Universität Stuttgart

Institute for Theoretical Physics III

Master's Thesis

Cross entropy measures for detecting entanglement transitions

Etienne Maurice Springer

27. November 2024

Primary Examiner
Prof. Dr. Hans Peter Büchler

Secondary Examiner
Prof. Dr. Maria Daghofer

Statutory Declaration

I herewith formally declare that I have written the submitted thesis independently. I did not use any outside support except for the quoted literature and other sources mentioned in the paper.

I clearly marked and separately listed all of the literature and all of the other sources which I employed when producing this academic work, either literally or in content.

Stuttgart, 27. November 2024 Etienne Maurice Springer

ABSTRACT

One of the most intriguing ideas in quantum theory is the concept of entanglement. It is well-famed in its role in quantum computation and quantum information, as well as for its puzzling peculiarities, challenging the way we think about nature. In general, entanglement emerges from the interaction of quantum systems with their environment. In a controlled setting, such as an experiment, entangling two systems is usually achieved by the application of unitary interactions.

A key feature of these non-local correlations is the fact that measuring a constituent of an entangled state erases part of the underlying entanglement structure. As such, the growth of entanglement in the lab by the application of unitaries can be stunted by locally measuring parts of the system. Circuits characterized by the interplay between unitaries and local measurements are called *hybrid circuits*.

In the past decade, the study of these hybrid circuits have unveiled intriguing patterns in the dynamics of entanglement. If measurements are sufficiently frequent, one finds that the degree of entanglement between subsystems follows an area law, while in the limit of rare measurements one has a volume law scaling. Remarkably, one can find that there is a phase transition between these two regimes, with a critical measurement rate. These transitions are called *measurement-induced entanglement phase transition*, or entanglement transition for short.

Entanglement transitions have not only been theorized in hybrid circuits, but also in setups, where the (discrete) time evolution is driven by projective measurements alone, where pairwise and local measurements compete. A paradigmatic measurement-only model with such a transition is the *projective transverse-field Ising model*, which features an area law to area law phase transition [1].

However, detecting entanglement transitions in an experimental setting proves to be a difficult, seemingly sysiphean task, due to the *postselection problem*—also referred to as *sampling problem*. It states that due to the probabilistic nature of quantum measurements, one is exponentially unlikely to obtain the same state twice when repeatedly applying the identical protocol. This then obfuscates the precise nature of the state, when probing for it, which is required to quantify the degree of entanglement.

Recently, some protocols have been proposed to circumvent the sampling problem. In particular, Li et al. proposed a quantity which acts as order parameter for the measurement-induced phase transition in hybrid circuits, called the *linear cross entropy*. Furthermore, Garratt and Altman proposed a different method, employing an upper bound from quantum information theory known as Klein's inequality.

In this thesis we put these protocols to test in the projective transverse-field Ising model. It consists of four main chapters and is structured as follows

Chapter 1: In this chapter we give an introduction to the concepts relevant throughout the thesis. We begin by outlining the stabilizer formalism, as it is a useful representation of quantum states and the foundation of the simulation algorithm employed for the numerical approaches in the thesis. We then provide an overview of the concept of entanglement transitions in general, before specifically introducing the projective transverse-field Ising model. This model serves as the primary subject of our studies. We then discuss the post-selection or sampling problem, motivating the rest of the thesis.

Chapter 2: In this chapter we examine the quantity defined by Li et al. in the context of the projective transverse-field Ising model. We particularily motivate post-selection algorithms and compare our results to previous investigations of similar nature (see Ref. [4]), while also raising critiques.

Chapter 3: In this chapter we investigate the approach of Garratt and Altman to detect the entanglement transition in the projective transverse-field Ising model. We first derive an expression for the cross- and relative entropy in the stabilizer formalism, before trying different numerical approaches to circumvent the post-selection problem. We finalize the chapter by outlining two regularization approaches to deal with appearing infinities.

Chapter 4: In this chapter we introduce the simulation algorithm, which is based on the stabilizer formalism outlined in Chapter 1. We first give a brief summary of the existing algorithms. We then introduce the extensions necessary for the numerical simulations performed throughout the thesis, especially concerning mixed states and the different entropies derived in Chapter 3.

ZUSAMMENFASSUNG

Eine der faszinierendsten Ideen in der Quantenmechanik ist das Konzept der Verschränkung. Dieses ist bekannt für seine zentrale Rolle in der Quanteninformation sowie für seine verblüffenden Eigenheiten, die unsere Vorstellungen von der Natur herausfordern. Allgemein entsteht Verschränkung durch die Wechselwirkung von Quantensystemen mit ihrer Umgebung. In kontrollierten Umgebungen, wie etwa Experimenten, wird die Verschränkung zweier Systeme üblicherweise durch die Anwendung unitärer Wechselwirkungen erreicht.

Ein zentrales Merkmal dieser nichtlokalen Korrelationen ist die Tatsache, dass das Messen eines Bestandteils eines verschränkten Zustands einen Teil der zugrunde liegenden Verschränkungsstruktur zerstört. Daher kann das Wachstum der Verschränkung im Labor durch die Anwendung von unitären Operationen gehemmt werden, wenn Teile des Systems lokal gemessen werden. Schaltkreise, die durch das Zusammenspiel von unitären Operationen und lokalen Messungen gekennzeichnet sind, werden als hybride Schaltkreise bezeichnet.

In den letzten zehn Jahren hat die Untersuchung solcher hybrider Schaltkreise faszinierende Muster in der Dynamik der Verschränkung offengelegt. Bei hinreichend häufigen Messungen zeigt sich, dass der Grad der Verschränkung zwischen Teilsystemen einem Flächengesetz folgt. Im Gegensatz dazu ergibt sich bei seltenen Messungen eine Skalierung nach einem Volumengesetz. Bemerkenswerterweise gibt es einen Phasenübergang zwischen diesen beiden Regimen, der durch eine kritische Messrate gekennzeichnet ist. Diese Übergänge werden als messungsinduzierte Verschränkungsphasenübergänge oder kurz Verschränkungsübergänge bezeichnet.

Verschränkungsübergänge wurden nicht nur in hybriden Schaltkreisen theoretisiert, sondern auch in Konfigurationen, bei denen die (diskrete) Zeitentwicklung allein durch projektive Messungen bestimmt wird, wobei paarweise und lokale Messungen konkurrieren. Ein paradigmatisches Modell mit einem solchen Übergang ist das projektive transversale Ising-Modell, das einen Übergang zwischen zwei Flächengesetz-Phasen zeigt [1].

