q-direction (we choose the Q-direction without loss of generality; in the P-direction, the squeezing factor can be defined similarly). In units of decibel, it is defined as

$$s_Q \coloneqq -10 \log_{10} \left(\sigma_Q^2 / \sigma_0^2 \right) \ .$$

Here, $\sigma_0^2 = 1/2$ is the variance in Q-direction associated with the vacuum state and the quantity σ_Q^2 stands for the Q-variance in the state under consideration. Since the basis states of the square lattice GKP code require infinite squeezing, one may instead quantify the amount of squeezing necessary to produce the basis states of $\mathsf{GKP}(\mathcal{L}_r)$ from the basis states of $\mathsf{GKP}(\mathcal{L}_\square)$: After the unitary encoding using U_{S_r}, the covariance matrix $V_\square$ of a $\mathsf{GKP}(\mathcal{L}_\square)$-state it is transformed as

$$S_r V_\square S_r^T = \begin{pmatrix} \sqrt{r} & 0 \\ 0 & 1/\sqrt{r} \end{pmatrix} \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{pmatrix} \begin{pmatrix} \sqrt{r} & 0 \\ 0 & 1/\sqrt{r} \end{pmatrix} = \begin{pmatrix} r\sigma^2 & 0 \\ 0 & \sigma^2/r \end{pmatrix}$$

for the matrix S_r from Eq. (5.31). Hence $\sigma_Q \mapsto r^2 \sigma_Q$. In comparison to the square GKP code (for $r = 1$), the squeezing factor in Q-direction (or in P-direction) under application of U_{S_r}

$$s_Q \mapsto s_Q - 10 \log_{10}(r)$$

is increased by the addition of the term $-10 \log_{10}(r)$ (respectively $+10 \log_{10}(r)$ in P-direction) which grows logarithmically with the asymmetry ratio r. For example, for $r = 2$, $r = 3$ and $r = 4$, this additional squeezing amounts to -3.01 dB, -4.77 dB and -6.02 dB, respectively.

5.3 Prior work on the surface(-GKP) code relevant to our analysis

The GKP-qubit (may it be encoded using the square GKP code or a more general encoding) can be used as the "physical" qubit for another code. One may consider the concatenation of the GKP code with a qubit-into-qubits QECC and analyse whether this concatenation improves the noise tolerance of the encoded qubit compared to the bare QECC. Here, we consider the concatenation of the GKP code with one of the most prominent qubit-into-qubits encodings, the surface code [30, 47].

In this section, let us discuss prior work which is directly relevant to our analysis of the surface-GKP code. The first (Section 5.3.1) concerns no-go results on Gaussian quantum error correction. We subsequently present the surface-GKP code and associated results in

Section 5.3.2. Furthermore, we describe the so-called BSV decoder [29] for the surface code in Section 5.3.4 which we employ in our numerical analysis of the asymmetric surface-GKP code and previous work on noise bias in the surface code in Section 5.3.4. The latter results motivate our idea to engineer a noise bias at the level of the logical GKP-qubits via an asymmetric GKP encoding.

5.3.1 Gaussian error correction and corresponding no-go results

Gaussian operations – that includes evolution under quadratic bosonic Hamiltonians (i.e., Gaussian unitary operations) and dilations thereof, the preparation of the vacuum, coherent (or more generally Gaussian) states, and homodyne or heterodyne measurements – are attractive from an experimental viewpoint as their realisation is feasible with standard quantum optics equipment. Unfortunately, information-processing capabilities of such operations are fundamentally constrained. Several no-go results have been established in the literature for error correction and related tasks. For example, Eisert et al. [56], Fiurasek et al. [64], as well as Giedke and Cirac [69] show that Gaussian states cannot be distilled by local Gaussian operations and classical communication. This is in contrast to the finite-dimensional case, where e.g., entanglement distillation protocols using Clifford operations only exist. Niset, Fiurášek and Cerf [123] show that Gaussian pre- and post-processing operations cannot be used to reduce entanglement degradation even when the noise is given by a Gaussian channel.

In more recent work [169], it was shown that embedding logical CV modes into a larger number of physical modes by means of Gaussian encoding unitaries (i.e., Gaussian CV-to-CV-encodings) does not reduce the amount of effective noise (when going from physical to logical information) in the presence of i.i.d. Gaussian phase space displacement noise. More precisely, it was shown that if the physical noise is Gaussian displacement noise with variance σ^2 on each mode, then the noise at the logical level (affecting k encoded modes) can be understood as Gaussian displacement noise in the Q_j and P_j-directions with variances satisfying $\sigma^2_{Q_j}\sigma^2_{P_j} = \sigma^2$ for $j = 1, \ldots, k$. This is with respect to certain rotated quadratures $\{(Q_j, P_j)\}_{j=1}^k$. This result can be seen as a kind of uncertainty relation. It implies that the noise cannot be eliminated but only squeezed in certain directions, generating a trade-off between pairs of canonical quadratures.

These no-go results show that non-Gaussian resources are a necessary building block for CV-fault-tolerant information processing, especially in the context of quantum memories and communication. We note that the case of dynamical decoupling considered in Chapter 4 does not fall into this framework because it is assumed that pulses can be applied at any time, thereby effectively changing the dynamics of the noise. Suitable non-Gaussian resources can include non-Gaussian resource states, non-Gaussian unitary evolution (e.g., under Hamiltonians that are higher-degree polynomials in the quadratures), or e.g., number state measurements (single photon detection).

Noh, Girvin and Jiang have proposed non-Gaussian encodings of several modes (so-called oscillator-to-oscillator codes) which use GKP states in auxiliary systems combined with Gaussian unitaries. It was shown that these constructions indeed lead to reduced physical noise variances [125], although limitations of this approach were subsequently found in [77]. A different approach considers DV-to-CV encodings involving GKP states as a non-Gaussian resource. The surface-GKP discussed in the next section is an example.

5.3.2 The surface-GKP code

The surface-GKP code and the related toric-GKP-code are natural examples of codes embedding individual qubits into many oscillators in such a way that the code states are non-Gaussian states. These codes are defined as the concatenation of the surface or the toric code, respectively, at the outer level and the GKP code at the inner level (cf. Section 3.4.4 on the concatenation of QECC and Section 3.4.5 on the surface code). Because surface codes constitute an infinite family of codes parametrised by systems size – i.e., number of qubits, here denoted by $n \in \mathbb{N}$ – and these codes are known to exhibit a fault-tolerance threshold under probabilistic Pauli noise, surface-GKP codes are candidates for the robust storage of logical qubits in many oscillators. Furthermore, because they involve non-Gaussian building blocks (in the form of GKP states), they sidestep the no-go results discussed in the previous section.

To study error thresholds of this proposal, suitable decoders have been considered. As a concatenated code, the surface-GKP permits a hierarchic decoding procedure where a recovery is applied to each GKP-qubit first, followed by a surface code decoder which may or may not use syndrome information obtained in the GKP recovery. This approach was taken by Fukui, Tomita, Okamoto and Fujii to estimate thresholds against symmetric Gaussian displacement noise. These authors use the standard GKP recovery procedure (see Section 5.1.2) and the minimum-weight matching decoder for the surface code decoding. In the case where GKP syndrome information is taken into account, an edge weight is suitably chosen for each GKP-qubit (i.e., surface code lattice), see [68, Eq. (10)]. Numerical estimates are obtained by using Monte-Carlo simulations. Reference [68] also considers the realisation of logical gates and includes a proposal for fault-tolerant measurement-based computation using a CV version of the 3D cluster state.

In [169], the toric-GKP code was considered by Vuillot et al. Again, numerical results were obtained using a concatenated decoder for several settings, including decoding with or without GKP syndrome information, decoding with and without measurement errors (with repeated measurements) both at the level of individual GKP-qubits and at the level of toric code syndrome measurements. For the case of ideal measurements, the decoders chosen are similar to those of [68] albeit a different weighting is chosen for the minimum weight matching [169, see Eq. (33)].

The work [169] also analytically studies the decoding problems in the context of measurements errors in detail. For a single GKP-encoded qubit, they develop a new recovery procedure the authors call "forward-minimisation" applicable to the case where both the physical mode, as well as the measurement outcome are affected by random Gaussian displacements. Their analysis connects maximum-likelihood decoding to a 1D Euclidean path-integral modelling a particle in a random cosine potential. This is closely related to the connection between logical error probabilities in the surface code and the partition function of a 2D random-bond Ising model [47]. The recovery procedure can be seen as a proxy for the evaluation of this path integral.

For toric-GKP codes with noisy syndrome measurements, the work [169] connects the maximum likelihood decoding problem to a 3D compact QED model in the presence of a quenched random gauge field. Again, a new associated decoder is constructed and numerical threshold estimates are obtained.

Both constructions of [169] and [68] rely on the square lattice-GKP code encoding a single qubit, cf. Fig. 5.10 which shows the quantum circuit realising a surface-GKP encoding such as

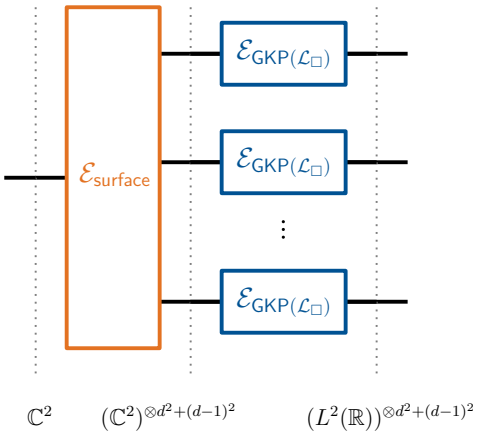


Figure 5.10: Encoding circuit of the surface-GKP code where one considers a surface code (encoding one qubit into n physical) at the inner and the square lattice GKP code $\mathsf{GKP}(\mathcal{L}_\square)$ at the outer level. As a consequence, the encoding channel is given by $\mathcal{E}_{\mathsf{surface-GKP}} \coloneqq (\mathcal{E}_{\mathcal{L}_\square})^{\otimes n} \circ \mathcal{E}_{\mathsf{surface}}$.

the one considered in [68]. In this thesis, we show that simply choosing a different GKP code provides operational advantages. However, this modification also necessitates using different decoders: instead of the minimum-weight matching decoders and the decoders considered in [169, 68], we use standard GKP-qubit recovery combined with the so-called BSV-decoder. This decoder is introduced in the next section.

5.3.3 The BSV decoder for the surface code

Bravyi, Suchara and Vargo [29] introduced a decoder for the surface code that is efficiently computable and approximates the maximum likelihood (ML) decoder. It depends on an integer parameter χ and becomes exact – i.e., equal to the ML decoder – in the limit $\chi \to \infty$. The key improvement of this decoder compared to the ML decoder is its efficiency. It approximates the coset probabilities (cf. Eqs. (3.51) in Section 3.4.5) by identifying a tensor network contraction with their computation.

Since the constructed asymmetric surface-GKP codes rely a distance d square surface code with rough top/bottom and smooth left/right boundary conditions (cf. Section 3.4.5), we focus on the BSV decoder for this case, here.

Assume independent Pauli noise on all qubits, i.e., $\mathcal{N}_\pi = \bigotimes_{e\in\mathsf{E}} \mathcal{N}_{\pi_e}$ where

$$\mathcal{N}_{\pi_e} : \mathcal{D}\left(\mathbb{C}^2\right) \to \mathcal{D}\left(\mathbb{C}^2\right) \quad , \quad \mathcal{N}_{\pi_e}(\rho) = \sum_{E_e\in\mathcal{P}_1} \pi_e(E_e) E_e \rho E_e^\dagger \, . \tag{5.82}$$

On every physical qubit $e \in \mathsf{E}$, the distribution of errors $E_e \in \mathcal{P}_1$ is given by $\pi_e : \mathcal{P}_1 \to [0,1]$ and we write $E = \bigotimes_{e\in\mathsf{E}} E_e$. The probability of the coset $E\mathcal{S}$ (where E is the error and $\mathcal{S}$ the stabiliser group) is

$$\pi(E\mathcal{S}) = \sum_{S\in\mathcal{S}} \prod_{e\in\mathsf{E}} \pi_e(E_e S_e) \tag{5.83}$$

where we similarly decompose a stabiliser as $S = \bigotimes_{e\in\mathsf{E}} S_e$.

The authors of [29] show that the coset probabilities $\pi(E\mathcal{S})$ can be written as the contraction of a two-dimensional tensor network.

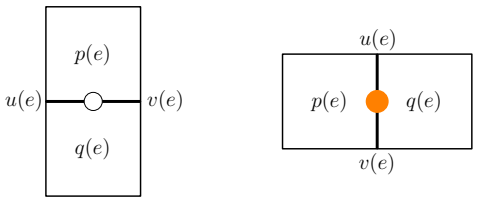


Figure 5.11: A horizontal edge e (white dot) has left/right adjacent vertices $u(e)$ and $v(e)$, respectively, and adjacent top/bottom plaquettes $p(e)$ and $q(e)$; similarly define for a vertical edge e (orange dot), the adjacent top/bottom vertices as $u(e)$ and $v(e)$ and the adjacent left/right plaquettes as $p(e)$ and $q(e)$.

More precisely, they parametrise a stabiliser $S \in \mathcal{S}$ as

$$S(\alpha, \beta) = \prod_{v \in \mathsf{V}} (A_v)^{\alpha_v} \prod_{p \in \mathsf{P}} (B_p)^{\beta_p}$$

for binary parameters $\alpha, \beta \in \{0,1\}^{d(d-1)}$. The restriction of $S(\alpha, \beta)$ to an edge $e \in \mathsf{E}$ then is

$$S_e(\alpha, \beta) = S(\alpha_{u(e)}, \alpha_{v(e)}, \beta_{p(e)}, \beta_{q(e)}) \ ,$$

see Fig. 5.11 for definitions of $u(e)$, $v(e)$, $p(e)$, $q(e)$. They rewrite Eq. (5.83) as

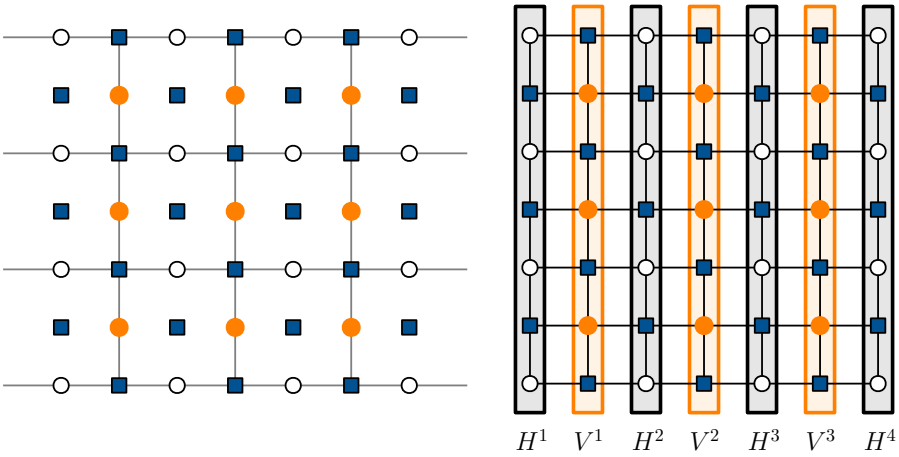
$$\pi(E\mathcal{S}) = \sum_{\alpha} \sum_{\beta} \prod_{e} \pi_e(E_e S_e(\alpha_{u(e)}, \alpha_{v(e)}, \beta_{p(e)}, \beta_{q(e)})) \ . \tag{5.84}$$

Bravyi, Suchara and Vargo show that the right hand side of Eq. (5.84) is a properly defined contraction of a 2D planar tensor network where the local tensors are given as in Fig. 5.12.

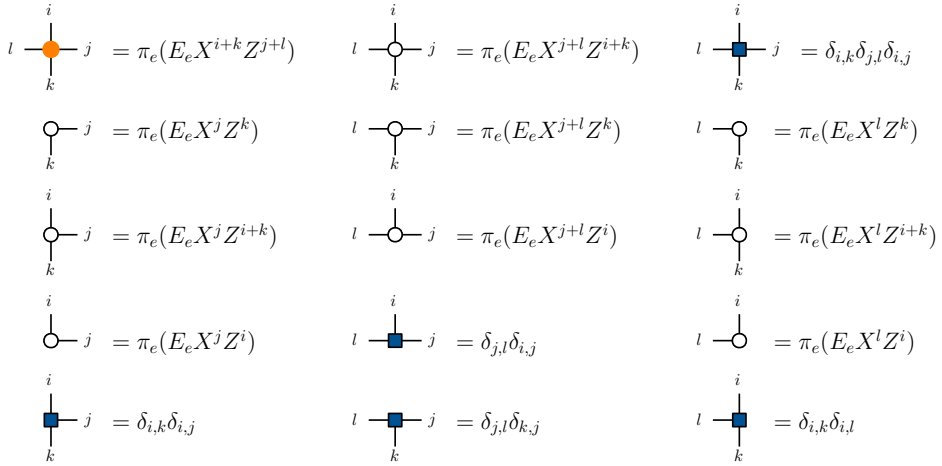
Such a tensor network contraction can be efficiently evaluated approximately: In order to do so, one partitions the extended lattice into columns H_1, ..., H_d and V_1, ..., V_{d-1} that contain the horizontal and vertical edges, respectively, cf. Fig. 5.12b. Here, the 'internal' columns $H^2, \ldots, H^{d-1} \in \mathcal{B}((\mathbb{C}^2)^{\otimes(2d-1)})$ and $V^1, \ldots, V^{d-1} \in \mathcal{B}((\mathbb{C}^2)^{\otimes(2d-1)})$ are matrix product operators (MPO) of bond dimension 2 and the first/last column $H^d, H^1 \in (\mathbb{C}^2)^{\otimes(2d-1)}$ are matrix product states (MPS) of bond dimension 2. The horizontal (vertical) links of the extended lattice (Fig. 5.12) correspond to physical (virtual) indices of the MPOs/MPSs. Then the coset probability (5.84) is

$$\pi(E\mathcal{S}) = \langle H^d | V^{d-1} H^{d-1} V^d \cdots H^2 V^1 | H^1 \rangle \ . \tag{5.85}$$

The tensor network contraction on the right hand side of Eq. (5.85) is evaluated approximately using an algorithm by Murg, Verstraete and Cirac [118]: The algorithm initialises $H^1 = \psi \in (\mathbb{C}^2)^{\otimes 2d-1}$ – an MPS of bond dimension 2 – and updates the MPS ψ at every step by MPO multiplication $V^j\psi$ or $H^j\psi$ (for even and odd steps, respectively). Since this would imply an exponential increase of the MPS bond dimension (in the number of steps $2d-1$) – leading to an overall inefficient algorithm – the parameter χ comes into play and one limits the bond dimension of ψ: At every step of the algorithm, when the update rule $\psi \mapsto \phi := V^j\psi$ (or $\psi \mapsto \phi := H^j\psi$) would (generally) increase the bond dimension of ψ from χ to up to 2χ, an additional 'truncation' is introduced to find an MPS ψ of bond dimension χ which approximates ϕ. In this sense, at every step, the algorithm retains an MPS of bond dimension χ. We refer to [29, Chapter IV.C] for the exact description of the truncation step. At a last step, a standard contraction method is used to compute the inner product between the two MPS ψ and H^d. The accuracy of the algorithm in approximating the right hand side of Eq. (5.85) depends on the parameter χ, becoming exact if χ is exponentially large in d.



(a) Extended lattice (b) Tensor network for the BSV decoder



(c) Local tensors; bulk tensors in the first and boundary tensors in the other lines

Figure 5.12: Construction of the tensor network to compute the coset probabilities for the BSV decoder in the 4×4 surface code: (a) First, one extends the surface code lattice such that it has three different types of nodes: white and orange nodes at the locations of horizontal edges and the vertical edges, respectively, of the original surface code lattice and blue squares at locations of stabiliser generators A_v and B_p (vertices and plaquettes of the original surface code lattice). (b) This extended lattice can be interpreted as a two-dimensional tensor network whose contraction yield the probabilities $\pi(E_s\overline{L}\mathcal{S})$ for the BSV decoder where E_s is a representative error for the syndrome s and $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$ is a logical operator. Local tensors are grouped into d columns $H_1, \ldots, H_d$ and $d-1$ columns $V_1, \ldots, V_{d-1}$, associated with the horizontal and vertical nodes, respectively. (c) The three different types of local bulk tensors in the first line where $E_s\overline{L} = \bigotimes_{e\in\mathsf{E}} E_e$ for qubits $e \in \mathsf{E}$ and $S_e(j,l,i,k) = X^{i+k}Z^{j+l}$; below, the similarly defined local tensors at the boundary of the surface code.

Let us briefly discuss the runtime of this algorithm: First note that an MPS of bond dimension χ can be described using $O(d\chi^2)$ real parameters, i.e., as a list of $2d-1$ tensors each of size $2 \times \chi \times \chi$. The time required for an MPO-MPS product (both of bond dimension χ) is of order $O(d\chi^2)$. Every truncation takes time $O(d\chi^3)$ since it involves $2d-1$ QR decompositions[1] of matrices of size $2\chi \times 2\chi$ and $2d-1$ singular value decompositions of matrices of size $2\chi \times \chi$, respectively. The final MPS-MPS contraction requires time $O(d\chi^3)$. To contract the tensor network, one needs $2d-2$ calls of both the truncation as well as of the MPO-MPS product. Hence the overall runtime is of order $O(d^2\chi^3)$ offering an efficiently computable approximation of the ML decoder.

The proposed decoder approximately computes for the syndrome s the four error probabilities $\pi(E_s)$, $\pi(E_s\overline{X}\mathcal{S})$, $\pi(E_s\overline{Y}\mathcal{S})$ and $\pi(E_s\overline{Z}\mathcal{S})$ in terms of the probabilities $\{\pi_e\}_{e\in E}$ in runtime $O(d^2\chi^3)$ and chooses the subset for which the probability is highest. This decoder will be referred to as the *BSV decoder*.

Bravyi, Suchara and Vargo [29] numerically compare their approximate ML decoder to the exact ML decoder and the maximum weight perfect matching decoder for different noise models. They show that the minimum weight perfect matching algorithm is usually suboptimal and that the approximate ML decoder converges quickly, becoming virtually indistinguishable from the exact decoder already for a relatively small bond dimension $\chi = 8$. The surface code with biased noise provides an important application of the BSV decoder.

5.3.4 Prior work on noise bias in the surface code

In many physically relevant scenarios, not all error sources are equally likely. For instance, dephasing is the predominant error source for a wide range of quantum computing architectures, e.g. for superconducting qubits [7] and for trapped ions [122]. Often, these biased error models are less severe in the sense that one may tailor quantum error correcting codes to such biases and hence achieve a higher total noise tolerance. This is why the study of biased or asymmetric noise models has a long history and has become increasingly important. Previous work on quantum error correcting codes that are tailored towards biased noise models include [59, 150, 9, 141, 152, 104, 183]. But we have to make the remark that such improvements do not hold generally for all qubit codes. For example, Tuckett et al. [160] showed numerically that in a variant of the colour code [23], the error threshold decreases with increasing noise bias.

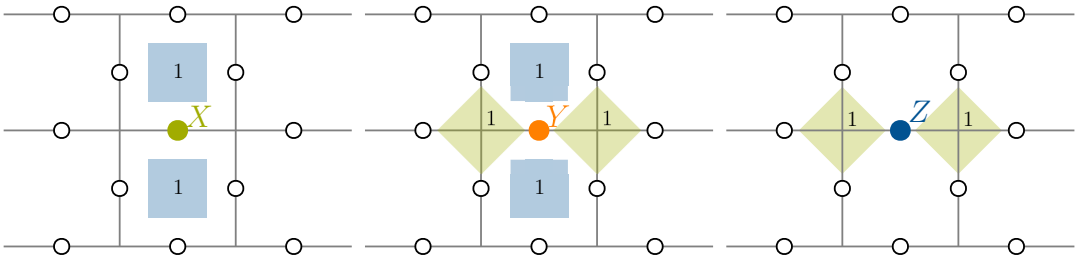
Here, we present results by Tuckett et al. [158, 160] on decoders tailored towards biased noise in the surface code. They assume the noise model is described by i.i.d. Pauli noise

$$\bigotimes_{e\in\mathsf{E}} \mathcal{N}_{(p_I,p_X,p_y,p_Z)} : \mathcal{D}\big((\mathbb{C}^2)^{\otimes|\mathsf{E}|}\big) \to \mathcal{D}\big((\mathbb{C}^2)^{\otimes|\mathsf{E}|}\big) \ ,$$

on all qubits $e \in \mathsf{E}$. The authors introduce the *bias parameter*

$$\eta \coloneqq \frac{p_Y}{p_X + p_Z} \tag{5.86}$$

which relates the error probabilities of the i.i.d. noise and they furthermore assume that $p_X = p_Z$. The value $\eta = 1/2$ corresponds to depolarising noise – where all three basic error types are equally likely, cf. Eq. (3.8) – and the limit $\eta \to \infty$ corresponds to pure Y-noise. Hence in the regime $\eta \in (1/2, \infty)$ the noise is biased towards Y-errors, while by $p_X = p_Z$ the X-



(a) Syndromes of an X-error (b) Syndromes of a Y-error (c) Syndromes of a Z-error

Figure 5.13: Three examples of single-qubit errors on the distance $d = 3$ surface code and the respective error syndromes on the adjacent vertex (X-type in green) stabilisers and plaquette (Z-type in blue) stabilisers. The X-error (as well as the Z-error) is detected by two stabilisers whereas the Y-error by four.

and Z-errors are equally likely.

Tuckett et al. show that the surface code is more resilient towards biased noise ($\eta > 1/2$) than towards unbiased noise ($\eta = 1/2$) in the sense that the maximum likelihood decoder achieves higher error thresholds for fixed total error probability $p = p_X + p_Y + p_Z$. They have a simple argument that motivates this idea: In the surface code, single-qubit Y-errors anticommute with both their neighbouring X- as well as Z-type stabilisers whereas X-errors (or Z-errors) only anticommute with Z-type (X-type) stabilisers, cf. Fig. 5.13. In this sense, the surface code syndromes potentially provide more information on Y-errors (twice as many bits of information) than on X- or Z-type errors.

Tuckett et al. numerically study the error threshold of the (square) surface code for biased Y-noise using the BSV decoder (cf. Section 5.3.3). The numerical analysis suggests that under the assumption $p_X = p_Z$, the surface code can tolerate errors $p_Y < 1/2$ for sufficiently large bias parameters η. More precisely, they numerically find the error threshold $p_c \approx 43.7(1)\%$ for an infinitely high bias [158]. In subsequent work [160], these authors extend this result in two directions. First, they analytically study the case of pure Y-noise. In this case, the surface error correction procedure is equivalent to the concatenation of two classical error correcting codes: the so-called cycle code at the outer level and the repetition code at the inner level. Since both of these codes have threshold values of 50%, the surface code with pure Y-noise achieves an error threshold of $p = p_X + p_Y + p_Z = 50\%$. Second, they also investigate the rotated surface code which is obtained by putting the boundary of the surface code lattice at a 45° rotated angle compared to the standard surface code. Here, an increased error threshold, of up to $p_c = 45.4(2)\%$, in the biased case is numerically found. Tuckett et al. [159] later computed the error threshold of the surface code under biased noise in the full fault-tolerant scenario, i.e., when taking errors in the syndrome measurements into account. Here, they consider the rotated surface code and introduce a decoder based on the minimum-weight perfect matching decoder which exploits the symmetries of the syndromes in the biased case.

Let us remark that these results are not restricted to a bias in Y-direction. They can be simply extended to pure (or dominantly) X- or Z-noise by exchanging the roles of X, Y and Z in the surface code stabilisers (e.g. plaquette stabilisers are products of Y Paulis and vertex stabilisers consist of X Paulis).

[1] A QR decomposition of a complex square matrix M is defined as $M = QR$ where Q is a unitary and R is an upper triangular matrix.

5.4 Modified asymmetric surface-GKP encoding and decoding

This section introduces the *modified asymmetric surface-GKP code* that we numerically analyse in the next sections. We concatenate the rectangular code $\mathsf{GKP}(\mathcal{L}_r)$ for $r > 0$ at the inner level with the distance d square surface code at the outer level and suitably modify the identification of quadratures with logical Pauli operators.

We start in Section 5.4.1 with some preliminary considerations that motivate our construction. Section 5.4.2 introduces the encoding of the modified asymmetric surface-GKP code and explains how it is obtained compared to the standard (symmetric) surface-GKP code. In Section 5.4.3, we analytically derive properties of the noise bias at the level of the encoded GKP-qubits using the logical GKP-error probabilities from Theorem 5.9. In Section 5.4.4, we describe two ways of decoding the modified asymmetric surface-GKP code.

5.4.1 General considerations: noise bias by squeezing

Notation

We consider the distance d surface code with square lattice and 'rough' left/right boundary conditions and 'smooth' top/bottom boundary conditions as described in Section 3.4.5. We will use two different labels for the $d^2+(d-1)^2$ surface code qubits, first $e \in \mathsf{E}$ and second the lower index tuple (j,k) where $j = 1,\dots,2d-1$ and $k = 1,\dots,d$ if j odd and $k = 1,\dots,d-1$ if j even. Therefore, tensor products over all GKP-qubits may be written as

$$\bigotimes_{j,k} := \bigotimes_{j=1}^{2d-1} \bigotimes_{k=1}^{d-(1+(-1)^j)/2} = \bigotimes_{e\in\mathsf{E}} .$$

The syndrome of a GKP-qubit $e \in \mathsf{E}$ or with label (j,k) is labelled using a lower index e or j,k, respectively, i.e., it is written as $s_e = s_{j,k}$.

Since the concatenation of two codes is considered – where the logical noise channel of the inner code corresponds to the physical noise channel of the outer code – one has to make an adaption in notation: one cannot use the same notation with bars $\bar{\cdot}$ for the logical errors of the GKP as well as for those of the surface code. This is why we change notation for the GKP code: in contrast to before (cf. Section 5.2.4) where the residual or logical noise of the GKP code was labelled with bars – i.e., $\overline{L} \in \{\overline{I},\overline{X},\overline{Y},\overline{Z}\}$ for the logical operations or $\overline{\mathcal{N}}_{(p_{\overline{I}},p_{\overline{X}},p_{\overline{Y}},p_{\overline{Z}})}$ for the logical noise channel – we will only use bars for the logical noise channel of the outer (surface) code and instead omit the bars for this inner code. In conclusion, the quantities $L \in \{I,X,Y,Z\}$ or $\mathcal{N}_{(p_I,p_X,p_Y,p_Z)}$ denote the logical operations and the logical noise channel of the GKP code, respectively.

Noise bias by squeezing

The main result of this chapter concerns the enhanced error threshold of modified asymmetric surface-GKP codes [76] by exploiting an engineered noise bias in the concatenated code. More precisely, we use that, by Theorem 5.9, physically symmetric displacement noise in the

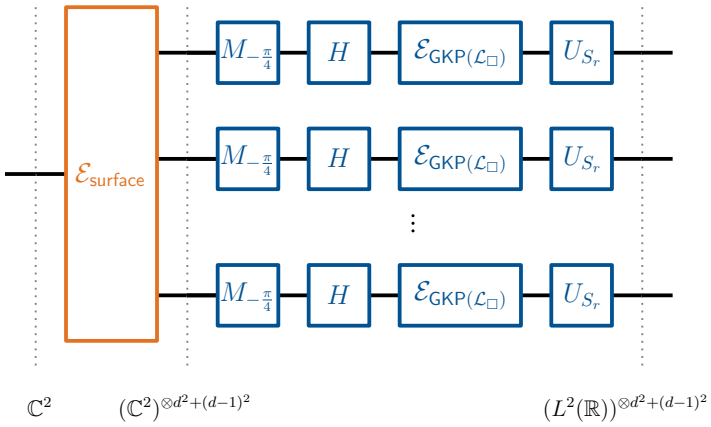


Figure 5.14: Encoding circuit for the modified asymmetric surface-GKP code. A single logical qubit is first mapped to a state on $d^2 + (d-1)^2$ qubits using the encoding associated with the distance d square surface code with rough top/bottom and smooth left/right boundary conditions. Each of these qubits is then encoded into a single bosonic mode using the modified asymmetric GKP encoding, which may be written as the modified encoding using the unitary gates H (Hadamard) and $M_{-\pi/4}$ (phase gate) as well as the rectangular lattice GKP encoding.

rectangular lattice GKP code is transformed to an asymmetric noise on the logical level of the encoded GKP-qubit.

Let us argue why we consider a modified and asymmetric version of the standard surface-GKP code. Assume that the physical noise on the modes is symmetric, i.e., it is given by the symmetric displacement noise channel $\mathcal{N}_{\sigma^2 I_2}$ from Eq. (5.41) on all modes, where the noise strength is characterised by a single parameter, the variance $\sigma^2 > 0$. By Theorem 5.11, a Gaussian unitary encoding of the square lattice GKP code with the squeezing unitary U_{S_r} from Eq. (5.31) transforms this symmetric physical noise into a biased logical noise at the level of the $\mathsf{GKP}(\mathcal{L}_r)$-qubit.

More precisely, if $r = 1$, i.e. in the symmetric case $\mathsf{GKP}(\mathcal{L}_\square)$, then $q_{\overline{X}} = q_{\overline{Z}}$ (from Eqs. (5.59) and (5.60)) and hence the probabilities $p_{\overline{L}}$ of a logical $L \in \{X, Y, Z\}$ on the encoded GKP-qubit are given by

$$p_{\overline{X}} \coloneqq q_{\overline{X}}(1 - q_{\overline{Z}}) = q_{\overline{Z}}(1 - q_{\overline{X}}) =: p_{\overline{Z}} \, .$$

In contrast, for $r > 1$ one has $q_{\overline{X}} < q_{\overline{Z}}$ and therefore

$$p_{\overline{X}} \coloneqq q_{\overline{X}}(1 - q_{\overline{Z}}) < q_{\overline{Z}}(1 - q_{\overline{Z}}) < q_{\overline{Z}}(1 - q_{\overline{X}}) =: p_{\overline{Z}} \, .$$

As a consequence, for $\mathsf{GKP}(\mathcal{L}_\square)$, the logical noise channel $\overline{\mathcal{N}}_{(p_{\overline{I}}, p_{\overline{X}}, p_{\overline{Y}}, p_{\overline{Z}})}$ on the encoded GKP-qubit is that of independent X- and Z-noise with both of them being equally likely. This is different in the asymmetric case. One still has a channel of independent X- and Z-noise but as $r > 1$ grows, the noise bias grows: Z-errors are more likely to occur than X-errors. Since the surface code is more resilient to noise which is biased towards Y-errors, it is useful to engineer such a bias on the logical GKP-qubit via a suitable GKP-encoding. More precisely, we make two adaptions of the standard (square) GKP encoding, the first one is the asymmetry of the unitary encoding U_{S_r} and the second one is a (cyclic) permutation of the roles of X-, Y-, and Z-errors.

5.4.2 Encoding

Let us first consider the standard (square lattice) surface-GKP encoding

$$\mathcal{E}_{\mathsf{surface-GKP}} \coloneqq (\mathcal{E}_{\mathcal{L}_\square})^{\otimes d^2+(d-1)^2} \circ \mathcal{E}_{\mathsf{surface}} \ , \tag{5.87}$$

where the surface code with encoding channel $\mathcal{E}_{\mathsf{surface}}$ is concatenated with the square lattice (or symmetric) GKP code $\mathsf{GKP}(\mathcal{L}_\square)$ with encoding $\mathcal{E}_{\mathcal{L}_\square}$, cf. Fig. 5.10. Note that throughout this chapter, we consider the distance d surface code based on a square lattice with rough top/bottom and smooth left/right (as presented in Section 3.4.5). This specific choice is motivated by the fact that Tuckett et al. studied the same code in their work [158] showing its increased resistance against biased noise. So we can compare our results with theirs. Moreover, our result serves rather as a proof of principle. Further analyses with other lattice structures of the surface code – rectangular lattices, different boundary conditions or a rotated lattice – may show improved noise tolerance. Furthermore, if the goal is the best code in practice, not only the error threshold but also the sub-threshold behaviour should be taken into account, but this goes beyond the scope of this thesis.

The *modified asymmetric surface-GKP encoding* has two adaptions compared to (5.87).

(i) First, we choose the GKP code with rectangular lattice $\mathcal{L}_r$ for the asymmetry ratio $r > 1$ instead of the square lattice GKP code. By Lemma 5.5 the code $\mathsf{GKP}(\mathcal{L}_r)$ is obtained by applying an additional Gaussian unitary U_{S_r} to the square lattice GKP encoding $\mathcal{E}_{\mathsf{GKP}(\mathcal{L}_\square)}$ where S_r is the squeezing matrix from Eq. (5.31). This is called the *asymmetric* GKP encoding.

(ii) Second, we introduce an additional unitary in the (asymmetric) GKP encoding; the logical rectangular lattice GKP-qubits are not directly used as the physical qubits of the surface code: instead of identifying Z-eigenstates of the logical GKP-qubit – these are $U_{S_r}|\overline{0}\rangle_\square$ and $U_{S_r}|\overline{1}\rangle_\square$ – with the physical surface code qubit's Z-eigenstates (as in (i)) we identify them with the Y eigenstates of the physical surface code qubits. This corresponds to the unitary encoding

$$\mathcal{D}(\mathbb{C}^2) \to \mathcal{D}(\mathbb{C}^2) \quad , \quad \rho \mapsto H M_{-\pi/4}\, \rho\, M_{-\pi/4}^\dagger H^\dagger \ ,$$

where H is the Hadamard gate (cf. Eq. (2.51)) and $M_{-\pi/4}$ is the phase gate for angle $-\pi/4$ (cf. Eq. (2.53)). It maps the Y eigenstates of the physical surface code qubits to the rectangular lattice GKP basis states as

$$\begin{aligned} |+i\rangle &\mapsto U_{S_r}|\overline{0}\rangle_\square \\ |-i\rangle &\mapsto U_{S_r}|\overline{1}\rangle_\square \end{aligned}$$

and extend by linearity to a modified GKP-qubit encoding $\mathbb{C} \to L^2(\mathbb{R})$. This is what we call the *modified* GKP encoding.

The encoding step (ii) is made in order to identify the logical operators $\overline{X}_r$ and $\overline{Z}_r$ of the (unmodified) GKP-qubit with the physical Z- and Y-operators of the surface code, respectively, i.e.,

$$\begin{aligned} Z &\leftrightarrow U_{S_r}\overline{X}_\square U_{S_r}^\dagger = \overline{X}_r \ , \\ Y &\leftrightarrow U_{S_r}\overline{Z}_\square U_{S_r}^\dagger = \overline{Z}_r \ . \end{aligned}$$

As a consequence, for $r > 1$, the noise at the level of the logical GKP-qubits (equivalently the physical noise of the surface code) is biased towards Y-errors (instead of a bias towards X-errors without the modification). The encoding circuit is depicted in Fig. 5.14. In summary, the *modified asymmetric GKP encoding* which incorporates the steps (i) and (ii) is denoted by $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$ and has the encoding unitary

$$\mathcal{E}_{\mathrm{mod},\mathcal{L}_r} : \mathcal{D}(\mathbb{C}^2) \to \mathcal{D}(L^2(\mathbb{R})) \quad , \quad \rho \mapsto H M_{-\pi/4} \left(\mathcal{E}_{\mathcal{L}_\square}(U_{S_r} \rho U_{S_r}^\dagger) \right) M_{-\pi/4}^\dagger H^\dagger \ .$$

where to shorten notation, we write $\mathcal{E}_{\mathrm{mod},\mathcal{L}_r} = \mathcal{E}_{\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)}$. Note that the closest lattice point recovery channel of the modified asymmetric GKP code is given by $\mathcal{R}_{\mathcal{L}_r}$, i.e., it is the same as the one for the asymmetric GKP code. In summary, the encoding that we investigate in this chapter is

$$\mathcal{E}_{\mathsf{asym-mod-surface-GKP}} \coloneqq (\mathcal{E}_{\mathrm{mod},\mathcal{L}_r})^{\otimes d^2+(d-1)^2} \circ \mathcal{E}_{\mathsf{surface}} \ .$$

5.4.3 Biased noise on the GKP-qubits

Before presenting our numerical analysis, let us analytically discuss the logical noise channel on the GKP-qubits and quantify the bias of the logical noise channel associated with the modified asymmetric GKP code.