Die experimentelle Detektion von Verschränkungsübergängen erweist sich jedoch als schwierige, scheinbare Sisyphusaufgabe, was auf das sogenannte Postselektionsproblem—auch als Sampling-Problem bezeichnet—zurückzuführen ist. Dieses Problem besagt, dass es aufgrund der probabilistischen Natur von Quantenmessungen exponentiell unwahrscheinlich ist, denselben Zustand zweimal zu erhalten, selbst wenn das identische Protokoll wiederholt angewendet wird. Dies verschleiert die Natur des Zustands bei der Untersuchung, deren Kenntnis erforderlich ist, um den Grad der Verschränkung zu quantifizieren.

Kürzlich wurden einige Protokolle vorgeschlagen, um das Sampling-Problem zu umgehen. Insbesondere schlugen Li u. a. eine Größe vor, die als Ordnungsparameter für den

messungsinduzierten Phasenübergang in hybriden Schaltkreisen dient, die sogenannte lineare Kreuzentropie. Außerdem schlugen Garratt und Altman eine alternative Methode vor, die eine obere Schranke aus der Quanteninformationstheorie nutzt, bekannt als Kleinsche Ungleichung.

In dieser Arbeit werden diese Protokolle im Kontext des projektiven transversalen Ising-Modells getestet. Die Arbeit besteht aus vier Hauptkapiteln und ist wie folgt aufgebaut:

Kapitel 1: In diesem Kapitel werden die für die Arbeit relevanten Konzepte eingeführt. Wir beginnen mit einer Einführung in den Stabilizer-Formalismus, da dieser eine nützliche Darstellung von Quantenzuständen bietet und die Grundlage für den Simulationsalgorithmus bildet, der für die numerischen Ansätze in der Arbeit verwendet wird. Anschließend geben wir einen Überblick über das Konzept der Verschränkungsübergänge im Allgemeinen, bevor wir das projektive transversale Ising-Modell speziell einführen. Dieses Modell ist das Hauptobjekt unserer Untersuchungen. Abschließend wird das Postselektions- bzw. Sampling-Problem erörtert, welches den weiteren Verlauf der Arbeit motiviert.

Kapitel 2: In diesem Kapitel untersuchen wir die von Li u. a. definierte Größe im Kontext des projektiven transversalen Ising-Modells. Besonders wird die Motivation hinter Postselektionsalgorithmen beleuchtet, und unsere Ergebnisse werden mit früheren Untersuchungen ähnlicher Natur (vgl. Ref. [4]) verglichen, wobei auch Kritikpunkte aufgezeigt werden.

Kapitel 3: In diesem Kapitel untersuchen wir den Ansatz von Garratt und Altman zur Detektion von Verschränkungsübergängen im projektiven transversalen Ising-Modell. Zunächst leiten wir einen Ausdruck für die Kreuz- und relative Entropie im Stabilizer-Formalismus her, bevor wir verschiedene numerische Ansätze testen, um das Postselektionsproblem zu umgehen. Abschließend werden zwei Regularisierungsansätze vorgestellt, um auftretende Divergenzen zu bewältigen.

Kapitel 4: In diesem Kapitel wird der Simulationsalgorithmus eingeführt, der auf dem in Kapitel 1 beschriebenen Stabilizer-Formalismus basiert. Zunächst geben wir eine kurze Zusammenfassung bestehender Algorithmen. Anschließend werden die Erweiterungen erläutert, die für die numerischen Simulationen in der Arbeit erforderlich sind, insbesondere in Bezug auf gemischte Zustände und die in Kapitel 3 abgeleiteten Entropien.

CONTENTS

ACKNOWLEDGEMENTS

In the *Phenomenology of Spirit*, Hegel writes: "It is customary to preface a work with an explanation of the author's aim, why he wrote the book, and the relationship in which he believes it to stand to other earlier or contemporary treatises on the same subject." (Hegel in *Phenomenology of Spirit* [5])

While I believe we have addressed this to some degree of satisfaction in the abstract, in the case of an academic work, however, it is further customary to preface a work with a long list of "thank you"s, giving the people behind the presented work credit where credit is due. While I do not want to break with this tradition—I am immensely grateful to, e.g., Professor Büchler for allowing me to work on this thesis and in turn at the ITP3—, I (a) fear the risk of sounding repetitive, and (b) fear that this list cannot, in any way, be comprehensive: what I could write here would never capture how much I learned from Felix and Nicolai in the past year, or from other architects of my intellectual home in the past six.

Thus, killing two birds with one stone, I will come ahead of sounding repetitive by paraphrasing Cheryl Harris in [6], as her eloquence transcends far beyond mine, encapsulating all of the gratitude I could possibly express.

Thank you to all the colleagues at the institute, who have provided support, and the most precious resource of time; to all the staff, who have done hours of invisible work; to all those who have read my work, and to all those whose works inspire me.

Thank you to my students, current and former, for letting me teach and for letting me learn from you. Thank you to all the mentors and comrades in arms, who have been, and still continue to be, supportive, having made the past years manageable.

Thank you to my friends and family, who I love and depend on more than I can say. Thank you for keeping me sane, allowing me to—at times—be hyperactive and deficient in my attention. You put up with crazy schedules, short tempers, mad dashes and all the rest, and I could not thank you enough.

1

INTRODUCTION

"I am fascinated by numbers"

Baron-Cohen et al.

This chapter serves to familiarize the reader with the core concepts relevant to this thesis. We will first introduce the stabilizer formalism, as it will later enable us to perform efficient numerical experiments on a classical computer. Next, we provide an overview of the field of entanglement transitions and cite some important examples and works of the field. We will then introduce the model system that is subject to the main part of this thesis, the projective transverse-field Ising model, motivate it and highlight its most intriguing features. Finalizing this chapter is a more metaphysical section on the sampling problem, which deals with the ontology of entanglement and phase transitions in entanglement.

I.I THE STABILIZER FORMALISM

In quantum computation and quantum computing we are exposed to a multitude of obstacles to overcome. For starters, there is the issue of quantum error correction. The encoding of information in a qubit allows for more than the usual bit-flip error from classical computing. Then there is the problem of simulating a quantum computer on a classical one. This requires, in principle, the storage of exponentially many amplitudes for every additional qubit one wants to simulate. While it seems almost absurd to suggest that these two problems would be dealt with by the same formalism, we nevertheless want to wash out doubts about this fact in the present section and present an introduction to the stabilizer formalism.

Although it was originally introduced as an approach to quantum error correction [8–11], the notion of stabilizers has become synonymous with efficient classical simulability in the world of quantum information theory [8, 12–15].

As the stabilizer formalism relies heavily on group theoretic arguments, we will first begin by giving a brief overview of the foundational algebra.