Using the modified asymmetric GKP encoding (the rectangular lattice GKP encoding from step (i) as well as the modification from step (ii) which permutes the roles of X, Y and Z), the logical noise channel $\mathcal{N}_\pi$ on every $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$-qubit with probabilities

$$\pi \coloneqq (p_I, p_X, p_Y, p_Z) = \big((1-q_{\overline{X}})(1-q_{\overline{Z}}), q_{\overline{X}}q_{\overline{Z}}, (1-q_{\overline{X}})q_{\overline{Z}}, q_{\overline{X}}(1-q_{\overline{Z}})\big) \tag{5.88}$$

for $q_{\overline{X}}$ and $q_{\overline{Z}}$ from Eqs. (5.59) and (5.60), respectively, is forwarded to the surface code. For every qubit, it describes independent Z- and Y-noise, i.e.,

$$\begin{array}{rcccl} p_I &=& (1-q_{\overline{Z}}) &\cdot& (1-q_{\overline{X}}) \\ p_X &=& q_{\overline{Z}} &\cdot& q_{\overline{X}} \\ p_Y &=& q_{\overline{Z}} &\cdot& (1-q_{\overline{X}}) \\ p_Z &=& (1-q_{\overline{Z}}) &\cdot& q_{\overline{X}} \end{array} \tag{5.89}$$

where the quantities $q_{\overline{X}}$ and $q_{\overline{Z}}$ are given by Eqs. (5.59) and (5.60), respectively. We use the terms p_I, p_X, p_Y, p_Z without bars to denote the logical error probabilities of the *modified* (asymmetric) GKP code in contrast to the expressions with bars for the (asymmetric) GKP code. As argued in Section 5.4.1, the interchanging of X, Y and Z by the modification achieves a bias towards Y instead of X-errors. More precisely, the noise π is biased towards Y-errors as r grows.

To prepare the numerical results on error thresholds of the modified asymmetric surface-GKP code, let us (analytically) show a few properties on the logical GKP-qubit error probabilities.

Lemma 5.13. Let $r, \sigma > 0$ and let $1-q_{\overline{X}}, 1-q_{\overline{Z}}$ be the quantities from Eqs. (5.59) and (5.60).

(i) As a function of σ, both $1-q_{\overline{X}}$ and $1-q_{\overline{Z}}$ are monotonically decreasing. Furthermore, as a function of r, $1-q_{\overline{X}}$ monotonically increases whereas $1-q_{\overline{Z}}$ monotonically decreases.

(ii) In the limit $\sigma \to \infty$, one has $1-q_{\overline{X}}, 1-q_{\overline{Z}} \to 1/2$ for all asymmetry ratios $0 < r < \infty$ and all noise variances $0 < \sigma^2 < \infty$.

(iii) One has $0 \leq q_{\overline{X}}, q_{\overline{Z}} < 1/2$.

(iv) For $r > 1$ one has $q_{\overline{X}} < q_{\overline{Z}}$ for all $\sigma > 0$.

(v) In the limit $r \to \infty$, one has $1 - q_{\overline{X}} \to 1$ and $1 - q_{\overline{Z}} \to 1/2$.

□

Proof. Note that by their definition in Eqs. (5.59) and (5.60), the quantities $q_{\overline{X}}$ and $q_{\overline{Z}}$ depend on the parameters σ and r. To clarify this dependence in the notation, define the function $\sigma \mapsto h(\sigma)$ such that $h(\sigma) \coloneqq q_{\overline{X}}$ for $0 < \sigma < \infty$ and for fixed $r = 1$. For $r \neq 1$, the quantity $q_{\overline{X}}$ is related to $h(\sigma)$ (i.e., $q_{\overline{X}}$ for $r = 1$) by squeezing $\sigma \mapsto \sigma/\sqrt{r}$ and similarly, $q_{\overline{Z}}$ can be obtained from $h(\sigma)$ by squeezing $\sigma \mapsto \sigma\sqrt{r}$.

(i) Let us show that $d/d\sigma(1 - h(\sigma)) < 0$. This is true by the following computation;

$$\begin{aligned}
\frac{d}{d\sigma}\big(1 - h(\sigma)\big) &= -\tfrac{2\sqrt{2}}{\sigma^2} \sum_{m\in\mathbb{Z}} e^{-\frac{2\pi}{\sigma^2}(m^2+\frac{1}{16})} \left(\tfrac{1}{4}\cosh\left(\tfrac{\pi}{\sigma^2}m\right) - m\sinh\left(\tfrac{\pi}{\sigma^2}m\right)\right) \\
&\leq -\frac{2\sqrt{2}}{\sigma^2} \sum_{m\in\mathbb{Z}} e^{-\frac{2\pi}{\sigma^2}(m^2+\frac{1}{16})} \left(\frac{1}{4} - m\right) \cosh\left(\frac{\pi}{\sigma^2}m\right) \\
&= -\frac{2\sqrt{2}}{\sigma^2} \sum_{m\in\mathbb{Z}} \frac{1}{4} e^{-\frac{2\pi}{\sigma^2}(m^2+\frac{1}{16})} \cosh\left(\frac{\pi}{\sigma^2}m\right) < 0
\end{aligned}$$

where we used that $x\sinh(x) \leq x\cosh(x)$ and $-\cosh(x) \leq -\cosh(0) = -1$ for all $x \in \mathbb{R}$ as well as that the function $m \mapsto -me^{-\frac{2\pi}{\sigma^2}(m^2+1/16)} \cosh\left(m\pi/\sigma^2\right)$ is antisymmetric in m and $\cosh(x) > 0$ for all $x \in \mathbb{R}$. Using the relations between $h(\sigma)$ and $1 - q_{\overline{X}}$ as well as $1 - q_{\overline{Z}}$, one obtains the desired relations.

(ii) For large (infinite) σ, the integrand in the Eqs. (5.73) and (5.74) becomes a widely (infinitely) spread normal distribution. For both the above referenced equations, the integration goes from $c_r(k - 1/4)$ to $c_r(k + 3/4)$ for all $k \in \mathbb{Z}$ (where the prefactors c_r depend on r but they are finite), i.e., the interval of integration always comprises half of the real line. Hence, in the limit $\sigma \to \infty$, we find $1 - q_{\overline{X}} \to 1/2$ and $1 - q_{\overline{Z}} \to 1/2$ irrespective of the asymmetry ratio $r > 0$.

(iii) It is sufficient to show that $0 \leq h(\sigma) < 1/2$ for all $\sigma > 0$. More precisely, we use that $h(\sigma) \to 1/2$ as $\sigma \to \infty$ (by the previous part) and that $h(0) = 0$ by direct computation. Then the claim follows by the monotonicity of the function h as shown in part (i).

(iv) Let $r > 1$. The claim follows directly by the monotonicity of h since one has

$$q_{\overline{X}} = h(\sigma/\sqrt{r}) < h(\sigma\sqrt{r}) = q_{\overline{Z}} \quad \text{for all } \sigma > 0 \,.$$

(v) Now consider the limit $r \to \infty$ for fixed $0 < \sigma < \infty$. Then, by the same reasoning as in the previous part, one finds $h(\sigma/\sqrt{r}) = q_{\overline{X}} \to 0$ and $h(\sigma\sqrt{r}) = q_{\overline{Z}} \to 1/2$. This corresponds (in the modified GKP code) to no Z-errors and no X-errors, but pure Y-noise with probability $1/2$.

■

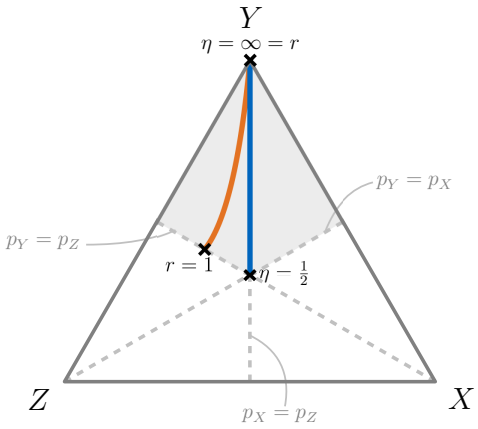


Figure 5.15: Qualitative comparison of the noise models: the encoded modified asymmetric GKP-qubit noise (orange) and the biased noise model considered by Tuckett et al. [158] (blue). For a Pauli error distribution $\pi = (1-p, p_X, p_Y, p_Z)$ where an L-error occurs with probability p_L for $L \in \{X, Y, Z\}$ and where $p = p_X + p_Y + p_Z$ is the total error probability, the simplex represents π in the following way: first, vertices denoted by $L \in \{X, Y, Z\}$ correspond to pure L-noise, i.e., $p_L = p$; second the edges opposite of the L-vertex correspond to $p_L = 0$; third the dotted lines denote distributions of two equally likely errors as indicated above; fourth, the point where these dottet lines meet corresponds to depolarising noise of equally likely errors $p_X = p_Y = p_Z$ and the shaded area corresponds to noise biased towards Y-errors. The blue line represents the noise model of biased Y-noise where $p_Y > p_X = p_Z$ as a function of the bias parameter $\eta = p_Y/(p_X + p_Z)$ whereas the orange curve qualitatively depicts the noise model of the logical qubit $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$ as a function of the asymmetry ratio r.

Part (iii) of Lemma 5.13 implies that $q_{\overline{Z}} < 1 - q_{\overline{Z}}$ for all $r, \sigma > 0$ and. Hence

$$p_X \coloneqq q_{\overline{X}} q_{\overline{Z}} < q_{\overline{X}}(1 - q_{\overline{Z}}) =: p_Z \ . \tag{5.90}$$

Then the following holds for the residual GKP-qubit noise channel $\mathcal{N}_\pi$ with probabilities $\pi = (p_I, p_Y, p_Z, p_X)$ from Eq. (5.89).

- **Symmetric case**: If $r = 1$, then by their respective definitions (from Eqs. (5.59) and (5.60)) one has $q_{\overline{X}} = q_{\overline{Z}}$ for all $\sigma \geq 0$. Therefore,

  $$p_X < p_Z = p_Y$$

  using Eq. (5.90) (i.e., Lemma 5.13). The modified square lattice GKP-qubit hence has equally likely Z and Y-errors and less likely X-errors.

- **Asymmetric case**: If $r > 1$ by part (iv) of Lemma 5.13 one has $q_{\overline{X}} < q_{\overline{Z}}$ and therefore $p_Z \coloneqq q_{\overline{X}}(1 - q_{\overline{Z}}) < q_{\overline{Z}}(1 - q_{\overline{Z}}) < q_{\overline{Z}}(1 - q_{\overline{X}}) =: p_Y$. Furthermore, combining this with Eq. (5.90) one finds

  $$p_X < p_Z < p_Y \ .$$

  As a result, X-errors occur less likely than Z-errors which are less likely than Y-errors on the encoded modified asymmetric GKP-qubits. This gives biased noise towards Y-errors. By the monotonicity of $q_{\overline{X}}$, and $q_{\overline{Z}}$ in r, one additionally has that p_Z (and p_Y) are monotonically decreasing (respectively increasing) as a function of r. Therefore, the noise bias towards Y-errors increases with growing asymmetry.

In this sense the logical noise channel on the encoded GKP-qubit is biased towards Y-errors and furthermore this bias increases with increasing asymmetry ratio r. But note that this noise model (and our bias measure r) differs from the one considered by Tuckett et al. [158] (cf. Section 5.3.4). Whereas we look at the error distribution π from Eq. (5.88) and use the asymmetry ratio r as a measure of bias, Tuckett et al. study a distribution of the form $\pi = (1-2p_X-p_Y, p_X, p_Y, p_X) = (1-p_Y(1+\eta)/\eta, p_Y/(2\eta), p_Y, p_Y/(2\eta))$ for the Y-error probability $p_Y = 2\eta p_X$ and the bias parameter $\eta > 1/2$ from Eq. (5.86). As a consequence, we consider a noise model of independent Y- and Z-noise (with bias towards Y) whereas Tuckett et al. consider equally likely X- and Z-errors and more likely Y-errors. One studies two different noise models that are both biased towards Y-errors and have the same limiting noise model of pure Y-noise in the limits $r, \eta \to \infty$. Fig. 5.15 shows a qualitative comparison of both noise models. In order to compare both bias parameters, let us compute η of the modified asymmetric logical GKP-qubits with different asymmetry ratios r. Note that, here, the bias η does not only depend on the asymmetry ratio but also on the physical noise strength σ of the displacement noise. A direct computation of the bias parameter

$$\eta(\sigma, r) := \frac{p_Y}{p_X + p_Z} = \frac{(1-q_{\overline{X}})q_{\overline{Z}}}{q_{\overline{X}}q_{\overline{Z}} + q_{\overline{X}}(1-q_{\overline{Z}})} = \frac{(1-q_{\overline{X}})q_{\overline{Z}}}{q_{\overline{X}}} \tag{5.91}$$

shows that the bias η increases with growing asymmetry ratio r but at the same time decreases with increasing noise strength σ. To give the reader an impression of the noise bias, we have computed several values of η for different parameters σ and r in the range of what we consider later in our numerical analysis (and is close to the computed thresholds):

$$\begin{array}{cccc}
\eta(0.4, 1.0) \approx 0.97, & \eta(0.5, 1.0) \approx 0.92, & \eta(0.6, 1.0) \approx 00.86, & \eta(0.7, 1.0) \approx 0.79, \\
\eta(0.4, 2.0) \approx 67.68, & \eta(0.5, 2.0) \approx 17.01, & \eta(0.6, 2.0) \approx 7.73, & \eta(0.7, 2.0) \approx 4.59, \\
\eta(0.4, 3.0) \approx 1614.50, & \eta(0.5, 3.0) \approx 141.71, & \eta(0.6, 3.0) \approx 36.06, & \eta(0.7, 3.0) \approx 14.99, \\
\eta(0.4, 4.0) \approx 28490.24, & \eta(0.5, 4.0) \approx 935.75, & \eta(0.6, 4.0) \approx 137.87, & \eta(0.7, 4.0) \approx 41.04\ .
\end{array}$$

The maximal noise bias of $\eta \approx 30,000$ for $r = 4$ and $\sigma = 0.4$ is already quite large.

5.4.4 Decoding without and with GKP-syndrome information

The decoding of the concatenated code contains two steps: decoding of the GKP code and then decoding of the surface code. The physical error channel of the outer code is given by the logical noise channels of the $d^2 + (d-1)^2$ GKP-qubits. Here we consider the BSV decoder [29] (cf. Section 5.3.3 for a brief recap of this decoder) to decode the surface code. It can be run without using the GKP-syndrome side information, as shown in Fig. 5.16, or with the use of it as depicted in Fig. 5.17.

In the first case – when the GKP-syndrome information is discarded – the BSV decoder of the surface code is computed with respect to the noise model

$$\bigotimes_{j,k} \mathcal{N}_\pi = (\mathcal{N}_\pi)^{\otimes d^2 + (d-1)^2}\ . \tag{5.92}$$

This i.i.d. Pauli noise channel is the $d^2 + (d-1)^2$-fold tensor product of the logical noise channel associated with a single GKP-qubit $\mathcal{N}_\pi := \mathcal{E}^\dagger_{\mathrm{mod},\mathcal{L}_r} \circ \mathcal{R}_{\mathcal{L}_r} \circ \mathcal{N}_{\sigma^2 I} \circ \mathcal{E}_{\mathrm{mod},\mathcal{L}_r}$, where $\pi = (p_I, p_X, p_Y, p_Z)$ is the probability distribution from Eq. (5.88).

When the GKP-syndrome information is kept and used as input of the BSV decoder, one may use the considerations from Section 5.3.2. In contrast to Eq. (5.92), the GKP-syndrome

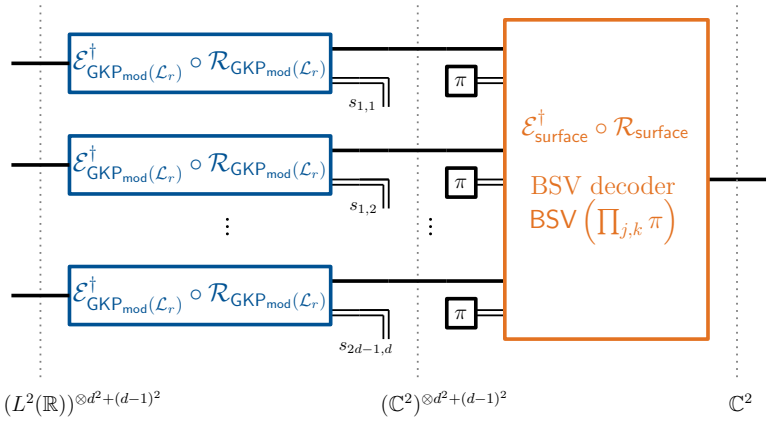


Figure 5.16: Decoding circuit of the surface-GKP code without using the GKP-syndrome information. For every qubit (j,k), the GKP error correction is given by the channel $\mathcal{E}^\dagger_{\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)} \circ \mathcal{R}_{\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)}$: Its output is a qubit state which is passed to the decoder of the surface code. The decoder is chosen to perform well under the noise channel $\bigotimes_{j,k} \mathcal{N}_\pi$, i.e., it depends on the (same) prior error distribution $\pi = (p_I, p_X, p_Y, p_Z)$ from Eq. (5.88) on all of the $d^2 + (d-1)^2$ qubits. The syndromes $s_{1,1}, \ldots, s_{2d-1,d} \in \mathcal{V}_{\mathcal{L}^\perp}$ of the GKP-error correction are discarded and not taken into account.

information can be taken into account when computing the logical Pauli error distribution of the outer surface code. Then the BSV decoder is conditioned on the effective error channel

$$\bigotimes_{j,k} \mathcal{N}_{\pi_{j,k}} \ .$$

Here, for every qubit (j,k), the error distribution $\pi_{j,k}$ depends on the respective GKP syndrome $s_0 = s_{j,k}$, i.e., it is given by

$$\pi_{j,k} = \left((1-q_Z^{s_0})(1-q_X^{s_0}), q_Z^{s_0} q_X^{s_0}, q_Z^{s_0}(1-q_X^{s_0}), (1-q_Z^{s_0}) q_X^{s_0}\right) \ , \qquad \text{where } s_0 = s_{j,k} \ , \tag{5.93}$$

where the quantities $1 - q_X^{s_0}$ and $1 - q_Z^{s_0}$ are defined in Eqs. (5.63).

5.5 Simulation methods

The central result of this chapter concerns the error threshold of the modified asymmetric surface-GKP code and how it depends on the asymmetry ratio r of the underlying GKP lattice. We have analytically shown in the precious section that by a squeezing unitary in the GKP encoding, one can engineer a noise bias at the level of the logical GKP-qubit. Since this asymmetric or biased noise is forwarded to the surface code – being a biased noise of the 'physical' surface code qubits – one may use the enhanced noise tolerance of the surface code towards biased noise in order to enhance the noise resilience of the whole concatenated code. We analyse this hypothesis by simulating the error correction procedure of such codes numerically using a Monte-Carlo algorithm. From these simulations, we compute the error threshold of the concatenated code for different asymmetry ratios r. Let us hence describe the simulation method used to obtain these threshold estimates.

In Section 5.5.1, we describe the individual subroutines of Monte Carlo simulations for the different considered variants of the modified asymmetric surface-GKP code (i.e., different GKP-lattices and surface code decoders). Section 5.5.2 discusses how the threshold estimated

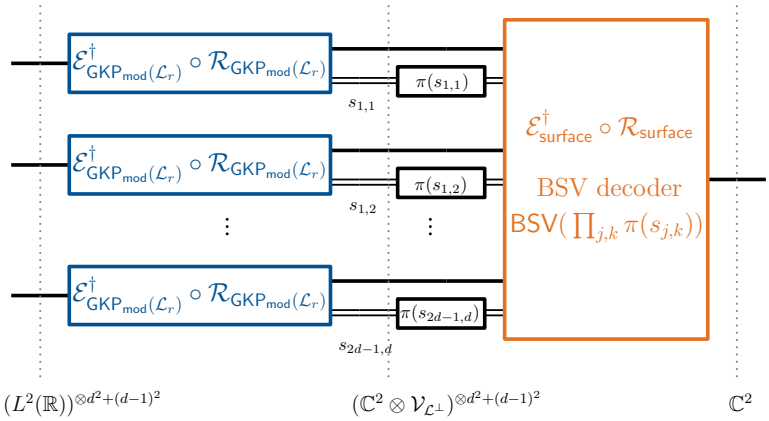


Figure 5.17: Decoding circuit of the surface-GKP code using the GKP-syndrome information: For every qubit (j,k), the GKP error correction channel $\mathcal{E}^\dagger_{\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L})} \circ \mathcal{R}_{\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L})}$ is seen as a quantum instrument with two outputs: the post-correction single-qubit quantum state and the GKP error syndrome $s_{j,k} \in \mathcal{V}_{\mathcal{L}^\perp}$ (the upper index denotes the respective qubit). The surface code decoder uses both as inputs. It is chosen with respect to the noise channel $\bigotimes_{j,k} \mathcal{N}_{\pi(s_{j,k})}$. More precisely, we condition the BSV decoder of the surface code on the single-qubit input error distribution $\pi_{j,k} = \pi(s_{j,k})$ for qubit (j,k) where $\pi(s) = (p^s_I, p^s_X, p^s_Y, p^s_Z)$ from Eq. (5.93).

are obtained from the simulation data and Section 5.5.3 briefly explains our choice parameter choices. Finally, we estimate in Section 5.5.4 the error which we make by approximating the error probabilities associated with the logical GKP-qubits.

5.5.1 Monte-Carlo simulations

We perform multiple Monte-Carlo simulations of the error correction procedure.

Different scenarios

We consider the modified asymmetric surface-GKP code as defined in Section 5.4. These results are compared to different scenarios such as the hexagonal and the asymmetric hexagonal surface-GKP code. We simulate the error correction process for the following four scenarios.

(i) Modified **symmetric** surface-GKP code, i.e., the concatenation of $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_\square)$ and square $d \times d$-surface code; the decoders are the closest lattice point decoder for the GKP code as well as the BSV decoder for the surface code where the latter does not use the GKP syndrome information.

(ii) Modified **asymmetric** surface-GKP code, i.e., the concatenation of $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$ for different ratios r and square $d \times d$-surface code; the decoders are the closest lattice point decoder for the GKP code as well as the BSV decoder of bond dimension χ for the surface code where the latter does not use the GKP syndrome information.

(iii) Modified **asymmetric** surface-GKP code, i.e., the concatenation of $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_{r=2})$ and square $d \times d$-surface code; the decoders are the closest lattice point decoder for the GKP code as well as the BSV decoder of bond dimension χ for the surface code where the latter **uses the GKP syndrome information** as an input.

(iv) Modified **hexagonal-asymmetric** surface-GKP code, i.e., the concatenation of the code $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_{r=2,\hexagon})$ and square $d \times d$-surface code; the decoders are the closest lattice point decoder for the GKP code as well as the BSV decoder of bond dimension χ for the surface code where the latter **uses the GKP syndrome information** as an input.

Note that we assume perfect syndrome measurements (in all scenarios), i.e., errors do not occur in the measurement process, neither in the GKP syndrome nor in the surface code syndrome measurements. Furthermore, we use the closest lattice point decoder to decode the respective GKP code.

Subroutine of the Monte-Carlo simulation

Every instance of the Monte-Carlo simulation simulates the surface-GKP error correction procedure of one randomly drawn error. The pseudocodes in Figs. 5.18 (without using the GKP syndrome information) and 5.19 (with the use of GKP syndrome information) describe one such instance. In more detail, the algorithm works as follows:

For every bosonic mode (j,k), one first chooses a random displacement error according to the symmetric Gaussian displacement noise (5.41) with variance σ^2, i.e., one randomly draws a displacement $\nu \in \mathbb{R}^2$ according to the distribution $\mathsf{N}(\mathbf{0}, \sigma^2 I_2)$, one for every considered mode. The next step is to compute the logical error on every GKP-qubit. It is determined according to the after closest lattice point correction for $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$ according to the identification from Eq. (5.13) where $\mathcal{L} = \mathcal{L}_r$ but with the interchanging of the roles of $X \mapsto Y \mapsto Z \mapsto X$ by the modification. Hence the collection of Pauli errors

$$\begin{aligned} &I \quad \text{if} \quad \nu \in \mathcal{V}_r^{\perp} + \mathcal{L}_r \ , && X \quad \text{if} \quad \nu \in \mathcal{V}_r^{\perp} + \mathcal{L}_r + \ell_{r2}^{\perp} + \ell_{r1}^{\perp} \ , \\ &Y \quad \text{if} \quad \nu \in \mathcal{V}_r^{\perp} + \mathcal{L}_r + \ell_{r1}^{\perp} \ , && Z \quad \text{if} \quad \nu \in \mathcal{V}_r^{\perp} + \mathcal{L}_r + \ell_{r2}^{\perp} \ , \end{aligned} \tag{5.94}$$

(where $\ell_{r1}^{\perp}$, $\ell_{r2}^{\perp}$, $\mathcal{L}_r$, and $\mathcal{V}_r^{\perp}$ are from (5.32), (5.30), (5.33)) is forwarded to the surface code. For $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_{\hexagon,r})$, (5.94) is defined similarly, replacing the vectors, the lattice and the Voronoi cell by the respective quantities from (5.38), (5.37), (5.39).

Let us first focus on the scenario where the **GKP-syndrome information is ignored**: The a priori error distribution of the modified asymmetric GKP code can be analytically computed; as discussed in the previous section, it is that of independent Y- and Z-noise with probability distribution (5.88) for every qubit (j,k), i.e.,

$$\pi_{j,k} = \pi = \big((1-q_{\overline{X}})(1-q_{\overline{Z}}), q_{\overline{X}}q_{\overline{Z}}, (1-q_{\overline{X}})q_{\overline{Z}}, q_{\overline{X}}(1-q_{\overline{Z}})\big) \ , \tag{5.95}$$

where $q_{\overline{X}}$ and $q_{\overline{Z}}$ are given by Eqs. (5.59) and (5.60), respectively, which themselves depend on the noise parameter σ and the asymmetry ratio r. Note that the formulas in Eqs. (5.59) and (5.60) are impractical since they contain infinite sums. This is why one introduces an approximation using a finite sum as discussed later in this section in the paragraph on *cutoff analysis*.

The actual error (5.94) is passed on to the outer code, the distance d square surface code. To decode the latter, one uses the BSV decoder as described in Section 5.3.3. The BSV decoder has the following inputs: the distance d of the surface code, the bond dimension χ of the BSV algorithm and the a priori probability distributions for the $d^2 + (d-1)^2$ surface code qubits (cf. Eq. (5.82)) where in the case at hand, for each qubit (j,k), one takes the same logical GKP-qubit probability distribution $\pi_{j,k} = \pi$ from Eq. (5.95). Then the BSV decoder identifies the most likely Pauli error which has produced the surface code syndrome

function MCWithoutSideInfo(σ^2, d, r)

Input: Variance σ^2, code distance d, asymmetry ratio r
Output: Residual logical Pauli error $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$

---

for each qubit (j,k) **do**
Sample displacement error vector $\nu_{j,k} \in \mathbb{R}^2$ according to $\mathsf{N}(\mathbf{0}, \sigma^2 I_2)$.
Compute the (logical) Pauli error $E_{j,k} \in \{I, X, Y, Z\}$ from $\nu_{j,k}$ according to (5.94) for $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$.
end for
Set $E_s \coloneqq \bigotimes_{j,k} E_{j,k}$.
Compute the distribution $\pi = (p_I, p_X, p_Y, p_Z)$ according to Eq. (5.95) for $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$.
Use the BSV decoder with input $\pi^{(d^2+(d-1)^2)}$ to compute an $(d^2+(d-1)^2)$-qubit Pauli correction $C \in \mathcal{P}_{(d^2+(d-1)^2)}$.
Compute $\overline{L} = \arg\max_{\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}} 1_{E_s C \in \overline{L}\mathcal{S}}$ (i.e., decide which coset $E_s C$ belongs to).
end function

Figure 5.18: Subroutine MCWithoutSideInfo simulates one instance of the error-recovery process when GKP side information is ignored. It returns the residual logical error.

and outputs a $d^2 + (d-1)^2$-qubit correction operation C. In a final step, the algorithm determines which coset $\overline{L}\mathcal{S}$ the product of actual error and correction belongs to and outputs the representative $\overline{I}$, $\overline{X}$, $\overline{Y}$ or $\overline{Z}$ of this coset.

If the output of the algorithm is $\overline{L} = \overline{I}$, then the modified asymmetric surface-GKP code error correction is successful whereas $\overline{L} \in \{\overline{X}, \overline{Y}, \overline{Z}\}$ then the error correction was unsuccessful and a logical error remains. The algorithm 5.18 outputs the residual Pauli error on the logical surface-GKP-qubit after a single randomly drawn error. Hence, by the law of large numbers, the average over a large number of simulation results, numerically estimates the resulting logical error probability of the modified asymmetric surface-GKP code. This (logical) error probability is denoted as $P_{\mathsf{err}}(\sigma, d, r) = 1 - P_{\mathsf{success}}(\sigma, d, r)$ and is a function of the three variables σ, d and r.

In addition to the effect of the asymmetry ratio on the noise threshold, we also analyse other scenarios, cf. Section 5.6.3 for the results. This includes finding the error threshold of the modified asymmetric surface-GKP code when the **BSV decoder uses the GKP-syndrome information** and second doing the same but with the **hexagonal-asymmetric lattice** $\mathcal{L}_{\hexagon,r}$ from Eq. (5.36) instead of $\mathcal{L}_r$. Let us briefly note where a single run of the Monte-Carlo simulation differs when one takes the syndrome information of the GKP code into account in the surface code decoder.

The error syndrome of the code $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L})$ is $s \in \mathcal{V}_{\mathcal{L}^\perp}$. Hence when there are $d^2 + (d-1)^2$ GKP-qubits – which are labelled by the tuple index (j,k) – then the GKP decoder yields the tensor product of $d^2 + (d-1)^2$ corrected qubit states and syndromes $s_{j,k} \in \mathcal{V}_{\mathcal{L}^\perp}$. The output of the GKP error correction on $d^2 + (d+1)^2$ qubits is

$$\bigotimes_{j,k} |\psi_{j,k}\rangle \otimes |s_{j,k}\rangle \in (\mathbb{C}^2 \otimes \mathcal{V}_{\mathcal{L}^\perp})^{\otimes d^2+(d-1)^2} \ .$$

Hence the algorithm using the GKP syndrome information in the BSV decoder only differs from that without using the information in the following step: Instead of computing the a priori error distribution π from (5.95) and using its $(d^2+(d-1)^2)$-fold tensor product as input

function MCWithSideInfo(σ^2, d, r)

Input: Variance σ^2, code distance d, asymmetry ratio r
Output: Residual logical Pauli error $\overline{L} \in \{\overline{I}, \overline{X}, \overline{Y}, \overline{Z}\}$

---

  for each qubit (j,k) **do**
    Sample displacement error vector $\nu_{j,k} \in \mathbb{R}^2$ according to $\mathsf{N}(\mathbf{0}, \sigma^2 I_2)$.
    Compute the (logical) Pauli error $E_{j,k} \in \{I, X, Y, Z\}$ from $\nu_{j,k}$ according to (5.13) for $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$ respectively $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_{\hexagon,r})$.
    Compute the syndrome $s_0 = s(\nu_{j,k}) = (z \quad y)^T$ according to Eq. (5.8).
    Compute the conditional distribution $\pi_{j,k} = (p_I^{s_0}, p_X^{s_0}, p_Y^{s_0}, p_Z^{s_0})$ according to (5.96) with probabilities from (5.76) (for $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$) respectively (5.80) (for $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_{\hexagon,r})$).
  end for
  Set $E_s \coloneqq \bigotimes_{j,k} E_{j,k}$.
  Use the BSV decoder with input $\prod_{j,k} \pi_{j,k}$ to compute an n-qubit Pauli correction C.
  Compute $\overline{L} = \arg\max_{\overline{L} \in \{\overline{I},\overline{X},\overline{Y},\overline{Z}\}} 1_{E_s C \in \overline{L}\mathcal{S}}$ (i.e., decide which coset $E_s C$ belongs to).
end function

Figure 5.19: Subroutine MCWithSideInfo Monte-Carlo-simulates one instance of the error-recovery process when GKP side information is used. It returns the residual logical error.

of the BSV decoder one computes the distribution $\pi_{j,k}$ conditioned on the GKP syndrome $s_{j,k}$ for every qubit (j,k). Subsequently, one uses the prior error distribution $\bigotimes_{j,k} \pi_{j,k}$ as input of the BSV decoder. Here for the qubit j,k one has probabilities

$$\pi_{j,k} = \pi^{s_{j,k}} = \left(p_I^{s_0}, p_X^{s_0}, p_Y^{s_0}, p_Z^{s_0}\right) = \left(p_{\overline{I}}^{s_0}, p_{\overline{Y}}^{s_0}, p_{\overline{Z}}^{s_0}, p_{\overline{X}}^{s_0}\right) \quad , \quad \text{where } s_0 = s_{j,k} \ , \tag{5.96}$$

and where $p_{\overline{L}}^{s_0}$ are defined in the Eqs. (5.76), (5.78) and (5.80) for the GKP codes with lattices $\mathcal{L}_r$, $\mathcal{L}_{\hexagon}$, and $\mathcal{L}_{\hexagon,r}$, respectively.

5.5.2 Computation of threshold estimates

The Monte-Carlo simulations described above serve to numerically compute error thresholds of the considered surface-GKP codes. They return results for the functions $\sigma \mapsto P_{\mathsf{err}}(\sigma, d, r)$ for different values of the surface code distances d and asymmetry ratios r and seek to determine a threshold value σ_c for the physical noise σ below which P_{err} becomes arbitrarily small by increasing the surface code distance d and above which P_{err} converges to one as $d \to \infty$. This threshold value $\sigma_c(r)$ may depend on the asymmetry ratio r. We assume that as $d \to \infty$ the resulting logical error probability P_{err} of the surface-GKP code takes the form

$$\lim_{d\to\infty} P_{\mathsf{err}}(\sigma, d, r) = \theta(\sigma - \sigma_c(r)) \ ,$$

of a step function θ with the step at the critical value $\sigma_c(r)$.

Note that the considered codes exhibit a threshold behaviour which can be easily seen as follows: first, recall that surface codes for i.i.d. stochastic Pauli noise admit a threshold and second the logical GKP-qubits – which constitute the 'physical' qubits of the surface code – experience stochastic Pauli noise $\mathcal{N}_\pi$ (cf. Eq. (3.5)) where the probability distribution π

depends on the physical noise strength σ as described in Eqs. (5.59), (5.60), (5.95), i.e., in a complicated manner.

To identify the error threshold σ_c of the modified asymmetric surface-GKP code for fixed asymmetry ratio r, one uses that it corresponds to the point where the family of curves $\{\sigma \mapsto P_{\mathsf{err}}(\sigma, d, r)\}_{d\in\mathbb{N}}$ intersect. Mathematically, we employ the critical exponent method from [170]: One first writes the correlation length as $\xi = (\sigma - \sigma_c)^{1/\mu}$, i.e., in terms of the threshold value σ_c and some critical exponent μ. Furthermore, one introduces the dimensionless parameter

$$x = d/\xi = (\sigma - \sigma_c) d^{1/\mu} \ , \tag{5.97}$$

where d is the surface code distance. The assumption of the critical exponent is that the logical error probability $P_{\mathsf{err}}(\sigma, d, r)$ may be written as a polynomial in x. In this work we fit the data to a second order polynomial in x (making the same fit as used for the surface code under biased noise by Tuckett et al. [158]) i.e., we write

$$P_{\mathsf{err}}(\sigma, d, r) \approx A + Bx + Cx^2 \tag{5.98}$$

for the fitting parameters A, B, C. The validity and limits of this scaling hypothesis (5.97) and (5.98) are discussed in [174]. The numerical fit has a statistical error; we give the square root of the mean square fitting error in the threshold values later.

5.5.3 Choice of parameters and necessary bond dimension

The parameters relevant for a single simulation are the physical noise strength σ, the surface code size d, and the GKP lattice asymmetry ratio r. Furthermore, two additional relevant parameters per tuple (σ, r, d) are the BSV decoder bond dimension χ and the number of Monte-Carlo simulations and the cutoff κ in the summation used to evaluate the expressions (5.59). Let us explain the choice of these parameters.

We run simulations for the following parameters (for the modified asymmetric surface-GKP code without and with using the GKP syndrome information).

- **Noise strength**: since prior work [169] has shown that the error threshold of the symmetric (standard) toric-GKP code is at $\sigma_c \approx 0.54$ without using side information and $\sigma_c \approx 0.61$ with the use of side information, we choose $\sigma \in [0.4, 0.7]$ in increments of 0.02. A higher resolution (increments of 0.01) is chosen for the data points around the estimated thresholds. These data points are used to compute the threshold estimates.
- **Surface code size**: we choose $d \in \{9, 13, 17, 21\}$. These code sizes are comparable to the ones previously used to study biased noise of the surface code using the BSV decoder in [158].
- **Asymmetry ratio**: we are interested in the symmetric case $r = 1.0$ as well as biases towards Y-noise which would correspond to $r > 1.0$. As $r = 4.0$ already stands for a relatively high bias (for $r = 4.0$ and $\sigma = 0.54$ the noise bias from Eq. (5.86) is approximately $\eta = p_Y/(p_X + p_Z) \approx 386$), we expect that this is sufficient for our purposes. We choose an asymmetry ratio between one and four in increments of 0.5, i.e., $r \in \{1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0\}$.

The resulting error probability is simulated for all combinations of the above stated values for σ, d, r. In contrast for the other parameters, only a single value is chosen per tuple (σ, d, r).

More precisely, since we analyse the functions $\sigma \mapsto P_{\mathsf{err}}(\sigma, d, r)$, we choose a suitable bond dimension for every tuple (d, r) and similarly, the number of Monte-Carlo simulations and the cutoff are chosen once for each tuple (r, d).