I.I.I Basic notions of group theory

Group theory is one of the most important concepts from algebra in the toolset of theoretical physicists. From classifying crystalline structures [16], (gauge) symmetries in the standard model [17], to the classification of (topological) phase transitions [18, 19], group theory perpetually permeates theoretical physiscs.

The stabilizer formalism makes no exception. It is a clever application of group theory, allowing for a more compact representation of quantum states compared to the state vector. We therefore introduce necessary prerequesites of group theory needed for the stabilizer formalism and its role in this thesis, starting with the notion of a group.

A group G is a collection of some particulars g that can be composed according to some convention. To be called a group, the set of entities $\{g_i\}$ and the operation[1] with which we compose them, need to obey a certain ruleset [20]. This ruleset, called the group axioms, reads as follows [21].

[1] In the discussion on groups and group theory, the words composition, group operation, and multiplication are used interchangeably

Definition 1.1 (Group)**.** A group G is a non-empty set equipped with a binary operation (here denoted with $\cdot$) that satisfies the following axioms.

Associativity: The group operation is associative, i.e.

$$\forall a, b, c \in G : \ (a \cdot b) \cdot c = a \cdot (b \cdot c).$$

Identity element: The group contains an identity element, which does nothing with respect to composition, i.e.

$$\exists I \in G \, \forall g \in G : I \cdot g = g \cdot I = g.$$

Inverse element: Each group element has a unique element associated to it that when composed with it gives the identity. In other words,

$$\forall g \in G \, \exists g^{-1} \in G : g g^{-1} = g^{-1} g = I.$$

Note that we do not require the group elements to commute with respect to multiplication. Groups that satisfy commutativity for all their elements are called *abelian* groups.

Numerous different mathematical objects and concepts can fall under the umbrella of group theory.[2] Although we will discuss specific groups in greater detail later, we do not want to fail to mention some other important groups appearing all across physics. There are the rotation groups $SO(n)$ and unitary groups $U(n)$ (both in n dimensions), the permutation group of n elements S_n and the group of square roots of 1 under multiplication, $\mathbb{Z}_2 = \{1, -1\}$. The first two are examples of continuous groups, while the others are discrete. Note that discrete and infinite are not mutually exclusive. The group $(\mathbb{Z}, +)$ is an example of a discrete group with infinite number of elements. The number of elements in a finite group G is called the *order* of a group, which we write as $\mathrm{ord}(G)$.

For larger finite groups, it can become cumbersome to keep track of all the elements and their relation to each other. Luckily, there is a way we can condense all the information to construct the whole group in its *presentation*, also known as its *generating set* [20]. The elements of such a generating set are referred to as *generators*.

[2] One can even describe the Rubik's Cube puzzle in the language of group theory

Definition 1.2 (Generating set and generators)**.** Let G be a finite group and S a subset of G. S is called a generating set of G if all elements in G can be obtained from (possibly repeated) multiplication of elements in the generating set. Generating sets are denoted by angled braces, such that we write

$$S = \langle g_1, \dots, g_n \rangle, \quad g_1, \dots, g_n \in G. \tag{1.1}$$

The g_i are called *generators* of G. The trivial group $(\{I\}, \cdot)$ is generated by the empty set.

While there are also generators of continuous or infinite groups, they take on a fundamentally different form compared to the finite counterpart. As an example of a generating set in the finite case, consider the group of fourth roots of 1 with multiplication, $Z_4 = (\{\pm 1, \pm i\}, \cdot)$. It is easy to verify that the subset $S = \{i\}$ of the group uniquely reproduces all of the elements in Z_4, by taking powers of i.

In the process of constructing a generating set we are generally faced with two restrictions. The first is the fact that the entire group needs to emerge from the multiplications of generators. We cannot simply choose arbitrary group elements. $S = \{-1\} \subset Z_4$ would be a perfectly fine set of generators for $\mathbb{Z}_2$, but not Z_4. Next, we ideally want to have the least number of generators possible to build up the rest of the group. This restriction is one we impose on ourselves rather than one imposed on us by necessity. Choosing $S = \{-1, i, -i\}$ as generators also recovers Z_4, however, we have already seen that $g = -1$ and $g = -i$ are redundant in this context. Theorem 1.3 gives an upper bound on the number of generators needed to generate finite groups.

Theorem 1.3 ([8])**.** The minimum size of a generating set for a finite group G of generators is at most $\log_2(\mathrm{ord}(G))$.

With Theorem 1.3 we have that for Z_4, choosing a generating set with 3 elements should be cause for concern, since we would need 2 at most. However, we also saw that Z_4 is special in that way, since we only needed one generator. A way of quantifying this quality is the *rank* of a group. It is defined as the size of the smallest generating set of G and is denoted by $|G|$. Thus, for the example of Z_4 we have $|Z_4| = 1$.

When discussing subsets of groups, one naturally arising concept is the notion of subgroups. Suppose we take some set of elements $\{g\}$ forming a group G under multiplication and take a subset $\{h\}$ thereof. If the subset also forms a group H, we call it a subgroup of G and write $H \leq G$.

Definition 1.4 (Subgroup [20])**.** A subgroup H of G, written as $H \leq G$, is a non-empty subset H of G, which forms a group under the same group operation as G.

Going back to some of the previous examples, we can consider $SO(2)$, rotations along the unit circle, as rotations on the equator of a unit sphere. We can consequently take $SO(2)$ as a subgroup of the rotation group of the unit sphere $SO(3)$. The group of permutations of $m \leq n$ elements is just the group of permutations of n elements, where $n-m$ elements are left invariant, and as such $S_m \leq S_n$. Note that for any group G we have $G \leq G$ and $(\{I\}, \cdot) \leq G$, where $(\{I\}, \cdot)$ is the trivial group containing only the identity. These two special cases are referred to as *trivial subgroups*. Subgroups that are not trivial are called *proper subgroups* denoted by $H < G$.

Before introducing another important family of subgroups, we define a special kind of operation known as *conjugation*. If $h, g \in G$, the conjugate of h with respect to g is $g^{-1}hg$.[3] We can not only perform this operation on individual group elements, but also subgroups of G. Consider the proper subgroup $H < G$ with elements $\{h_i\}$. If we take any element $g \in G \setminus H$, we can arrive at another subgroup of G by conjugating all elements of the subgroup H with g, written as

$$g^{-1}Hg = \{g^{-1}hg \mid h \in H\}. \tag{1.2}$$

In general, the two subgroups H and $g^{-1}Hg$ need not be the same. However, if there are some g which leave H invariant under conjugation, we say that these g *normalize* H. A collection of these normalizing elements can be compiled together to form yet another subgroup of G, called the normalizer.

Definition 1.5 (Normalizer [11])**.** Let G be a group and $H < G$ a proper subgroup of G. The normalizer of H in G is the subgroup of G that leaves H invariant under conjugation, i.e.

$$N_G(H) = \{g \in G \mid g^{-1}Hg = H\}. \tag{1.3}$$

In the special case of every element of G normalizing H, that is $N_G(H) = G$, H is called an *invariant subgroup* of G [20]. In the next section we will introduce an important example of a finite group and its normalizer.

[3] This operation is colloquially referred to as "sandwiching" h with g.

1.1.2 The Pauli group and Clifford gates

Consider the Pauli matrices with the identity,

$$\sigma_0 = I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \text{and} \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{1.4}$$

These well-known matrices are both hermitian and unitary, and consequently square to the identity. For the latter three of them one can show that they satisfy the following commutation and anticommutation relations,

$$\begin{aligned} [\sigma_j, \sigma_k] &= 2\mathrm{i}\epsilon_{jkl}\sigma_l \\ \{\sigma_j, \sigma_k\} &= 2I\delta_{jk} \quad \text{and} \\ \sigma_j\sigma_k &= \frac{1}{2}\left([\sigma_j, \sigma_k] + \{\sigma_j, \sigma_k\}\right) = \delta_{jk}I + \mathrm{i}\epsilon_{jkl}\sigma_l, \end{aligned} \tag{1.5}$$

with the Levi-Civita tensor ϵ_{jkl} (where Einstein summation convention is implied) and the Kronecker delta δ_{jk}. Furthermore, the latter three are traceless and have eigenvalues of ± 1. To ease up on the indices, especially once tensor products of Pauli matrices come into play, one also writes the Pauli matrix with the corresponding capital letter, $\sigma_x = X, \ldots$. These matrices also form a basis for hermitian 2×2 matrices. Recall that physical observables are represented by hermitian matrices. We can therefore consider the Pauli matrices as a basis for physical observables on qubits.

While they also play an important role in representation theory, especially of Lie and Clifford algebras, they themselves also form a group known as the Pauli group $\mathcal{P}$. The single-qubit Pauli group is defined as the Pauli matrices with phases ± 1 and $\pm i$,

$$\mathcal{P} = \{\pm I, \pm iI, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}. \tag{1.6}$$

This definition can also be generalized to n qubits.

Definition 1.6 (Pauli group [11])**.** The Pauli group $\mathcal{P}_n$ is composed of tensor products of I, X, Y, and Z on n qubits with an overall phase of ± 1 and $\pm i$.

We can deduce from Equation (1.5) that $\mathcal{P}_n$ is not Abelian. The commutation relations can be extended for the tensor products of Paulis, but there are some cases that then commute non-trivially. For instance, $X \otimes X \equiv X_1X_2$ commutes with $Z \otimes Z \equiv Z_1Z_2$, even though the individual Pauli matrices in the tensor products do not commute. These commutation relations can be compacted into a general statement on tensor products of Pauli operators. Two n-qubit Pauli operators commute non-trivially if there are an even number of anti-commuting pairs in the tensor product structure. This excludes the identity, of course,

since everything commutes trivially with the identity. The task of finding the commutation relation between two operators then becomes a counting task.

Another group worth considering is the Clifford group, C_n, which is often defined as the subgroup of unitary matrices with dimension 2^n that normalize (cf. Definition 1.5) the n-qubit Pauli group,

$$\mathsf{C}_n = \{u \in U(2^n) \mid u^\dagger \mathcal{P}_n u = \mathcal{P}_n\}.$$

Instead of being a finite group the Clifford group defined in this way is infinite, since it includes all matrices of the form $u = e^{i\varphi} I$ with some phase $\varphi \in \mathbb{R}$. By defining it in terms of a finite subgroup of the above definition, the physical significance of the Clifford group also becomes apparent.

Definition 1.7 (Clifford gates [11])**.** The Clifford group is the group generated by the Hadamard, Phase and CNOT gates. These are called the Clifford gates.

The Clifford gates form an important subset of gates in quantum computing and especially quantum error correction [22–26]. However, it should be noted that they do not form a universal set of quantum gates. While the gates in the Clifford group can create entangled states with the Hadamard and CNOT gates, one needs an additional gate to achieve universal quantum computation.

1.1.3 The stabilizer group

So far we have only examined groups in isolation. Among other things, we have shown that the group generated by the Clifford gates normalizes the Pauli group. However, the major role group theory plays in physics can best be demonstrated if one considers the action of group elements on other mathematical objects outside of the group. These could be, for example, Lagrangians, or more relevant for us, state vectors. Consider the two-qubit state vector $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$. As an eigenstate of the Pauli operator X, we can tell that this state is resistant to bitflips. The group-theoretic way to put this is to say that $|+\rangle$ is $\mathbb{Z}_2$-symmetric. This notion of symmetry is where group theory finds most of its utility in physics. As an additional example, consider the 2-qubit Bell state

$$|\phi\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}. \tag{1.7}$$

Note that the unitary operations X_1X_2 and Z_1Z_2 both have $|\phi\rangle$ as eigenstate with eigenvalue +1. Since

$$X_1X_2Z_1Z_2 |\phi\rangle = |\phi\rangle = Z_1Z_2X_1X_2 |\phi\rangle,$$

we can also see how these two operators commute with each other. The operators X_1X_2 and Z_1Z_2 are then said to *stabilize* the state $|\phi\rangle$. It is easy to convince oneself that these operations stabilizing the state $|\phi\rangle$ should form a group. Doing nothing, i.e. the identity I, clearly stabilizes the state, and since the Pauli matrices square to the identity, each element of this stabilizer group is its own inverse (we omit verifying associativity, as this is inherited from $\mathcal{P}_n$). The matrices X_1X_2 and Z_1Z_2 therefore generate a symmetry group of $|\phi\rangle$, since their product is clearly also a symmetry transformation on $|\phi\rangle$. Therefore, the final concept of group theory we introduce is that of a symmetry group.

Definition 1.8 (Symmetry group)**.** Let G be a group acting on a set M. Let $a \in M$. We then call the subgroup

$$H = \{h \in G \mid ha = a\} \leq G$$

symmetry group or *fixpoint group* of a.

In our example, we have the group of 2-qubit Pauli matrices, $\mathcal{P}_2$, acting on the 2-qubit Hilbert space $H^{\otimes 2}$, with a subgroup of this group, $\mathcal{S} = \{I, X_1X_2, -Y_1Y_2, Z_1Z_2\} \leq \mathcal{P}_2$, being the symmetry group of the state $|\phi\rangle \in H^{\otimes 2}$. In general, we say that a unitary U stabilizes a pure state $|\psi\rangle$ if $U|\psi\rangle = |\psi\rangle$. In other words, the stabilizer group of a pure state $|\psi\rangle$ is the set of all unitaries that have $|\psi\rangle$ as eigenvector with eigenvalue +1. For all further considerations we restrict the unitaries to Pauli operators. Thus, the formal definition of an n-qubit stabilizer group can be stated as follows.