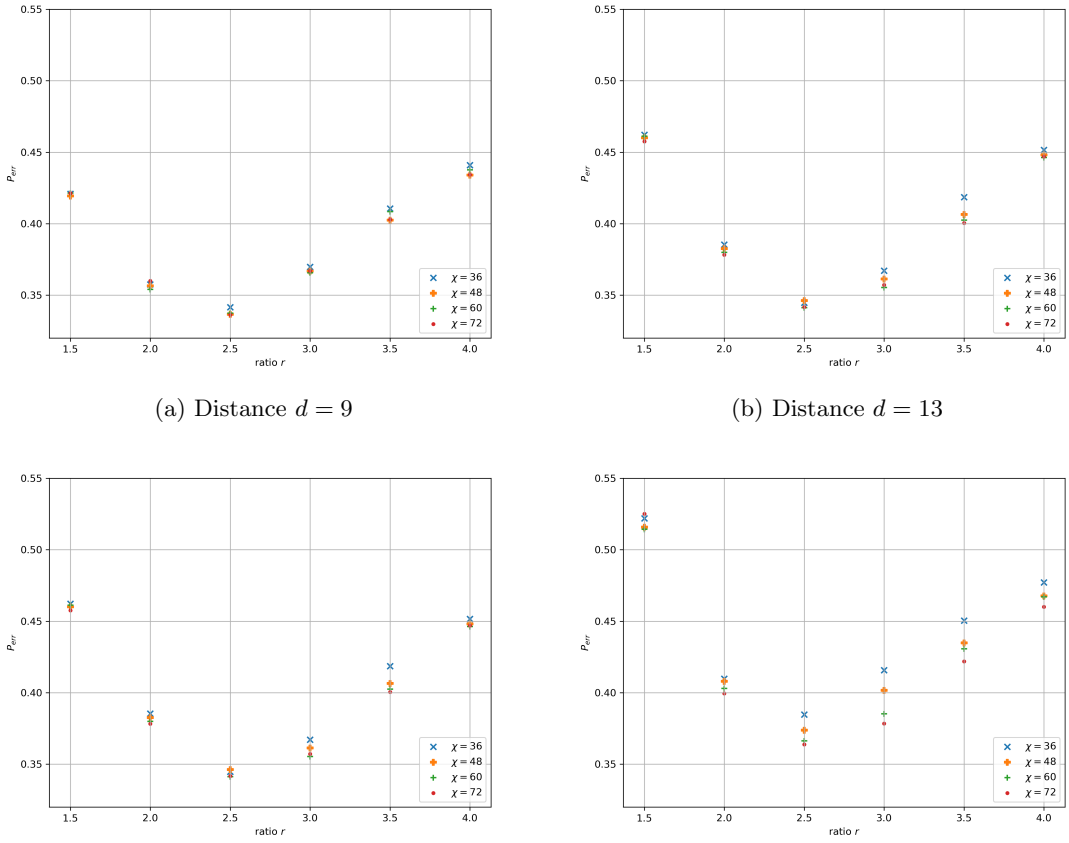
For the choice of these other parameters, it is important to note that the threshold value for the symmetric case should serve as a benchmark (obtained with a slightly different method than the known threshold for the surface-GKP code from the literature [68]). Since we compare the threshold values σ_c for different ratios r and investigate whether it increases with growing asymmetry, we compute the threshold value for $r = 1$ as exactly as possible (preventing any favour of our interpretation due to numerical inaccuracies). This is why, for the symmetric case, we use the highest bond dimension that we employ in our algorithms, and we use a higher cutoff. Note that if the BSV decoder receives as an input inaccurate approximation of the actual a priori error probabilities it still defines a valid decoder which of course performs worse than the maximum likelihood decoder. As a consequence, the conducted simulations – using approximations in the cutoff and with a finite bond dimension – compute the logical failure probability of some valid but a non-ideal decoder for the surface-GKP code, i.e., the numerically computed threshold is a lower bound on the achievable error thresholds.

- **Bond dimension:** the bond dimension should be chosen sufficiently large such that the BSV decoder gives a good approximation for the actual ML decoder. In order to numerically determine how well the algorithm has converged for a finite bond dimension, we simulate $P_{\mathsf{err}}(\sigma, d, r)$ for $\sigma = 0.58$ (which is close to the expected threshold value) with different bond dimensions. The results (cf. Fig. 5.20) indicate that the necessary bond dimension seems to primarily depend on the distance such that we choose $\chi = 48$ for $d = 9$, $\chi = 60$ for $d = 13$, $\chi = 72$ for $d = 17$ and $\chi = 100$ for $d = 21$ for all $r > 1$. Since the symmetric case should serve as a benchmark, we choose the highest bond dimension $\chi = 100$ for all distances in the symmetric case $r = 1$. For comparison, note that the bond dimensions by Fukui et al [68] are between 5 and 15.
  In the other scenarios, i.e., when the GKP-side information is used and when the hexagonal GKP lattice is considered, the simulations appear to converge faster and the same bond dimensions expected to be sufficient. Hence we choose the same χ for the tuples (d, r) as above.
- **Number of simulations:** the number of simulations that we use depends on the distance d, similar to the bond dimension. We use 50,000, 30,000, 20,000 and 10,000 simulations for $d = 9, 13, 17, 21$, respectively. Similar as for the bond dimension, we choose the highest available number of simulations $50,000$ for the reference value in the symmetric case $r = 1$.
- **Cutoff:** we choose a cutoff of $\kappa = 10$ for all asymmetric cases whereas the higher value of $\kappa = 15$ for is chosen for $r = 1$ as explained in more detail in Section 5.5.4.

Since the accuracy depends on the bond dimension, we choose the highest bond dimension $\chi = 100$ for the symmetric case, yielding the most accurate threshold estimate.

5.5.4 Cutoff analysis

The expressions $q_{\overline{X}}$ and $q_{\overline{Z}}$ from Eqs. (5.59) and (5.60) contain infinite sums. Hence – when computed in practice such as in the algorithms 5.18 and 5.19 – one may only compute



(a) Distance $d = 9$

(b) Distance $d = 13$

(c) Distance $d = 17$

(d) Distance $d = 21$

Figure 5.20: Estimation of the necessary bond dimension of the BSV decoder for the surface-GKP decoding without use of the GKP syndrome information. Each plot shows the logical error probability of the modified asymmetric surface-GKP code (without side information) for the bond dimensions $\chi \in \{36, 48, 60, 72\}$ and asymmetry ratios $r \in \{1.5, 2.0, 2.5, 3.0, 3.5, 4.0\}$ and for the noise strength $\sigma = 0.58$ and a different distance d. If the data points for a ratio agree, then the algorithm seems to have converged. The necessary bond dimension appears to depend on the distance.

approximations of them. Note that these quantities enter the algorithms as inputs of the BSV decoder, more precisely in the probability distributions π and $\pi_{j,k}$, respectively.

Let us introduce a cutoff $\kappa \in \mathbb{N}$ for the summation in Eqs. (5.59) and (5.60). This cutoff implies that all summands with indices $\geq \kappa$ and $\leq -\kappa$ are neglected in the computation of $q_{\overline{X}}$ and $q_{\overline{Z}}$. In contrast to the full quantities $q_{\overline{X}}$, $q_{\overline{Z}}$ and π with infinite summation, we denote their approximations with cutoff κ using a tilde symbol, i.e., by $\widetilde{q}_{\overline{X}}$, $\widetilde{q}_{\overline{Z}}$, $\widetilde{\pi}$ and $\widetilde{\pi}_{j,k}$. Let us estimate the relative error introduced by the cutoff $\kappa > 10$ compared to using the originally infinite sums. In order to do so, we introduce the two quantities

$$\alpha_+ \coloneqq \max_{r,\sigma} \left\{ \sqrt{2\pi r/\sigma^2}, \sqrt{2\pi/(r\sigma^2)} \right\} \quad , \quad \alpha_- \coloneqq \min_{r,\sigma} \left\{ \sqrt{2\pi r/\sigma^2}, \sqrt{2\pi/(r\sigma^2)} \right\} ,$$

where we maximise and the minimise, respectively, over the parameter regimes for σ and r that we consider in the simulations. Note that the introduction of a cutoff κ approximates the BSV decoder such that even in the limit of infinitely high bond dimension $\chi \to \infty$, it does not correspond to the ML decoder. But we still simulate some decoder which in contrast to the

ML decoder does not maximise the success probability. Note that the quantity $\widetilde{q}_{\overline{L}}$ is defined by Eq. (5.59) respectively (5.60) but with an adapted summation from $m \geq \kappa$ and $m \leq -\kappa$ instead of $m \in \mathbb{Z}$. Hence $\widetilde{q}_{\overline{L}} \leq q_{\overline{L}}$ and the error introduced by the cutoff is given by

$$\begin{aligned}\max_{\overline{L}\in\{\overline{X},\overline{Z}\}} (1-q_{\overline{L}}) - (1-\tilde{q}_{\overline{L}}) &\leq \frac{2}{\pi}\sum_{m=\kappa}^{\infty}\int_{\alpha_- m-\alpha_+/4}^{\alpha_- m+\alpha_+/4} e^{-\nu^2}d\nu \leq \frac{2}{\pi}\int_{\alpha_-\kappa-\alpha_+/4}^{\infty} e^{-\nu^2}d\nu \\ &\leq \frac{2}{\pi}\int_{\alpha_-\kappa-\alpha_+/4}^{\infty} \frac{\nu}{\alpha_-\kappa-\alpha_+/4} e^{-\nu^2}d\nu \ .\end{aligned}$$

In the first step, we used that the contribution of the summands decreases with increasing $m \in \mathbb{N}$ (since the error function from Eq. (5.61) becomes flatter as the absolute value grows) and that therefore, the quantity

$$\frac{1}{2}\big(\operatorname{erf}(\alpha_- m+\alpha_+/4) - \operatorname{erf}(\alpha_- n-\alpha_+/4)\big) = \frac{1}{\sqrt{\pi}}\int_{\alpha_- m-\alpha_+/4}^{\alpha_- n+\alpha_+/4} e^{-\nu^2}d\nu$$

is an upper bound on the increment of the nth summand in the sum. Furthermore, the quantity $1-q_{\overline{L}}$ can be bounded from below by its $m=0$ summand, i.e., by

$$1-q_{\overline{L}} \geq \operatorname{erf}(\alpha_-/4) \ ,$$

using that $\operatorname{erf}(-x) = -\operatorname{erf}(x)$ for all $x \in \mathbb{R}$. As a consequence, the relative error of the approximate values $1-\widetilde{q}_{\overline{L}}$ compared to the exact ones $1-q_{\overline{L}}$ are given by

$$\begin{aligned}\max_{\overline{L}\in\{\overline{X},\overline{Z}\}} \frac{(1-q_{\overline{L}}) - (1-\widetilde{q}_{\overline{L}})}{1-q_{\overline{L}}} &\leq \frac{2}{\pi\operatorname{erf}(\alpha_-/4)}\int_{\alpha_-\kappa-\alpha_+/4}^{\infty}\frac{\nu}{\alpha_-\kappa-\alpha_+/4}e^{-\nu^2}d\nu \\ &= \frac{e^{-(\alpha_-\kappa-\alpha_+/4)^2}}{\sqrt{\pi}(\alpha_-\kappa-\alpha_+/4)\operatorname{erf}(\alpha_-/4)}\end{aligned} \tag{5.99}$$

For the parameter regimes that we consider (i.e., $\sigma \in [0.4, 0.7]$ and $r \in [1.0, 4.0]$), one finds that $(\alpha_+, \alpha_-) = 5\sqrt{2\pi}(1, 1/7)$. Hence for $\kappa = 10$ the approximate values for $1-q_{\overline{L}}$ are within a $10^{-93}\%$ deviation of the exact ones; the relative error from Eq. (5.99) below $10^{-93}\%$. Since in the algorithms, we use the cutoff parameter $\kappa = 10$, the effect of the cutoff approximation on the algorithms' outcomes can be considered as negligible.

It is important for the analysis of asymmetry that the symmetric case yields an accurate reference value. Correspondingly, the associated threshold estimate and hence the numerically estimated logical error probabilities should be computed as exactly as possible. Therefore, the higher cutoff value $\kappa = 15$ is chosen for $r = 1$.

5.6 Numerical results for threshold estimates

This section answers the central research question of this chapter: Can one enhance the noise tolerance of surface-GKP codes by introducing asymmetry at the level of the GKP code in the form of an asymmetric lattice $\mathcal{L}_r$ for $r > 0$. In Section 5.6.1, we presents the results of our conducted numerical simulations as described in Section 5.5.1: an enhanced noise tolerance for asymmetry ratios up to $r \approx 3$. In Section 5.6.2, we discuss the observation, that the threshold values do not increase for larger asymmetry ratios and that the logical error probability of the concatenated code appears to be not-monotonic in this ratio. Section 5.6.3 presents the threshold estimates for other scenarios, i.e., when the GKP syndrome information is taken into account in the BSV decoder. Finally, Table 5.6.3 summarises our results for threshold estimates in the different scenarios and compares them to the ones obtained by other authors for the standard (symmetric) surface-(toric-)GKP code.

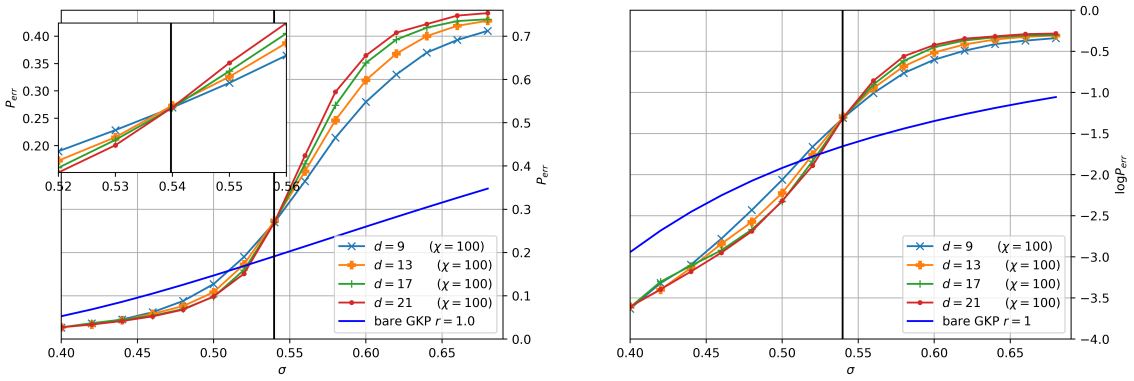
5.6.1 Enhanced noise tolerance in asymmetric surface-GKP codes

We evaluate the physical error threshold for the modified surface-GKP code using the simulated data. In order to find the value, we use the critical exponent method as described in the previous section. We start with the symmetric case, i.e., where $r = 1$. This threshold value serves as a point of reference and is compared to those for higher asymmetry ratios $r > 1$ in order to quantify the influence of asymmetry on the noise threshold. In Fig. 5.21, the logical failure probabilities of different surface-GKP codes are plotted as a function of the physical error strength, i.e., the standard deviation of the Gaussian displacement noise channel (the square root of the shift-error variance). Each subfigure represents a different asymmetry ratio.

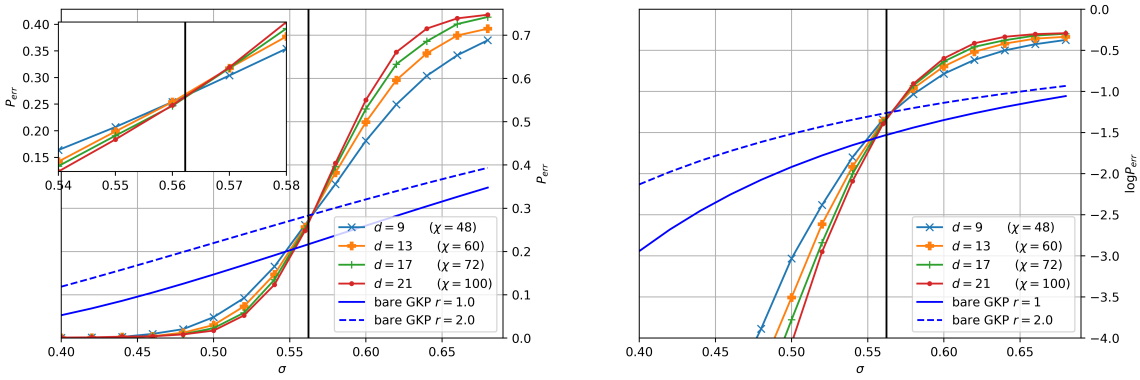
The threshold value for the physical noise strength σ in the symmetric case $r = 1$ is $\sigma_c \approx 0.540$, cf. Fig. 5.21a. The standard deviation (square root of the means square error) of this fitted threshold value (using the critical exponent method) is approximately 0.0006. As a general check of validity, let us compare this value to the noise threshold for the surface-GKP code numerically computed in the literature by other authors: Using the minimum weight matching decoder, Vuillot et al. [169] obtained a threshold value between 0.54 and 0.55 for the toric-GKP code. We note that Vuillot et al. considered the toric code instead of the square surface code and used a different decoder. Additionally, Vuillot et al. map the GKP errors differently to qubit errors (they use the standard GKP code, instead of what we call the *modification*, cf. Section 5.4.2). Furthermore, Fukui et al. [68] obtain $\sigma_c \approx 0.542$ for the surface-GKP code. Here, the authors consider the surface code (the same as considered here) but a different decoder for the surface code, a minimum-weight perfect-matching algorithm and similarly to Vuillot et al. no modification (in the sense of Section 5.4.2).

As one increases the asymmetry ratio, one finds noise thresholds for the displacement error standard deviation which are higher than $\sigma_c \approx 0.540$ (the reference value for $r = 1$). Fig. 5.22 shows that the threshold value σ_c grows monotonically from $r = 1.0$ up to $r = 3.0$, where it takes its maximal value $\sigma_c \approx 0.581$, and then reduces again for higher asymmetry ratios, i.e., to $\sigma_c \approx 0.568$ for $r = 4$.

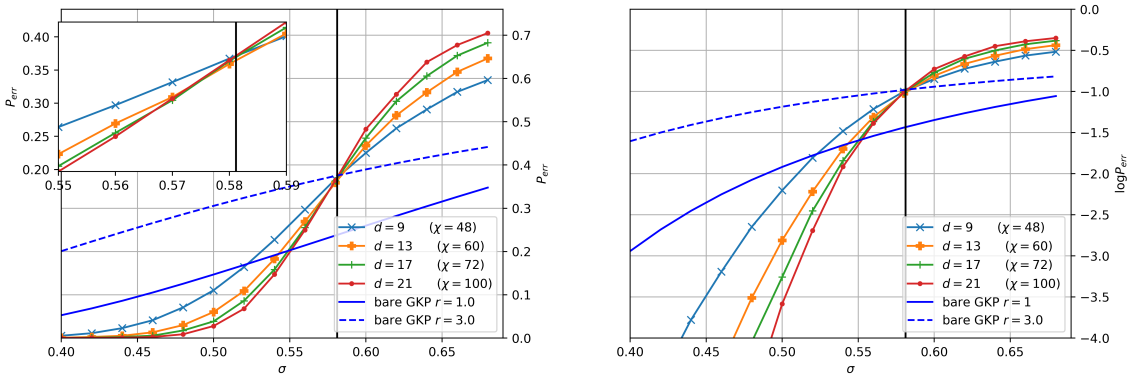
Since the variance σ^2 of the Gaussian displacement noise quantifies the physical noise at the level of the GKP modes, one may want to translate the threshold values into the corresponding noise strengths at the level of the logical GKP-qubits (i.e., the physical noise of the surface code). The threshold values $\sigma_c = 0.540$ and $\sigma_c \approx 0.581$ (for $r = 1.0$ and $r = 3.0$, respectively) correspond to (square) surface code probability of a Y-type error (occurring independently of an X-type error in the independent Y- and X-noise model) of 10.1% and 12.7%. More precisely, the latter percentages are found by computing $q_{\overline{X}}$ from Eq. (5.59) for the parameters $\sigma = \sigma_c$ and $r = 1$. Here, we regain the $\approx 10\%$ error threshold of the surface code for pure X (or equivalently Y- or Z-errors). The logical GKP-qubit error rates $p = p_X + p_Y + p_Z$ – i.e., the physical error rates of the surface code – depend on σ and r, i.e., let us write $p(\sigma_c, r)$. These rates at the threshold values σ_c grow with increasing asymmetry ratio r. Direct computation (employing Eq. (5.59) and Eq. (5.60)) shows that they range between $p(0.54, 1) \approx 0.19$ in the symmetric GKP code, $p(0.562, 2) \approx 0.28$ for the asymmetry ratio $r = 2$, $p(0.581, 3) \approx 0.38$ for $r = 3$ where we achieved the highest threshold value $\sigma_c \approx 0.5813$ up to $p(0.568, 4) \approx 0.42$ for $r = 4$. We note that the last value, the logical GKP-qubit error probability of 42% at $r = 4$, corresponds to a high noise bias $\eta = p_Y/(p_X + p_Z) \approx 230$ where we take η from Eq. (5.91) is close to the threshold value $p_c = 43.7(1)\%$ found for the surface code with the BSV decoder and the infinitely-biased (i.e., $\eta \to \infty$) noise model $\pi = (1 - p_Y(1 + 1/\eta), p_Y/(2\eta), p_Y, p_Y/(2\eta))$ from [158].