Definition 1.9 (Stabilizer group)**.** Let $H^{\otimes n}$ denote the n-qubit Hilbert space. Given a subset $V \subseteq H^{\otimes n}$, the stabilizer is defined as

$$\mathcal{S}_V = \{g \in \mathcal{P}_n \mid g|\phi\rangle = |\phi\rangle \ \forall\, |\phi\rangle \in V\} \leq \mathcal{P}_n. \tag{1.8}$$

It would, in principle, be possible to define stabilizer groups of all unitaries instead of the Pauli group. We will later see, however, that this restriction leads to an important result, namely Theorem 1.14.

We note that global phase matters here. The operators with prefactor, such as $-I$ are not in the stabilizer. Furthermore, we have that the stabilizer is an Abelian subgroup of $\mathcal{P}_n$, which can be shown by generalizing the example above. We can show the necessity of this condition in the following. Suppose $|\phi\rangle \in V_S$ is non-zero, and M and N are in $\mathcal{S}$. Since M and N are tensor products of Pauli matrices, they either commute or anticommute. If they anticommute we have

$$|\phi\rangle = MN|\phi\rangle = -NM|\phi\rangle = -|\phi\rangle, \tag{1.9}$$

leading to a contradiction, since we had $|\phi\rangle$ being non-zero. (By the same argument we can rule out $-I$ to be in the stabilizer.)

At this point, we need to be careful not to put the cart before the horse. The stabilizer group is not the stabilizer of V as such. If that were the case, then $V = \{|00\rangle, |11\rangle\}$ would be stabilized by X_1X_2. Rather, V_S is the intersection of subspaces spanned by the eigenvalue $+1$ eigenspaces of the operators in $\mathcal{S}$. What this means in practice is that when working with the stabilizer formalism, we would much rather first write out an Abelian subgroup of the Pauli group (without $-I$) and then deduce the subspace stabilized by this subgroup.

Definition 1.10 (Stabilized state space)**.** Let $\mathcal{S} \leq \mathcal{P}_n$ be Abelian with $-I \notin \mathcal{S}$. The space of states stabilized by $\mathcal{S}$ is

$$V_S = \{|\phi\rangle \mid g|\phi\rangle = |\phi\rangle \ \forall g \in \mathcal{S}\}. \tag{1.10}$$

In the literature this space is also sometimes equivalently referred to as the code space of the stabilizer.

So far, it is not entirely obvious how keeping track of operators growing exponentially in size is a worthwhile method of representing quantum states or subspaces of a larger state space. However, we can simplify the problem drastically by realizing two facts.

The first is summarized in Definition 1.2 and Theorem 1.3. Recall that we can equivalently write a finite group as a collection of at most $\log(\mathrm{ord}(G))$ generators. The Bell state example from above has a full stabilizer group of $\mathcal{S} = \{I, X_1X_2, -Y_1Y_2, Z_1Z_2\}$. With Equation (1.5) we can infer that $X_1X_2 \cdot Z_1Z_2 = -Y_1Y_2$, and since all of them square to the identity, an equivalent form of the stabilizer group is given by the generating set $G = \langle X_1X_2, Z_1Z_2\rangle$. At this point we remark that while the generating set is explicitly not the entire group, the distinction between the two is kept rather loosely. Oftentimes we will write that some stabilizer group $\mathcal{S}$ is equal to its generating set. This is mostly a matter of convenience and readability. If context does not explicitly demand it, we write the generating set and group interchangeably.

The second fact is a more subtle one. It uses the commutation relations laid out in Equation (1.5). If we briefly neglect the phases again, or keep track of them separately, we can write $Y = XZ$. As such, we can encode the entire stabilizer group in a bit matrix, where $I \equiv 00$, $Z \equiv 01$, $X \equiv 10$, and $Y \equiv 11$. The bit matrix of the Bell state is

$$\mathcal{S} \equiv \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}. \tag{1.11}$$

This check matrix has a multitude of different use cases, one of them being the focus of the entirety of Chapter 4. Another one is that if the rows of the check matrix are linearly independent (mod 2), we have a minimal generating set. The generators of such a minimal generating set are then called independent.

Stabilizer density matrix

Before we go on to discuss the dynamics in the stabilizer formalism, i.e. stabilizer circuits, we define the density matrix for stabilizers. We know that a stabilizer group $\mathcal{S}$ consists of Pauli operators, which share eigenstates. Thus, the density matrix of a pure state $\rho = |\phi\rangle\langle\phi|$ is a projector onto all +1 eigenstates of the Pauli operators in the stabilizer. This can be constructed by multiplying the projectors onto +1 eigenstates of the generating set. This follows from $\mathbb{P}^2 = \mathbb{P}$ and the fact that the generating set recovers the whole group with multiplication. Since we can write projectors of Pauli operators g as $\mathbb{P} = \frac{1}{2}(I + g)$, we can write

$$\rho = \frac{1}{2^n} \prod_{i=1}^{n} (I + g_i) \tag{1.12}$$

for the density matrix of a pure state.

A mixed state in the stabilizer formalism can be described by removing generators from the generating set.

Definition 1.11 (Stabilizer density matrix)**.** Let $\mathcal{S}$ be an n-qubit stabilizer group with generating set $\langle g_1, \dots, g_l \rangle$ with $0 \leq l \leq n$. The density matrix corresponding to this stabilizer group is given by

$$\rho = \frac{1}{2^n} \prod_{i=1}^{l} (I + g_i) \tag{1.13}$$

or alternatively as

$$\rho = \frac{1}{2^n} \sum_{s \in \mathcal{S}} s. \tag{1.14}$$

We can convince ourselves that this definition is consistent with the general properties we demand density matrices to fulfill.

1. $\mathrm{Tr}[\rho] = 1$, which is given by the fact that every matrix in the product of Equation (1.13) or the sum of Equation (1.14) is either a traceless Pauli matrix, or the identity of dimension 2^n. The trace of the latter thus gets absorbed by the leading factor.

2. $\mathrm{Tr}[\rho^2] \leq 1$ is given by the fact that $\rho = c\mathbb{P}_S$ with $c \leq 1$, and $\mathbb{P}_S$ the projector onto the code space of the stabilizer group. We have

$$\mathrm{Tr}[\rho^2] = \mathrm{Tr}[c^2 \mathbb{P}_S^2] = \mathrm{Tr}[c^2 \mathbb{P}_S] \leq \mathrm{Tr}[c\mathbb{P}_S] = \mathrm{Tr}[\rho] = 1. \tag{1.15}$$

3. $\rho = \rho^\dagger$, which is given by the hermiticity of the Pauli matrices.

Entropy of entanglement

With the notion of a density matrix defined in the stabilizer formalism, we can ask the question about entropic quantities. Since we introduce methods to compute other entropic and information-theoretic quantities in Section 3.1.2, we will here only define the entropy of entanglement. Its definition can be stated as follows.

Definition 1.12 (Entropy of entanglement)**.** Let $|\phi\rangle \in H^{\otimes N}$ be a bipartite pure state with subsystems A and B. The entropy of entanglement of $|\phi\rangle$ then reads

$$S_{\mathrm{E}}\left(|\phi\rangle\right) = -\operatorname{Tr}\left[\rho_B \log \rho_B\right] = S_{\mathrm{vn}}(\rho_B), \tag{1.16}$$

where $\rho_B = \operatorname{Tr}_A[|\phi\rangle\langle\phi|]$ is the reduced density matrix of subsystem B and $S_{\mathrm{vn}}(\rho_B)$ is the von Neumann entropy, where one usually drops the index "vn" when referring to the von Neumann entropy. Conventionally, one uses the logarithm of base 2.

In bipartite states this quantity measures how entangled one subsystem is with the other. That is, it gives a numerical value to the non-local correlations between subsystems. If we split our stabilizer group into local subgroups of A and B, $\mathcal{S}_A$ and $\mathcal{S}_B$, we have the possibility of another subgroup remaining, namely $\mathcal{S}_{AB}$ accounting for correlations. $\mathcal{S}_{A(B)}$ are subgroups containing only operators acting on $A(B)$. If we split our stabilizer group in this way, we find that there is a nice closed-form expression for the entanglement entropy [27].

Theorem 1.13 (Entropy of entanglement – stabilizer)**.** The entropy of entanglement of a bipartite pure state $|\phi\rangle_{AB}$ with partitions A and B is given by

$$S_E(|\phi\rangle) = \frac{1}{2}|\mathcal{S}_{AB}|. \tag{1.17}$$

For the 2-qubit Bell state we once again have $\mathcal{S} = \langle XX, ZZ\rangle$. Notice that $\mathcal{S}_A = \mathcal{S}_B = \{I\}$ and $\mathcal{S}_{AB} = \mathcal{S}$. The entropy of entanglement between A and B should be 1 since there is only one independent entangled pair. We could additionally compute the reduced density matrices and the matrix logarithm instead, arriving at the same result.

With Equation (1.17) we find that this result is also obtained by counting the generators in $\mathcal{S}_{AB}$, which is 2, then dividing by 2, which gives the correct result of 1.

The proof of Theorem 1.13 and the practical computational implementation are rather involved. The interested reader is invited to consult Ref. [27], where the method is introduced. For now, it suffices to say that the generating set can be brought into a canonical form, consisting of two sets generating A and B, respectively, and two sets of generators, which anticommute in pairs, that generate AB. The size of this last set is the one which gives the entanglement entropy.

1.1.4 Stabilizer circuits

Finally, we discuss the dynamics of stabilizer states in quantum circuits. This is where we can see the main advantages of the stabilizer formalism, as it will ultimately lead to the efficient simulation of a wide class of quantum circuits with classical computers. As introductory note we define a stabilizer circuit to be a quantum circuit using only Clifford gates and measurement gates of Pauli operators. Let us therefore begin by discussing unitary gates.

Unitary gates

Suppose we apply a unitary U to a vector space V_S stabilized by $\mathcal{S}$. For any $|\phi\rangle \in V_S$ and any $g \in \mathcal{S}$ we have

$$U|\phi\rangle = Ug|\phi\rangle = Ug\underbrace{U^\dagger U}_{I}|\phi\rangle . \tag{1.18}$$

We thus have that the state $U|\phi\rangle$ is stabilized by the operator $UgU^\dagger$. Since our choices of $|\phi\rangle$ and g were arbitrary in V_S and $\mathcal{S}$, respectively, we have that the transformed vector space is stabilized by the conjugated stabilizer group

$$U\mathcal{S}U^\dagger = \{UgU^\dagger \mid g \in \mathcal{S}\}. \tag{1.19}$$

Since we restricted ourselves to unitary gates in the Clifford group, which normalizes the Pauli group, we still exclusively have Pauli operators in $USU^\dagger$. Our task then becomes to track the effects of Clifford group elements on a subset of Pauli operators. In particular, we conjugate the generators in the generating set with the unitary operation corresponding to the applied gate. Let us therefore consider the conjugation of the Pauli matrices with the unitary Clifford gates. For the Hadamard gate $H = \frac{X+Z}{\sqrt{2}}$ we have

$$HXH^\dagger = Z, \quad HYH^\dagger = -Y, \quad HZH^\dagger = X \tag{1.20}$$

For the CNOT gate U with qubit 1 as control and qubit 2 as target we have

$$UX_1U^\dagger = X_1X_2, \quad UX_2U^\dagger = X_2 \tag{1.21}$$

$$UZ_1U^\dagger = Z_1, \qquad UZ_2U^\dagger = Z_1Z_2, \tag{1.22}$$

where for all other two qubit Pauli operators we can use the above relations and Equation (1.5) to deduce their conjugation relations. Furthermore, we already know an efficient method of tracking generating sets through X and Z alone. Lastly, we want to consider the action of the phase gate on the Pauli operators,

$$SXS^\dagger = Y \quad SZS^\dagger = Z. \tag{1.23}$$

Note that the Clifford gates being the normalizer of the Pauli group means that we will inevitably lose gates, which would realize universal quantum computing. Notable examples of gates not included in the gate set generated by the Clifford group are the T gate, also known as $\pi/8$ gate, and the Toffoli gates.

Measurements

We now know the mechanisms behind unitary gates in the stabilizer formalism. However, we can also include measurement gates in quantum circuits. It turns out that measurements can also be described in a simple way in the stabilizer formalism. Since physical observables are represented by hermitian operators, we can assume that the measurement operator is a product of Pauli matrices $M \in \mathcal{P}_n$. Suppose we measure M in a system in a state $|\phi\rangle$ with stabilizer group $\mathcal{S} = \langle g_1, \dots, g_n \rangle$. Two questions naturally arise: what is the measurement result, and how does the stabilizer group transform under this measurement?

To answer both of these, we first need to realize the two distinct possibilities:

1. M commutes with all the generators of the stabilizer group.
2. M anticommutes with at least one of the generators. By restructuring the generating set, we can ensure that there is at most one generator anticommuting with M. We thus assume without loss of generality that M anticommutes with g_1.