(a) $r = 1.0$, threshold value $\sigma_c \approx 0.540$



(b) $r = 2.0$, threshold value $\sigma_c \approx 0.562$



(c) $r = 3.0$, threshold value $\sigma_c \approx 0.581$ (fittig standard deviation 0.0019)

Figure 5.21: The error probability $P_{\mathsf{err}}(\sigma, d, r)$ as a function of the physical noise strength σ (left) for decoding without GKP side information for different asymmetry ratios $r \in \{1.0, 2.0, 3.0\}$. Insets give higher-resolution data around the observed threshold estimate σ_c for the critical noise standard deviation. The latter is indicated by a vertical line. The right hand side shows a log-plot of the error probability $P_{\mathsf{err}}(\sigma, d, r)$. For comparison, the error probability of the bare GKP code, i.e., without concatenation with the surface code, is shown in blue with the solid line corresponding to ratio $r = 1.0$ and the dashed lines corresponding to the respective ratio of the figure. *(Figure directly taken from the article [76])*

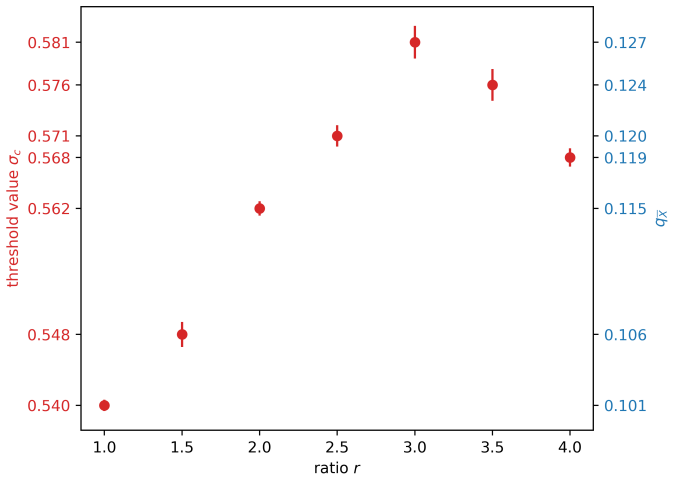


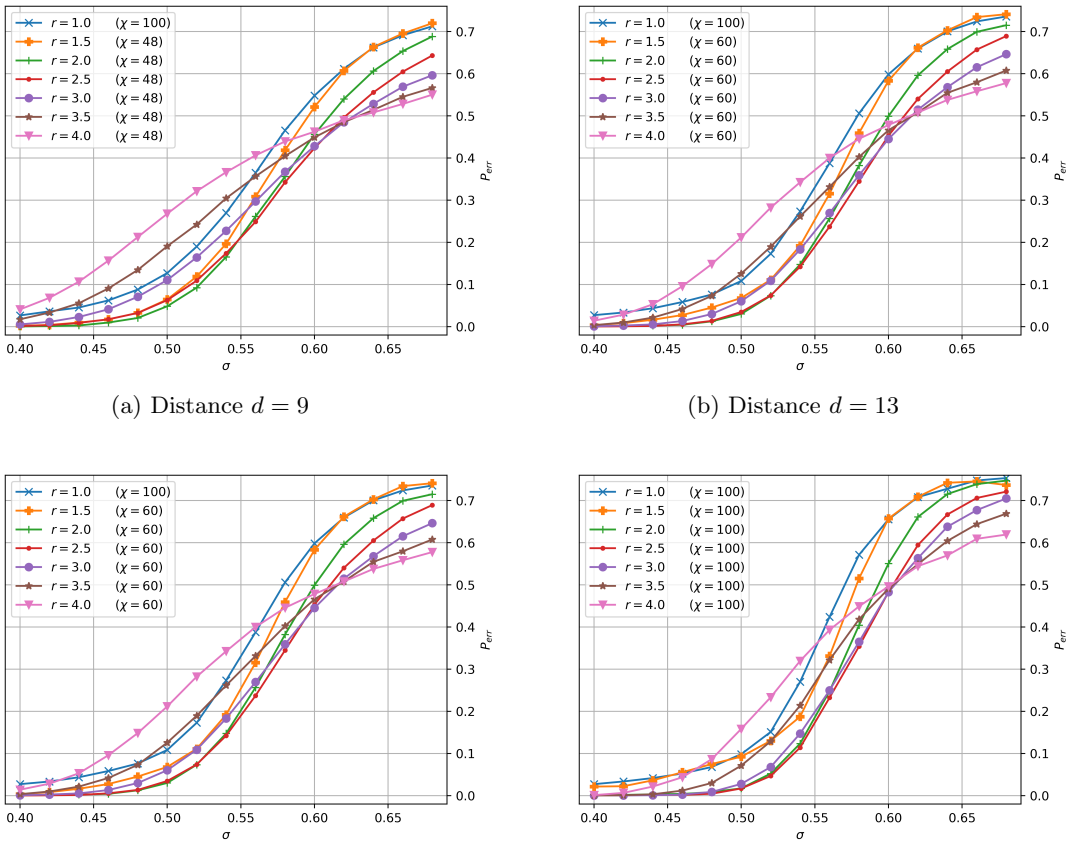
Figure 5.22: Empirically computed thresholds σ_c of the modified asymmetric surface-GKP code for different asymmetry ratios $r \in \{1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0\}$. The error bars depict the standard deviation (the square root of the means squared error of the employed fitting procedure) of the fitted threshold value σ_c (left hand side axis) and range from 0.0006 (for $r = 1$) to 0.0019 (for $r = 3$). The values on the right hand side axis correspond to $q_{\overline{X}}$ (from Eq. (5.59)) for the parameters $\sigma = \sigma_c$ and $r = 1$. *(Figure directly taken from the article [76])*

This shows that the noise resilience of the surface code towards biased noise can be used to enhance noise tolerance of the concatenated surface-GKP code. This comes at a cost of an additional encoding step, the modified asymmetric encoding, which involves squeezing of $\log(r)$ dB.

5.6.2 Monotonicity of the logical error probability

In addition to the threshold value, it might also be interesting to analyse how the logical error probability changes as the asymmetry ratio grows. Let us investigate the question whether for fixed σ and d, the logical error probability $P_{\mathsf{err}}(\sigma, d, r)$ decreases monotonically if the asymmetry ratio r increases. If this were the case, an increasing asymmetry would improve the noise resilience of the modified asymmetric surface-GKP code for all code sizes and over the whole range of physical noise variances σ^2.

The answer to this question is a priori not clear since two qualitatively opposing effects play a role, here: On the one hand, by increasing the asymmetry, the logical failure probability of the bare GKP code $\mathsf{GKP}(\mathcal{L}_r)$ increases as well. By 'bare GKP code' we mean the GKP code without concatenation with the surface code. This behaviour is visualised in Fig. 5.21 where the blue curves give the logical error probability of the bare GKP code (without concatenation with the surface code): More precisely, the blue solid lines stand for the modified symmetric GKP code $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_\square)$ and the dashed blue lines are those of the code $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_r)$ for the asymmetry ratio $r > 1$ of the respective subfigure. The solid line lies below the dashed blue lines in each of the subfigures, i.e., increasing the asymmetry of the bare (modified) asymmetric GKP code decreases its noise tolerance, cf. Fig. 5.21b and 5.21c. On the other hand, increasing the asymmetry ratio implies an increasingly biased independent Y- and Z-noise at the surface code level, a noise model to which the surface code is more resilient. Whereas the former effect suggests an increased logical failure probability $P_{\mathsf{err}}(\sigma, d, r)$ of the concatenated code with increasing asymmetry, this latter suggests that it decreases. Hence



(a) Distance $d = 9$

(b) Distance $d = 13$

(c) Distance $d = 17$

(d) Distance $d = 21$

Figure 5.23: The logical error probability $P_{\mathsf{err}}(\sigma, d, r)$ of the modified asymmetric surface-GKP code where decoding of the surface code is conducted without the GKP side information. Each subfigure presents the data for a code distance $d \in \{9, 13, 17, 21\}$. The families of curves $\{\sigma \mapsto P_{\mathsf{err}}(\sigma, d, r)\}_{r \in \{1.0,1.5,2.5,3.0,3.5,4.0\}}$ for the four different distances suggest that for fixed σ the monotonicity of the function $r \mapsto P_{\mathsf{err}}(\sigma, d, r)$ appears to depend on the code size d. For small code sizes, this monotonicity only applies for small $r < 2$, whereas for $d = 21$, it applies up to $r = 2.5$. *(Figure directly taken from the article [76])*

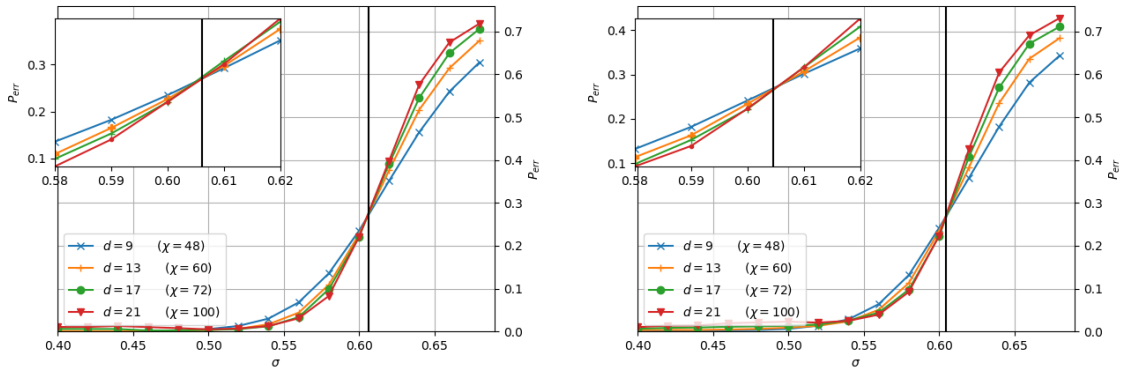
it is a priori unclear which of these two effects is dominant for which parameter ranges.

Numerically, this question may be answered by the conducted simulations. The monotonicity property appears to be distance-dependent. Comparing the curves for $d = 9$ and $d = 21$ in Fig. 5.23. For $d = 9$, one has

$$P_{\mathsf{err}}(\sigma, d = 9, r = 2.0) < P_{\mathsf{err}}(\sigma, d = 9, r = 1.5) < P_{\mathsf{err}}(\sigma, d = 9, r = 1.0) \tag{5.100}$$

for all considered values of σ, indicating an improved noise tolerance by increasing the asymmetry up to $r = 2$ throughout the whole range of noise strengths σ. But this does not extend to higher ratios. For $r \geq 2.5$ there are values $\sigma \in [0.4, 0.7]$ such that $P_{\mathsf{err}}(\sigma, d = 9, r' = 3.0) > P_{\mathsf{err}}(\sigma, d = 9, r)$ where $r' > r$. This shows that for $d = 9$, the noise tolerance improves for all noise strengths by increasing r up to $r = 2$ but it does not improve further for higher asymmetry ratios. In contrast, for $d = 21$, this improvement is achieved also for the higher ratio $r = 2.5$. The curves in Fig. 5.23 show that the behaviour from Eq. (5.100) extends to $d = 21$ and that additionally

$$P_{\mathsf{err}}(\sigma, d = 21, r = 2.5) < P_{\mathsf{err}}(\sigma, d = 21, r = 2.0)$$



(a) Rectangular lattice GKP with side information, threshold value $\sigma_c \approx 0.6062$ (with fitting standard deviation 0.0007)

(b) Asymmetric hexagonal lattice GKP with side information, threshold value $\sigma_c \approx 0.6045$ (with fitting standard deviation 0.0009)

Figure 5.24: Other improvements for asymmetry ratio $r = 2.0$: Here the plots show the logical error probability $P_{\mathsf{err}}(\sigma, d, r = 2)$ as a function of σ for decoding with side information. Fig. 5.24a shows the error probability for the rectangular lattice GKP code whereas Fig. 5.24b for the asymmetric hexagonal lattice GKP code. Insets mark the observed threshold estimates σ_c are indicated with vertical lines.

for all considered values of σ. Similarly, for $d = 9$, there are values $\sigma \in [0.4, 0.7]$ such that $P_{\mathsf{err}}(\sigma, d = 9, r = 3.5) > P_{\mathsf{err}}(\sigma, d = 9, r = 1.0)$ whereas for $d = 21$, $P_{\mathsf{err}}(\sigma, d = 21, r = 3.5) < P_{\mathsf{err}}(\sigma, d = 21, r = 1.0)$ for all values of σ. As a consequence, for $d = 9$, the noise tolerance for $r = 3.5$ is not higher than that for $r = 1.0$ whereas for distance $d = 21$ it is.

As discussed above, a potential reason for this behaviour is the reduced noise tolerance of the bare asymmetric GKP code compared to the symmetric GKP code. Another reason for the non-monotonic behaviour might be finite size effects. One typically observes finite-size effects when comparing lower to higher distances. This effect results in the observation that the curves of the logical error probability for different distances do not meet in the exact same point and that the computed threshold estimate decreases when larger distances are taken into account. In our analysis, the behaviour that for higher ratios ($r = 3.0$, $r = 3.5$ and $r = 4.0$) the curves for different distances do not cross at the same point – cf. e.g. in the insets in Fig. 5.21c – can be explained by finite-size effects. Finite-size effects have been discussed for the surface code, e.g. by Tuckett et al. [158], and as well for the concatenated surface-GKP code, e.g. by Fukui et al. [68], in the literature. In the former, Tuckett et al. [158] observe finite-size effects for QEC in the square surface code using the BSV decoder: the curves (of the logical error probability for different distances) do not meet at the exact same point, especially for higher noise biases η, and consequently, the error threshold does not appear to grow up to $p_c = 50\%$ – but rather up to $p_c = 43.7\%$ – although it should by analytical computation. Note that these authors do see a similar behaviour but slightly different error threshold in the rotated surface code [160] – the surface code lattice is rotated by an angle of 45° compared to the standard square surface code – finding an error threshold of $p_c = 45.4(2)\%$ at infinitely high noise bias and hence less severe finite-size effects. Fukui et al. [68] observe larger finite-size effects in the analogue error correction (using no syndrome side information) compared to the one with syndrome side information.

5.6.3 Results for further scenarios

(a) Threshold values of the modified asymmetric surface-GKP code

			GKP code $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L})$							
			$\mathcal{L}_{\square}$	$\mathcal{L}_{1.5}$	$\mathcal{L}_{2}$	$\mathcal{L}_{2.5}$	$\mathcal{L}_{3}$	$\mathcal{L}_{3.5}$	$\mathcal{L}_{4}$	$\mathcal{L}_{\hexagon,2}$
	surface	NoS	0.540	0.548	0.562	0.571	0.581	0.576	0.568	–
	BSV	+S	–	–	0.606	–	–	–	–	0.605

(b) Literature

			GKP code $\mathsf{GKP}(\mathcal{L})$							
			$\mathcal{L}_{\square}$	$\mathcal{L}_{1.5}$	$\mathcal{L}_{2}$	$\mathcal{L}_{2.5}$	$\mathcal{L}_{3}$	$\mathcal{L}_{3.5}$	$\mathcal{L}_{4}$	$\mathcal{L}_{\hexagon,2}$
[68]	surface	NoS	0.542	–	–	–	–	–	–	–
	MWM	+S	0.607	–	–	–	–	–	–	–
[169]	toric	NoS	0.54-0.55	–	–	–	–	–	–	–
	MWM	+S	0.60	–	–	–	–	–	–	–

Table 5.1: Comparison of the error thresholds σ_c for different concatenated surface-GKP codes against random Gaussian displacement noise with shift-error standard deviation σ. The upper table (a) presents the results of our work [76] whereas the lower one (b) those of Fukui et al. [68] and Vuillot et al. [169] for comparison. The rows stand for different surface/toric decoders and the columns for different GKP codes. We abbreviate certain decoders: i.e., BSV stands for the tensor network decoder by Bravyi, Suchara, and Vargo [29] , (that we employ for surface code decoding, cf. 5.3.3), 'MWM' for a minimum weight matching decoder used in [68, 169]; NoS and +S indicate that the corresponding surface/toric code decoder is performed without or with the use of the GKP syndrome information, respectively (NoS for "no syndrome" and +S for "with syndrome"). When the GKP-syndrome information is neglected, the BSV decoder uses the same a priori noise distribution $\prod_{j,k}\pi$ for π from Eq. (5.95) on all qubits j,k as input, whereas in the case when the syndrome information is taken into account, the BSV decoder is conditioned on $\prod_{j,k}\pi(s_{j,k})$ for $\pi_{jk}=\pi(s_{j,k})$ from Eq. (5.96). The comparison shows that the highest threshold estimates are obtained when the GKP-syndrome information is passed to the surface/toric code decoder. But when the syndrome information is neglected, our asymmetric codes outperform the previously studied symmetric codes.

Let us compare the above results to other scenarios. There are several potential improvements on the code which we did not consider yet.

First, one might use a different decoder for the surface code. One may use the syndrome information of the GKP decoding as an input of the BSV decoder in order to improve its success probability as described in Section 5.3.3. To understand whether this leads to an improvement we simulate the surface-GKP error correction procedure when additionally using this side information as an input of the BSV decoder. Fig. 5.24 shows the logical error probability of the surface-GKP code when using the asymmetric GKP lattice $\mathcal{L}_{r=2}$. The error threshold is at $\sigma_c \approx 0.6062 \pm 0.0007$. This shows that this further improvements by using GKP-syndrome information are compatible with the ones from increasing the asymmetry. This amount of noise tolerance is also comparable with previous results, since for the symmetric case, it was shown that using the GKP-syndrome information leads to an error threshold improvement. Fukui et al. [68] numerically found that for the surface-GKP code, the error threshold achieves about $\sigma \approx 0.607$, a value close to the Hashing bound for the quantum capacity of the Gaussian channel (they refer to taking the syndrome information into account as analog QEC compared to digital QEC when not taking it into account).

Vuillot et al. [169] considered the (symmetric) toric-GKP code with a different surface code decoder as explained before. They found a threshold value of $\sigma_c \approx 0.60$ (in the symmetric case) when taking the GKP-syndrome information into account.