For the first case, we have that either M or $-M$ is itself an element of the stabilizer group, since for arbitrary generators g_j

$$g_j M |\phi\rangle = M g_j |\phi\rangle = M |\phi\rangle . \tag{1.24}$$

As this holds for any g_j, we have that $M |\phi\rangle = \pm |\phi\rangle$, where either $\pm M$ is in the stabilizer group. Assuming w.l.o.g. that $M \in \mathcal{S}$, we have that the measurement of M yields the result $+1$ with probability one, i.e. deterministically. Since the stabilizer group already contained M, it does not change.

In the second case, we have that the measurement operator anticommutes with g_1. Note that the projectors for the measurement outcomes are $\mathbb{P} = \frac{1}{2}(I \pm M)$. Therefore, the probabilities for the respective outcomes are given by

$$P(\pm 1) = \mathrm{Tr}\left[\frac{1}{2}(I \pm M)\,|\phi\rangle\langle\phi|\right]. \tag{1.25}$$

With $g_1 |\phi\rangle = |\phi\rangle$ and $\{g_1, M\} = 0$ we have

$$\begin{aligned} P(+1) &= \mathrm{Tr}\left[\frac{I+M}{2} g_1 |\phi\rangle\langle\phi|\right] \\ &= \mathrm{Tr}\left[g_1 \frac{I-M}{2} |\phi\rangle\langle\phi|\right] \\ &= \mathrm{Tr}\left[\frac{I-M}{2} |\phi\rangle\langle\phi| g_1\right] \\ &= \mathrm{Tr}\left[\frac{I-M}{2} |\phi\rangle\langle\phi|\right] = P(-1). \end{aligned} \tag{1.26}$$

Since we can only measure $+1$ or -1 and the probabilities for either are equal, we can deduce that $P(\pm 1) = 1/2$. After the measurement, the generating set gets affected such that it is now $\langle \pm M, g_2, \dots, g_n \rangle$ depending on the measurement outcome.

Stabilizer circuits and error correction

No discussion of the stabilizer formalism would be complete without as much as a mention of its applications in quantum error correction. To this end, consider the quantum circuit shown in Figure 1.1. A variation of this circuit will play an important role in our discussion of the projective transverse-field Ising model in Section 1.3.

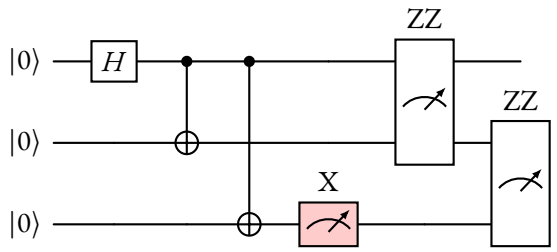


Figure 1.1.: Circuit showing the error detection and correction capabilities within the stabilizer formalism. First, a Bell cluster is created by the application of the Hadamard and two CNOT gates. Afterwards, an error occurs in the form of an X measurement, which we can attempt to detect via stabilizer measurements.

Initially, we have the state $|000\rangle$, which is stabilized by $\mathcal{S} = \langle Z_1, Z_2, Z_3\rangle$. After applying H to qubit 1 in the circuit, the stabilizer generators are conjugated with H_1, such that we now have $\mathcal{S} = \langle X_1, Z_2, Z_3\rangle$. The corrresponding state stabilized by $\mathcal{S}$ is $|+\rangle \otimes |00\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} \otimes |00\rangle$. After the first CNOT, we have $\mathcal{S} = \langle X_1X_2, Z_1Z_2, Z_3\rangle$ according to the transformation rules given in Equation (1.21). Before the measurement on qubit 1, we apply another CNOT gate, thus creating a 3-qubit Bell cluster $(|000\rangle + |111\rangle)/\sqrt{2}$ with stabilizer group $\mathcal{S} = \langle X_1X_2X_3, Z_1Z_2, Z_2Z_3\rangle$.

Then, an error occurs in the form of an X measurement on qubit 3. The stabilizer generators are now $\langle X_1X_2, Z_1Z_2, X_3\rangle$, which we are, in principle, unaware of. However, since we are performing syndrome measurements afterwards in the form of ZZ measurements, we can detect, and subsequently correct the error. In our example, a measurement of Z_1Z_2 would be a measurement of a stabilizer generator, and would thus yield $+1$. Next, we measure Z_2Z_3. According to the previously introduced rules, the outcome is ± 1 with probability $\frac{1}{2}$ and the stabilizer generators afterwards are $\mathcal{S} = \langle X_1X_2X_3, Z_1Z_2, \pm Z_2Z_3\rangle$. In the case where the outcome is $+1$, we effectively corrected the error already, since we recovered the previous stabilizers. In the converse case, we at least detected the syndrome of the error: bit flip on qubit 3 must have occurred! We can then correct the error appropriately and flip it back by means of applying a Pauli-X.

Simulation of stabilizer circuits

Simulating stabilizer circuits can thus be done by keeping track of all the generators and updating them accordingly. As we have seen in above example, we were not required to perform exponentially hard calculations to get to the final state of the circuit. This fact can be used to formulate the following theorem [28].

Theorem 1.14 (Gottesman-Knill theorem)**.** Suppose a quantum computation is performed which involves only the following elements: state preparations in the computational basis, Clifford gates, and measurements of observables in the Pauli group (which includes measurement in the computational basis as a special case), together with the possibility of classical control conditioned on the outcome of such measurements. Such computation may be efficiently simulated on a classical computer.

We will forego a detailed discussion of the simulation of stabilizer circuits on classical computers, as well as the proof to Theorem 1.14 until Chapter 4. For now we remind ourselves that the Clifford gate set does not constitute a universal set of quantum gates. Thus, classical computers *cannot* suffice to efficiently simulate quantum computers.

1.2 ENTANGLEMENT TRANSITIONS

Quantum entanglement is one of the most important features of quantum mechanics, opening the door to quantum computation and quantum information. It distinguishes quantum information from classical information, as the creation, manipulation, and exploitation of entanglement is what quantum information science is ultimately all about. In the past, entanglement has famously perplexed even the greatest physicists [29, 30]. This tradition has been kept ongoing as it still, to this day, excites the curiosity of physicists; from the study of black holes to entanglement growth on a microscopic scale [31, 32], among other things.

One particular excitation of the curious minds specifically relevant to this thesis is the field of measurement-induced entanglement phase transitions, or entanglement transitions for short. This area in the study of entanglement, kickstarted by three groups independently [33–35], has since become an active field in theoretical physics [36–43]. In this section we provide an overview over the key concepts of measurement-induced phase transitions relevant to the thesis.