One further parameter that one might consider is the angle of the underlying GKP-lattice. So far, the analysis is restricted to rectangular GKP lattices since this gives a simple proof of principle. This also renders the effective GKP-qubit noise to be an independent Y- and Z-noise which is no longer true if the lattice is not rectangular. As already discussed in the initial paper [71] by Gottesman, Kitaev and Preskill, the optimal lattice of a single GKP-qubit was shown to be the hexagonal lattice $\mathcal{L}_{\hexagon}$ since it allows for the densest packing of an equally sized unit (i.e., Voronoi) cells in the two-dimensional phase space. Here, we simulate the surface-GKP error correction process for the GKP code $\mathsf{GKP}_{\mathsf{mod}}(\mathcal{L}_{\hexagon,r=2})$, which we defined by the squeezed hexagonal lattice (5.36), and where we use the GKP syndrome information in the BSV decoding process. We find the threshold value $\sigma_c \approx 0.6045 \pm 0.0009$, cf. Fig. 5.24b what is comparable to the one for the rectangular surface-GKP code with syndrome side information.

6 Conclusion

This thesis presents new schemes for protecting quantum information encoded in continuous-variable systems. Since such systems are ubiquitous in physics and their manipulation is experimentally well-explored, they provide a natural platform for quantum information processing. Basic fault-tolerance mechanisms, such as those identified in this work, can facilitate the construction of robust information-processing protocols.

Two complementary approaches are pursued in this thesis: The first one – dynamical decoupling (DD) – is motivated by control theory, and seeks to reduce physical error rates by making optimal use of available capabilities. Pioneered in the context of NMR [74, 73, 175], DD is now a well-established theoretical tool and has proven beneficial in finite-dimensional systems [167, 161]. Motivated by this and earlier work on similar pulse sequences in CV systems [12], we have shown how to lift general qubit-based protocols to the bosonic setting. Specifically, we construct DD pulse sequences which effectively suppress system-bath interactions to any desired order. Additionally, the system's evolution is converted to that of uncoupled harmonic oscillators rotating with identical frequencies. In contrast to the qubit setting, the effective evolution of a CV system with a general quadratic Hamiltonians cannot be rendered trivial [12]: the achieved homogenised evolution is optimal in this sense. A distinguishing feature of our protocols is their simplicity, which makes them attractive for experimental realisation: The pulse sequences are efficient, requiring only a polynomial number of pulses in the desired suppression order. Furthermore, the individual pulses are realisable by passive linear optics: two-mode beam splitters and single-mode phase space rotations are sufficient. The resulting homogenised evolution creates natural decoherence-free subspaces in which quantum information can be encoded.

The second approach towards fault-tolerance pursued in this thesis follows the standard route of error-correcting codes. As in the case of DD, the corresponding theory is better developed for qubit-based codes: with the exception of a handful of examples (such as the GKP code [71], the cat code [39, 105, 115] and binomial codes [114]), the study of the potential of CV systems for error correction has been limited. Historically, this may be a consequence of several no-go results showing that CV quantum information cannot be protected against noise using Gaussian resources only [69, 56, 123]. However, these results do not apply to scenarios where non-Gaussian resources such as GKP states are employed. For example, recent work [68, 169] shows that concatenated codes obtained by combining the surface/toric code with the GKP code achieve error thresholds for the storage of logical qubits in CV systems. This thesis builds on these developments and shows that a modification of the encoding procedure leads to an improvement of error thresholds. The basic idea is to use an asymmetric encoding map that effectively biases the noise at the logical GKP-qubit level. This approach allows to exploit the resilience of the surface code to biased noise previously observed in [158, 160]. Our threshold estimates are obtained by using the BSV-decoder [29] and numerically simulating the process of encoding, noise, and error correction using a Monte

Carlo algorithm. We remark that throughout, we consider an idealised setup where no errors occur during the syndrome measurements of error correction process. Similar ideas making use of noise-tolerance of certain codes against biased noise have also been explored, for example in [135].

While these two approaches have different objectives, they can be combined in principle: DD can lower the physical error rate below the threshold, i.e., help to reach a regime where the surface-GKP code provides more reliable logical qubits. This raises interesting new questions for future work. For example, it is currently not clear to what extent the surface-GKP code protects against more general noise than random phase space displacements. Obtaining quantitative estimates of corresponding error thresholds will require new analytical ideas, as well as/or new efficient simulation algorithms.

Both for DD and error correction, our work showcases how formal similarities between qubit- and CV systems translate to direct relationships between corresponding information-processing protocols. In the case of DD, the corresponding "lifting" procedure, which converts a qubit DD-pulse sequence to a CV-scheme, exemplifies a general connection between the Lie groups and algebras describing multi-qubit systems and those of bosonic systems whose dynamics is generated by quadratic Hamiltonians. It is conceivable that this mapping can be exploited to extend other quantum information processing protocols to the CV setting. In the case of surface-GKP codes, the particular choice of encoding permits to use specific features of qubit codes in an advantageous way in the context of CV codes. Importantly, this phenomenon crucially relies on the simultaneous consideration of CV- and DV properties: A similar effect could not be obtained using CV-to-CV-encodings only. This improvement is obtained by an apparently minor modification of the encoding. This illustrates the importance of studying CV- and DV-systems in tandem to fully explore the potential of a given quantum device.

Bibliography

[1] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In *Proceedings of the forty-third Annual ACM Symposium on Theory of Computing*, pages 333–342, 2011.

[2] D. Aharonov and M. Ben-Or. Fault-tolerant quantum computation with constant error. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing*, page 176–188, New York, NY, USA, 1997. Association for Computing Machinery.

[3] Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error rate. *SIAM Journal on Computing*, 38(4):1207–1282, 2008. arXiv:quant-ph/990612.

[4] Victor V Albert, Kyungjoo Noh, Kasper Duivenvoorden, Dylan J Young, RT Brierley, Philip Reinhold, Christophe Vuillot, Linshu Li, Chao Shen, Steven M Girvin, et al. Performance and structure of single-mode bosonic codes. *Physical Review A*, 97(3):032346, 2018.

[5] Francesca Albertini and Domenico D'Alessandro. Notions of controllability for bilinear multilevel quantum systems. *IEEE Transactions on Automatic Control*, 48(8):1399–1403, 2003.

[6] Panos Aliferis. Level reduction and the quantum threshold theorem. *arXiv:quant-ph/0703230*, 2007.

[7] Panos Aliferis, Frederico Brito, David P DiVincenzo, John Preskill, Matthias Steffen, and Barbara M Terhal. Fault-tolerant computing with biased-noise superconducting qubits: a case study. *New Journal of Physics*, 11(1):013061, 2009.

[8] Panos Aliferis, Daniel Gottesman, and John Preskill. Quantum accuracy threshold for concatenated distance-3 codes. *Quantum Information and Computation*, 6(2):97–165, 2006.

[9] Panos Aliferis and John Preskill. Fault-tolerant quantum computation against biased noise. *Physical Review A*, 78:052331, 2008.

[10] Takao Aoki, Go Takahashi, Tadashi Kajiya, Jun-ichi Yoshikawa, Samuel L Braunstein, Peter Van Loock, and Akira Furusawa. Quantum error correction beyond qubits. *Nature Physics*, 5(8):541–546, 2009.

[11] Christian Arenz, Daniel Burgarth, Paolo Facchi, and Robin Hillier. Dynamical decoupling of unbounded Hamiltonians. *Journal of Mathematical Physics*, 59(3):032203, 2018.

[12] Christian Arenz, Daniel Burgarth, and Robin Hillier. Dynamical decoupling and homogenization of continuous variable systems. *Journal of Physics A: Mathematical and Theoretical*, 50(13):135303, 2017.

[13] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[14] Nandor L Balazs and André Voros. The quantized Baker's transformation. *Annals of Physics*, 190(1):1–31, 1989.

[15] Ben Q Baragiola, Giacomo Pantaleoni, Rafael N Alexander, Angela Karanjai, and Nicolas C Menicucci. All-gaussian universality and fault tolerance with the gottesman-kitaev-preskill code. *Physical Review Letters*, 123(20):200502, 2019.

[16] Rami Barends, Julian Kelly, Anthony Megrant, Andrzej Veitia, Daniel Sank, Evan Jeffrey, Ted C White, Josh Mutus, Austin G Fowler, Brooks Campbell, et al. Superconducting quantum circuits at the surface code threshold for fault tolerance. *Nature*, 508(7497):500–503, 2014.

[17] Christian Barthel, James Medford, Charles M Marcus, Micah P Hanson, and Arthur C Gossard. Interlaced dynamical decoupling and coherent operation of a singlet-triplet qubit. *Physical Review Letters*, 105(26):266808, 2010.

[18] Paul Benioff. The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of Statistical Physics*, 22(5):563–591, 1980.

[19] Charles H Bennett, David P DiVincenzo, John A Smolin, and William K Wootters. Mixed-state entanglement and quantum error correction. *Physical Review A*, 54(5):3824, 1996.

[20] Michael J Biercuk, Hermann Uys, Aaron P VanDevender, Nobuyasu Shiga, Wayne M Itano, and John J Bollinger. Experimental Uhrig dynamical decoupling using trapped ions. *Physical Review A*, 79(6):062324, 2009.

[21] Michael J Biercuk, Hermann Uys, Aaron P VanDevender, Nobuyasu Shiga, Wayne M Itano, and John J Bollinger. Optimized dynamical decoupling in a model quantum memory. *Nature*, 458(7241):996–1000, 2009.

[22] Sergio Blanes, Fernando Casas, Jose-Angel Oteo, and José Ros. The magnus expansion and some of its applications. *Physics Reports*, 470(5-6):151–238, 2009.

[23] H. Bombin and M. A. Martin-Delgado. Topological quantum distillation. *Physical Review Letters*, 97:180501, 2006.

[24] A Bouzouina and Stephan De Bievre. Equipartition of the eigenfunctions of quantized ergodic maps on the torus. *Communications in Mathematical Physics*, 178(1):83–105, 1996.

[25] P Oscar Boykin, Tal Mor, Matthew Pulver, Vwani Roychowdhury, and Farrokh Vatan. A new universal and fault-tolerant quantum basis. *Information Processing Letters*, 75(3):101–107, 2000.

[26] Ola Bratteli and Derek William Robinson. *Operator Algebras and Quantum Statistical Mechanics 1: C*-and W*-Algebras. Symmetry Groups. Decomposition of States.* Operator Algebras and Quantum Statistical Mechanics. Springer Science & Business Media, 1987.

[27] Samuel L Braunstein. Error correction for continuous quantum variables. *Physical Review Letters*, 80:4084–4087, 1998.

[28] Samuel L Braunstein. Quantum error correction for communication with linear optics. *Nature*, 394(6688):47–49, 1998.

[29] Sergey Bravyi, Martin Suchara, and Alexander Vargo. Efficient algorithms for maximum likelihood decoding in the surface code. *Physical Review A*, 90:032326, 2014.

[30] Sergey B Bravyi and Alexei Yu Kitaev. Quantum codes on a lattice with boundary. *arXiv:quant-ph/9811052*, 1998.

[31] Jonas Bylander, Simon Gustavsson, Fei Yan, Fumiki Yoshihara, Khalil Harrabi, George Fitch, David G Cory, Yasunobu Nakamura, Jaw-Shen Tsai, and William D Oliver. Noise spectroscopy through dynamical decoupling with a superconducting flux qubit. *Nature Physics*, 7(7):565–570, 2011.

[32] A Robert Calderbank and Peter W Shor. Good quantum error-correcting codes exist. *Physical Review A*, 54(2):1098, 1996.

[33] Philippe Campagne-Ibarcq, Alec Eickbusch, Steven Touzard, Evan Zalys-Geller, Nicholas E Frattini, Volodymyr V Sivak, Philip Reinhold, Shruti Puri, Shyam Shankar, Robert J Schoelkopf, et al. Quantum error correction of a qubit encoded in grid states of an oscillator. *Nature*, 584(7821):368–372, 2020.

[34] Herman Y Carr and Edward M Purcell. Effects of diffusion on free precession in nuclear magnetic resonance experiments. *Physical Review*, 94:630–638, 1954.

[35] Filippo Caruso, Jens Eisert, Vittorio Giovannetti, and Alexander S Holevo. Multi-mode bosonic gaussian channels. *New Journal of Physics*, 10(8):083030, 2008.

[36] Filippo Caruso, Jens Eisert, Vittorio Giovannetti, and Alexander S Holevo. Optimal unitary dilation for bosonic gaussian channels. *Physical Review A*, 84(2):022306, 2011.

[37] Nicolas J Cerf, Gerd Leuchs, and Eugene S Polzik. *Quantum Information with Continuous Variables of Atoms and Light.* Imperial College Press (distributed by World Scientific Publishing Co.), 2007.

[38] Christopher Chamberland, Tomas Jochym-O'Connor, and Raymond Laflamme. Thresholds for universal concatenated quantum codes. *Physical Review Letters*, 117(1):010501, 2016.

[39] Paul T Cochrane, Gerard J Milburn, and William J Munro. Macroscopically distinct quantum-superposition states as a bosonic code for amplitude damping. *Physical Review A*, 59(4):2631, 1999.

[40] Antonio D Córcoles, Easwar Magesan, Srikanth J Srinivasan, Andrew W Cross, Matthias Steffen, Jay M Gambetta, and Jerry M Chow. Demonstration of a quantum error detection code using a square lattice of four superconducting qubits. *Nature Communications*, 6(1):1–10, 2015.

[41] David G Cory, Raymondd Laflamme, Emanuel Knill, Lorenza Viola, Timothy F Havel, Nicolas Boulant, Gregory Boutis, Evan Fortunato, Seth Lloyd, R Martinez, et al. NMR based quantum information processing: achievements and prospects. *Fortschritte der Physik: Progress of Physics*, 48(9-11):875–907, 2000.

[42] Sajeev Damodarakurup, Marco Lucamarini, Giovanni Di Giuseppe, David Vitali, and Paolo Tombesi. Experimental inhibition of decoherence on flying qubits via "bang-bang" control. *Physical Review Letters*, 103(4):040502, 2009.

[43] Christoph Dankert, Richard Cleve, Joseph Emerson, and Etera Livine. Exact and approximate unitary 2-designs and their application to fidelity estimation. *Physical Review A*, 80(1):012304, 2009.

[44] Maurice A De Gosson. *Symplectic Geometry and Quantum Mechanics*, volume 166. Springer Science & Business Media, 2006.

[45] Gijs De Lange, ZH Wang, Diego Ristè, Viatcheslav V Dobrovitski, and Ronald Hanson. Universal dynamical decoupling of a single solid-state spin from a spin bath. *Science*, 330(6000):60–63, 2010.

[46] Brennan de Neeve, Thanh-Long Nguyen, Tanja Behrle, and Jonathan P Home. Error correction of a logical grid state qubit by dissipative pumping. *Nature Physics*, 18(3):296–300, 2022.

[47] Eric Dennis, Alexei Y Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505, 2002.

[48] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439(1907):553–558, 1992.

[49] Dennis G B J Dieks. Communication by EPR devices. *Physics Letters A*, 92(6):271–272, 1982.

[50] David P DiVincenzo. Topics in quantum computers. In *Mesoscopic electron transport*, pages 657–677. Springer, 1997.

[51] Daoyi Dong and Ian R Petersen. Quantum control theory and applications: a survey. *IET Control Theory & Applications*, 4(12):2651–2671, 2010.

[52] Jiangfeng Du, Xing Rong, Nan Zhao, Ya Wang, Jiahui Yang, and RB Liu. Preserving electron spin coherence in solids by optimal dynamical decoupling. *Nature*, 461(7268):1265–1268, 2009.

[53] Biswadeb Dutta, Narasimhaiengar Mukunda, Rajiah Simon, and Arvind. The real symplectic groups in quantum mechanics and optics. *Pramana Journal of Physics*, 45(6):471–497, 1995.

[54] Freeman J Dyson. The radiation theories of tomonaga, schwinger, and feynman. *Physical Review*, 75(3):486, 1949.

[55] Jack Edmonds. Paths, trees, and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965.

[56] Jens Eisert, Stefan Scheel, and Martin B Plenio. Distilling gaussian states with gaussian operations is impossible. *Physical Review Letters*, 89(13):137903, 2002.

[57] Jens Eisert and Michael M Wolf. Gaussian quantum channels. In *Quantum Information with Continuous Variables of Atoms and Light*, London, 2007. Imperial College Press. arXiv:quant-ph/0505151.

[58] Joseph Emerson, Robert Alicki, and Karol Życzkowski. Scalable noise estimation with random unitary operators. *Journal of Optics B: Quantum and Semiclassical Optics*, 7(10):S347, 2005.

[59] Zachary W E Evans, Ashley M Stephens, Jared H Cole, and Lloyd C L Hollenberg. Error correction optimisation in the presece of X/Y asymmetry. *arXiv:0709.3875*, 2007.

[60] Philippe Faist, Sepehr Nezami, Victor V Albert, Grant Salton, Fernando Pastawski, Patrick Hayden, and John Preskill. Continuous symmetries and approximate quantum error correction. *Physical Review X*, 10(4):041018, 2020.

[61] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. *arXiv preprint quant-ph/0001106*, 2000.

[62] Alessandro Ferraro, Stefano Olivares, and Matteo GA Paris. Gaussian states in continuous variable quantum information. *arXiv preprint quant-ph/0503237*, 2005.

[63] Richard P Feynman. Simulating physics with computers. *International Journal of Theoretical Physics,*, 21:467–488, 1982.

[64] Jaromír Fiurášek. Gaussian transformations and distillation of entangled gaussian states. *Physical Review Letters*, 89(13):137904, 2002.

[65] Christa Flühmann, Thanh Long Nguyen, Matteo Marinelli, Vlad Negnevitsky, Karan Mehta, and JP Home. Encoding a qubit in a trapped-ion mechanical oscillator. *Nature*, 566(7745):513–517, 2019.

[66] Austin G Fowler, Ashley M Stephens, and Peter Groszkowski. High-threshold universal quantum computation on the surface code. *Physical Review A*, 80(5):052312, 2009.

[67] Michael H Freedman and David A Meyer. Projective plane and planar quantum codes. *Foundations of Computational Mathematics*, 1(3):325–332, 2001.

[68] Kosuke Fukui, Akihisa Tomita, Atsushi Okamoto, and Keisuke Fujii. High-threshold fault-tolerant quantum computation with analog quantum error correction. *Physical Review X*, 8:021054, 2018.

[69] Géza Giedke and J Ignacio Cirac. Characterization of Gaussian operations and distillation of Gaussian states. *Physical Review A*, 66:032316, 2002.

[70] Daniel Gottesman. *Stabilizer codes and quantum error correction*. California Institute of Technology, 1997.

[71] Daniel Gottesman, Alexei Y Kitaev, and John Preskill. Encoding a qubit in an oscillator. *Physical Review A*, 64:012310, 2001.

[72] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, pages 212–219, 1996.

[73] Ulrich Haeberlen and John S Waugh. Coherent averaging effects in magnetic resonance. *Physical Review*, 175:453–467, 1968.

[74] Erwin L Hahn. Spin echoes. *Physical Review*, 80:580–594, 1950.

[75] Brian Hall. *Lie groups, Lie algebras, and Representations: an Elementary Introduction*, volume 222. Springer, 2015.

[76] Lisa Hänggli, Margret Heinze, and Robert König. Enhanced noise resilience of the surface–Gottesman-Kitaev-Preskill code via designed bias. *Physical Review A*, 102(5):052408, 2020.

[77] Lisa Hänggli and Robert König. Oscillator-to-oscillator codes do not have a threshold. *IEEE Transactions on Information Theory*, 68(2):1068–1084, 2021.

[78] John H Hannay and Michael V Berry. Quantization of linear maps on a torus-fresnel diffraction by a periodic grating. *Physica D: Nonlinear Phenomena*, 1(3):267–290, 1980.

[79] Patrick Hayden, Sepehr Nezami, Sandu Popescu, and Grant Salton. Error correction of quantum reference frame information. *arXiv:1709.04471*, 2017.

[80] Patrick Hayden, Sepehr Nezami, Grant Salton, and Barry C Sanders. Spacetime replication of continuous variable quantum information. *New Journal of Physics*, 18(8):083043, 2016.

[81] Teiko Heinosaari and Mario Ziman. Guide to mathematical concepts of quantum theory. *arXiv:0810.3536*, 2008.

[82] Margret Heinze and Robert König. Universal Uhrig dynamical decoupling for bosonic systems. *Physical Review Letters*, 123(1):010501, 2019.

[83] Alexander S Holevo and Reinhard F Werner. Evaluating capacities of bosonic gaussian channels. *Physical Review A*, 63(3):032312, 2001.

[84] Andreas Honecker, Marco Picco, and Pierre Pujol. Universality class of the nishimori point in the 2D±J random-bond Ising model. *Physical Review Letters*, 87(4):047201, 2001.

[85] Garng M Huang, Tzyh J Tarn, and John W Clark. On the controllability of quantum-mechanical systems. *Journal of Mathematical Physics*, 24(11):2608–2618, 1983.

[86] Liang Jiang and Adilet Imambekov. Universal dynamical decoupling of multiqubit states from environment. *Physical Review A*, 84:060302, 2011.

[87] Julian Kelly, Rami Barends, Austin G Fowler, Anthony Megrant, Evan Jeffrey, Theodore C White, Daniel Sank, Josh Y Mutus, Brooks Campbell, Yu Chen, et al. State preservation by repetitive error detection in a superconducting quantum circuit. *Nature*, 519(7541):66–69, 2015.

[88] Kaveh Khodjasteh and Daniel A Lidar. Fault-tolerant quantum dynamical decoupling. *Physical Review Letters*, 95(18):180501, 2005.

[89] Kaveh Khodjasteh and Daniel A Lidar. Performance of deterministic dynamical decoupling schemes: Concatenated and periodic pulse sequences. *Physical Review A*, 75(6):062310, 2007.

[90] A Yu Kitaev. Quantum measurements and the abelian stabilizer problem. *arXiv:quant-ph/9511026*, 1995.

[91] Alexei Y Kitaev. Quantum error correction with imperfect gates. In O. Hirota, A. S. Holevo, and C. M. Caves, editors, *Quantum communication, computing, and measurement*, pages 181–188. Springer, 1997.

[92] Alexei Y Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, 2003.

[93] Andreas Klappenecker and Martin Rötteler. Unitary error bases: Constructions, equivalence, and applications. In *International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, pages 139–149. Springer, 2003.

[94] Emanuel Knill. Quantum computing with realistically noisy devices. *Nature*, 434(7029):39–44, 2005.

[95] Emanuel Knill and Raymond Laflamme. Concatenated quantum codes. *arXiv:quant-ph/9608012*, 1996.

[96] Emanuel Knill and Raymond Laflamme. Theory of quantum error-correcting codes. *Physical Review A*, 55(2):900, 1997.

[97] Emanuel Knill, Raymond Laflamme, and Wojciech Zurek. Threshold accuracy for quantum computation. *arXiv:quant-ph/9610011*, 1996.

[98] Emanuel Knill, Raymond Laflamme, and Wojciech H Zurek. Resilient quantum computation. *Science*, 279(5349):342–345, 1998.

[99] Emanuel Knill, Raymond Laflamme, and Wojciech H Zurek. Resilient quantum computation: error models and thresholds. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454(1969):365–384, 1998.

[100] Karl Kraus. General state changes in quantum theory. *Annals of Physics*, 64(2):311–335, 1971.

[101] Karl Kraus, Arno Böhm, John D Dollard, and WH Wootters. *States, Effects, and Operations*. Springer Berlin, Heidelberg, 1983.

[102] David W Kribs, Raymond Laflamme, David Poulin, and Maia Lesosky. Operator quantum error correction. *arXiv:quant-ph/0504189*, 2005.

[103] Wan-Jung Kuo and Daniel A Lidar. Quadratic dynamical decoupling: Universality proof and error analysis. *Physical Review A*, 84:042329, 2011.

[104] Giuliano G La Guardia. On the construction of asymmetric quantum codes. *International Journal of Theoretical Physics*, 53(7):2312–2322, 2014.

[105] Zaki Leghtas, Gerhard Kirchmair, Brian Vlastakis, Robert J Schoelkopf, Michel H Devoret, and Mazyar Mirrahimi. Hardware-efficient autonomous quantum memory protection. *Physical Review Letters*, 111(12):120501, 2013.

[106] Dietrich Leibfried, Rainer Blatt, Christopher Monroe, and David Wineland. Quantum dynamics of single trapped ions. *Reviews of Modern Physics*, 75(1):281, 2003.

[107] Daniel A Lidar and Todd A Brun. *Quantum Error Correction*. Cambridge University Press, 2013.

[108] Daniel A Lidar, Isaac L Chuang, and K Birgitta Whaley. Decoherence-free subspaces for quantum computation. *Physical Review Letters*, 81(12):2594, 1998.

[109] Seth Lloyd and Samuel L Braunstein. Quantum computation over continuous variables. *Physical Review Letters*, 82:1784–1787, 1999.

[110] Seth Lloyd and Jean-Jacques E Slotine. Analog quantum error correction. *Physical Review Letters*, 80:4088–4091, 1998.

[111] Wilhelm Magnus. On the exponential solution of differential equations for a linear operator. *Communications on Pure and Applied Mathematics*, 7(4):649–673, 1954.

[112] Saul Meiboom and David Gill. Modified spin echo method for measuring nuclear relaxation times. *Review of Scientific Instruments*, 29(8):688–691, 1958.

[113] Nicolas C Menicucci, Peter Van Loock, Mile Gu, Christian Weedbrook, Timothy C Ralph, and Michael A Nielsen. Universal quantum computation with continuous-variable cluster states. *Physical Review Letters*, 97(11):110501, 2006.

[114] Marios H Michael, Matti Silveri, RT Brierley, Victor V Albert, Juha Salmilehto, Liang Jiang, and Steven M Girvin. New class of quantum error-correcting codes for a bosonic mode. *Physical Review X*, 6(3):031006, 2016.

[115] Mazyar Mirrahimi, Zaki Leghtas, Victor V Albert, Steven Touzard, Robert J Schoelkopf, Liang Jiang, and Michel H Devoret. Dynamically protected cat-qubits: a new paradigm for universal quantum computation. *New Journal of Physics*, 16(4):045014, 2014.

[116] Musawwadah Mukhtar, Thuan Beng Saw, Wee Tee Soh, and Jiangbin Gong. Universal dynamical decoupling: Two-qubit states and beyond. *Physical Review A*, 81(1):012331, 2010.

[117] Musawwadah Mukhtar, Wee Tee Soh, Thuan Beng Saw, and Jiangbin Gong. Protecting unknown two-qubit entangled states by nesting Uhrig's dynamical decoupling sequences. *Physical Review A*, 82(5):052338, 2010.

[118] Valentin Murg, Frank Verstraete, and J Ignacio Cirac. Variational study of hard-core bosons in a two-dimensional optical lattice using projected entangled pair states. *Physical Review A*, 75:033605, 2007.

[119] Christine A Muschik, Hanna Krauter, Klemens Hammerer, and Eugene S Polzik. Quantum information at the interface of light with atomic ensembles and micromechanical oscillators. *Quantum Information Processing*, 10(6):839–863, 2011.

[120] Hui Khoon Ng, Daniel A Lidar, and John Preskill. Combining dynamical decoupling with fault-tolerant quantum computation. *Physical Review A*, 84(1):012305, 2011.

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

[122] Daniel Nigg, Markus Mueller, Esteban A Martinez, Philipp Schindler, Markus Hennrich, Thomas Monz, Miguel A Martin-Delgado, and Rainer Blatt. Quantum computations on a topologically encoded qubit. *Science*, 345(6194):302–305, 2014.

[123] Julien Niset, Jaromír Fiurášek, and Nicolas J Cerf. No-go theorem for Gaussian quantum error correction. *Physical Review Letters*, 102(12):120501, 2009.

[124] Hidetoshi Nishimori. Internal energy, specific heat and correlation function of the bond-random Ising model. *Progress of Theoretical Physics*, 66(4):1169–1181, 1981.

[125] Kyungjoo Noh, SM Girvin, and Liang Jiang. Encoding an oscillator into many oscillators. *Physical Review Letters*, 125(8):080503, 2020.

[126] Nissim Ofek, Andrei Petrenko, Reinier Heeres, Philip Reinhold, Zaki Leghtas, Brian Vlastakis, Yehan Liu, Luigi Frunzio, Steven M Girvin, Liang Jiang, et al. Extending the lifetime of a quantum bit with error correction in superconducting circuits. *Nature*, 536(7617):441–445, 2016.

[127] Stefano Pasini and Götz S Uhrig. Optimized dynamical decoupling for power-law noise spectra. *Physical Review A*, 81:012309, 2010.

[128] Vern Paulsen. *Completely Bounded Maps and Operator Algebras*. Number 78 in Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2003.

[129] Gerardo A Paz-Silva and DA Lidar. Optimally combining dynamical decoupling and quantum error correction. *Scientific reports*, 3(1):1–6, 2013.

[130] Asher Peres. Reversible logic and quantum computers. *Physical Review A*, 32(6):3266, 1985.

[131] Stefano Pirandola, Stefano Mancini, David Vitali, and Paolo Tombesi. Continuous-variable entanglement and quantum-state teleportation between optical and macroscopic vibrational modes through radiation pressure. *Physical Review A*, 68(6):062317, 2003.

[132] Bibek Pokharel, Namit Anand, Benjamin Fortman, and Daniel A Lidar. Demonstration of fidelity improvement using dynamical decoupling with superconducting qubits. *Physical Review Letters*, 121(22):220502, 2018.

[133] David Poulin. Stabilizer Formalism for Operator Quantum Error Correction. *Physical Review Letters*, 95:230504, 2005.

[134] John Preskill. Lecture notes on quantum computation, 2011. URL: `http://theory.caltech.edu/~preskill/ph229/`. Last visited on 2022/10/13.

[135] Shruti Puri, Lucas St-Jean, Jonathan A Gross, Alexander Grimm, Nicholas E Frattini, Pavithran S Iyer, Anirudh Krishna, Steven Touzard, Liang Jiang, Alexandre Blais, et al. Bias-preserving gates with stabilized cat qubits. *Science advances*, 6(34):eaay5901, 2020.

[136] Robert Raussendorf and Hans J Briegel. A one-way quantum computer. *Physical Review Letters*, 86(22):5188, 2001.

[137] Robert Raussendorf and Jim Harrington. Fault-tolerant quantum computation with high threshold in two dimensions. *Physical Review Letters*, 98(19):190504, 2007.

[138] Michael Reck, Anton Zeilinger, Herbert J Bernstein, and Philip Bertani. Experimental realization of any discrete unitary operator. *Physical Review Letters*, 73(1):58, 1994.

[139] Michael Reed and Barry Simon. *Methods of Modern Mathematical Physics. Volume I: Functional Analysis*. Academic Press, 1980.

[140] Michael Reed and Barry Simon. *Methods of Modern Mathematical Physics. Volume II: Fourier Analysis, Self-Adjointness*. Academic Press, 1980.

[141] Pradeep Kiran Sarvepalli, Andreas Klappenecker, and Martin Rötteler. Asymmetric quantum codes: constructions, bounds and performance. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 465(2105):1645–1672, 2009.

[142] Yunong Shi, Christopher Chamberland, and Andrew Cross. Fault-tolerant preparation of approximate gkp states. *New Journal of Physics*, 21(9):093007, 2019.

[143] Peter W Shor. Scheme for reducing decoherence in quantum computer memory. *Physical Review A*, 52(4):R2493, 1995.

[144] Peter W Shor. Fault-tolerant quantum computation. In *Proceedings of 37th Conference on Foundations of Computer Science*, pages 56–65. IEEE, 1996.

[145] Rajiah Simon, Narasimhaiengar Mukunda, and Biswadeb Dutta. Quantum-noise matrix for multimode systems: U(n) invariance, squeeting, and normal forms. *Physical Review A*, 49:1567–1583, 1994.

[146] Harpreet Singh, Arvind, and Kavita Dorai. Experimental protection of arbitrary states in a two-qubit subspace by nested Uhrig dynamical decoupling. *Physical Review A*, 95:052337, 2017.

[147] Andrew Steane. Multiple-particle interference and quantum error correction. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 452(1954):2551–2577, 1996.

[148] Andrew M Steane. Error correcting codes in quantum theory. *Physical Review Letters*, 77(5):793, 1996.

[149] Andrew M Steane. Active stabilization, quantum computation, and quantum state synthesis. *Physical Review Letters*, 78(11):2252, 1997.

[150] Ashley M Stephens, Zachary W E Evans, Simon J Devitt, and Lloyd C L Hollenberg. Asymmetric quantum error correction via code conversion. *Physical Review A*, 77:062335, 2008.

[151] Ashley M Stephens and Zachary WE Evans. Accuracy threshold for concatenated error detection in one dimension. *Physical Review A*, 80(2):022313, 2009.

[152] Ashley M Stephens, William J Munro, and Kae Nemoto. High-threshold topological quantum error correction against biased noise. *Physical Review A*, 88(6):060301, 2013.

[153] W Forrest Stinespring. Positive functions on C*-algebras. *Proceedings of the American Mathematical Society*, 6(2):211–216, 1955.

[154] Masuo Suzuki. Decomposition formulas of exponential operators and lie exponentials with some applications to quantum mechanics and statistical physics. *Journal of Mathematical Physics*, 26(4):601–612, 1985.

[155] Piotr Szańkowski, Guy Ramon, Jan Krzywda, Damian Kwiatkowski, et al. Environmental noise spectroscopy with qubits subjected to dynamical decoupling. *Journal of Physics: Condensed Matter*, 29(33):333001, 2017.

[156] Maika Takita, Antonio D Córcoles, Easwar Magesan, Baleegh Abdo, Markus Brink, Andrew Cross, Jerry M Chow, and Jay M Gambetta. Demonstration of weight-four parity measurements in the surface code architecture. *Physical Review Letters*, 117(21):210505, 2016.

[157] Barbara M Terhal. Quantum error correction for quantum memories. *Reviews of Modern Physics*, 87(2):307, 2015.

[158] David K Tuckett, Stephen D Bartlett, and Steven T Flammia. Ultrahigh error threshold for surface codes with biased noise. *Physical Review Letters*, 120:050505, 2018.

[159] David K Tuckett, Stephen D Bartlett, Steven T Flammia, and Benjamin J Brown. Fault-tolerant thresholds for the surface code in excess of 5% under biased noise. *Physical Review Letters*, 124(13):130501, 2020.

[160] David K Tuckett, Andrew S Darmawan, Christopher T Chubb, Sergey Bravyi, Stephen D Bartlett, and Steven T Flammia. Tailoring surface codes for highly biased noise. *Physical Review X*, 9:041031, 2019.

[161] Götz S Uhrig. Keeping a quantum bit alive by optimized π-pulse sequences. *Physical Review Letters*, 98:100504, Mar 2007.

[162] Götz S Uhrig. Exact results on dynamical decoupling by π pulses in quantum information processes. *New Journal of Physics*, 10(8):083024, 2008.

[163] Götz S Uhrig and Daniel A Lidar. Rigorous bounds for optimal dynamical decoupling. *Physical Review A*, 82:012301, Jul 2010.

[164] Hilma M Vasconcelos, Liliana Sanz, and Scott Glancy. All-optical generation of states for "encoding a qubit in an oscillator". *Optics Letters*, 35(19):3261–3263, 2010.

[165] Lorenza Viola and Emanuel Knill. Random decoupling schemes for quantum dynamical control and error suppression. *Physical Review Letters*, 94(6):060502, 2005.

[166] Lorenza Viola, Emanuel Knill, and Seth Lloyd. Dynamical decoupling of open quantum systems. *Physical Review Letters*, 82:2417–2421, Mar 1999.

[167] Lorenza Viola and Seth Lloyd. Dynamical suppression of decoherence in two-state quantum systems. *Physical Review A*, 58:2733–2744, Oct 1998.

[168] David Vitali and Paolo Tombesi. Using parity kicks for decoherence control. *Physical Review A*, 59(6):4178, 1999.

[169] Christophe Vuillot, Hamed Asasi, Yang Wang, Leonid P Pryadko, and Barbara M Terhal. Quantum error correction with the toric Gottesman-Kitaev-Preskill code. *Physical Review A*, 99:032344, 2019.

[170] Chenyang Wang, Jim Harrington, and John Preskill. Confinement-higgs transition in a disordered gauge theory and the accuracy threshold for quantum memory. *Annals of Physics*, 303(1):31–58, 2003.

[171] David S Wang, Austin G Fowler, Ashley M Stephens, and Lloyd CL Hollenberg. Threshold error rates for the toric and surface codes. *arXiv:0905.0531*, 2009.

[172] Zhen-Yu Wang and Ren-Bao Liu. Protection of quantum systems by nested dynamical decoupling. *Physical Review A*, 83:022306, 2011.

[173] Zhi-Hui Wang, Wenxian Zhang, Alexei M Tyryshkin, SA Lyon, JW Ager, EE Haller, and Viatcheslav V Dobrovitski. Effect of pulse error accumulation on dynamical decoupling of the electron spins of phosphorus donors in silicon. *Physical Review B*, 85(8):085206, 2012.

[174] Fern HE Watson and Sean D Barrett. Logical error rate scaling of the toric code. *New Journal of Physics*, 16(9):093045, 2014.

[175] John S Waugh, Lee M Huber, and Ulrich Haeberlen. Approach to high-resolution nmr in solids. *Physical Review Letters*, 20:180–182, 1968.

[176] Christian Weedbrook, Stefano Pirandola, Raúl García-Patrón, Nicolas J Cerf, Timothy C Ralph, Jeffrey H Shapiro, and Seth Lloyd. Gaussian quantum information. *Reviews of Modern Physics*, 84(2):621, 2012.

[177] Jacob R West, Bryan H Fong, and Daniel A Lidar. Near-optimal dynamical decoupling of a qubit. *Physical Review Letters*, 104:130501, 2010.

[178] Eugene Wigner. On the Quantum Correction For Thermodynamic Equilibrium. *Physical Review*, 40(5):749–759, 1932.

[179] Mischa P Woods and Álvaro M Alhambra. Continuous groups of transversal gates for quantum error correcting codes from finite clock reference frames. *Quantum*, 4:245, 2020.

[180] Matt J Woolley, Gerard J Milburn, and Carlton M Caves. Nonlinear quantum metrology using coupled nanomechanical resonators. *New Journal of Physics*, 10(12):125018, 2008.

[181] William K Wootters and Wojciech H Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982.

[182] Xiangkun Xu, Zixiang Wang, Changkui Duan, Pu Huang, Pengfei Wang, Ya Wang, Nanyang Xu, Xi Kong, Fazhan Shi, Xing Rong, et al. Coherence-protected quantum gate by continuous dynamical decoupling in diamond. *Physical Review Letters*, 109(7):070502, 2012.

[183] Xiaosi Xu, Qi Zhao, Xiao Yuan, and Simon C Benjamin. High-threshold code for modular hardware with asymmetric noise. *Physical Review Applied*, 12(6):064006, 2019.

[184] Wen Yang and Ren-Bao Liu. Universality of Uhrig dynamical decoupling for suppressing qubit pure dephasing and relaxation. *Physical Review Letters*, 101(18):180403, 2008.

[185] Shota Yokoyama, Ryuji Ukai, Seiji C Armstrong, Chanond Sornphiphatphong, Toshiyuki Kaji, Shigenari Suzuki, Jun-ichi Yoshikawa, Hidehiro Yonezawa, Nicolas C Menicucci, and Akira Furusawa. Ultra-large-scale continuous-variable cluster states multiplexed in the time domain. *Nature Photonics*, 7(12):982–986, 2013.

[186] Paolo Zanardi and Fausto Rossi. Quantum information in semiconductors: noiseless encoding in a quantum-dot array. *Physical Review Letters*, 81(21):4752, 1998.