This section is a rough adaption of and largely follows Ref. [44], if not explicitly stated otherwise. In this spirit, we invite the interested reader to follow the suggestion in the aforementioned paper, and further deepen their understanding by referring to the review papers [15] and [45], as well as the author's initial introduction to the field, Ref. [36].

1.2.1 Quantifying entanglement

A quantum state is called *entangled* if we cannot specify it by describing the components independently. Alternatively, a more mathematical description can be stated as follows. Suppose that a quantum system contains two distinct parts A and B. If we can write the state of the system as $|\phi\rangle = |\phi_A\rangle \otimes |\phi_B\rangle$, where $|\phi_A\rangle$ and $|\phi_B\rangle$ are the states of subsystems A and B, respectively, the state of the system is called *separable*. If $|\phi\rangle$ cannot be decomposed in this way, then A and B are *entangled*.

As an example, consider the state

$$\begin{aligned} |\varphi\rangle &= |+_A\rangle \otimes |+_B\rangle \\ &= \frac{1}{\sqrt{2}}\left(|0_A\rangle + |1_A\rangle\right) \otimes \frac{1}{\sqrt{2}}\left(|0_B\rangle + |1_B\rangle\right) \\ &= \frac{1}{2}\left(|0_A 0_B\rangle + |0_A 1_B\rangle + |1_A 0_B\rangle + |1_A 1_B\rangle\right). \end{aligned} \tag{1.27}$$

The state $|\varphi\rangle$ in this example is separable, since the two partitions of the system are in their respective $|+\rangle$ states. Conversely, we can consider the state

$$|\phi\rangle = \frac{1}{\sqrt{2}}\left(|0_A 0_B\rangle + |1_A 1_B\rangle\right). \tag{1.28}$$

Here, the state $|\phi\rangle$ cannot be separated as before. We therefore call this an entangled state.

Although technically correct, this definition, along with the paradigmatic examples given above, can be misleading. It insinuates a sort of binary relation in the sense that a system is either in an entangled state or in a separable (product) state. It thus does not obviously lead to a quantitative description of the degree of entanglement between A and B. To move beyond this limiting binary perspective, we can reframe entanglement through the lens of correlations, not unlike the correlations known from classical statistics, such that we can employ the toolset of classical statistics and information theory. That is, we can just as well consider entanglement in terms of possible measurement outcomes for the subsystems A and B for subsequent measurements on B or A. The quantum nature of entanglement then arises from the non-locality of these correlations.[4]

For a better grasp of this intuition, we return to the examples from above. Suppose we perform a computational basis measurement on qubit A, while the system is in the state described by Equation (1.27). This measurement yields either $+1$ or -1 as outcome with equal probability, $P(\pm 1) = 1/2$, and collapses the state of the qubit accordingly. A subsequent measurement of qubit B in the computational basis then turns out to be unaffected by the previous measurement; A and B are thus uncorrelated, i.e. unentangled. The probabilities of the measurement outcomes $+1$ or -1 on qubit B are still the same as before.

If we were to perform the same procedure on a system in the state described by Equation (1.28), this would no longer be the case. A measurement of qubit A in the computational basis would still yield the outcomes ± 1 with equal probability, $P(\pm 1) = 1/2$. However, the wave function now collapses in a fundamentally quantum manner: the measurement of one qubit instantaneously collapses the state of the other as well, since the total wave function collapses to one of $|0_A 0_B\rangle$ or $|1_A 1_B\rangle$. This yields a definitive outcome for a subsequent measurement of qubit B. For instance, if we measured $+1$, corresponding to $|0_A\rangle$, the state after the measurement is $|\phi'\rangle = |0_A 0_B\rangle$. A measurement of B would therefore also yield $+1$. Therefore, by measuring one of the constituents of the bipartite system, we have eliminated the uncertainty of the measurement outcome of the other.

This uncertainty can be quantified by the entanglement entropy. Formally, it quantifies the reduction of the statistical entropy of measurement outcomes of A by completely measuring subsystem B. We have already discussed the entropy of entanglement in the context of the stabilizer formalism in Section 1.1.3 with Definition 1.12. Here, we motivate it further and convince ourselves that the previously given definition is reasonable.

First, construct the density matrix ρ of the system. For a pure state $|\phi\rangle$, the density matrix

[4] Note that entanglement is not the only type of quantum correlation. There is a wide variety of correlations that are entirely non-classical in nature. This non-classicality is characterized by a measure called "quantum discord" [46]

is the projector onto $|\phi\rangle$,

$$\rho = |\phi\rangle\langle\phi| . \tag{1.29}$$

As the density matrix is an extension of (discrete) probability distributions over the state space of qubits, the density matrix of a pure state is just the state itself with probability 1. We can now marginalize over the measurement outcomes of measurements on a qubit. This is done by the partial trace operation and defines the reduced density matrix

$$\rho_A = \mathrm{Tr}_B[\rho]. \tag{1.30}$$

For the previous example of the entangled state $|\phi\rangle$ we have a density matrix

$$\rho = |\phi\rangle\langle\phi| = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} \tag{1.31}$$

with the rows and columns corresponding to the two-qubit basis vectors. The partial trace now corresponds to a trace over the 2×2 submatrices in the corners,

$$\rho_A = \mathrm{Tr}_B[\rho] = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{1.32}$$

Note that this reduced density matrix corresponds to a mixed state. It is not a coherent superposition of basis states, but rather a classical mixture of $|0_A\rangle$ and $|1_A\rangle$, both with probability $1/2$. The entanglement entropy is then the statistical entropy of this specific mixture. As already given in Definition 1.12, it is defined as

$$S_{\mathrm{E}}(|\phi\rangle) = -\mathrm{Tr}[\rho_A \log \rho_A], \tag{1.33}$$

which is equivalent to the von Neumann entropy of the subsystem density matrix ρ_A, which is in turn equivalent to the Shannon entropy [47, 48].

We remark that the entanglement relation between subsystems A and B is a symmetric one: asking how entangled A is with B yields the same degree of entanglement as asking how entangled B is with A. This symmetry resolves the seemingly contradictory notation in Definition 1.12 and Equation (1.33). It is the same phenomenon from either perspective. While that is the case for subsystems A and B once they have been fixed, the same level of agnosticism cannot be ascribed to the choice of subsystems. If the partitioning of A and B is ambiguous, the degree of entanglement is dependent on how one slices the system apart. Once one fixes a choice of A and B, the degree of entanglement between them is symmetric.

1.2.2 Random circuits

In the context of entanglement transitions, we often deal with unitary circuits, either consisting of randomly chosen unitary operators, or with unitaries randomly placed in a circuit layout (cf. Figure 1.2). This naturally raises two questions: why unitary circuits, and why should they, in any way, be random? In this subsection we try to answer both of these, starting with the first.

Why unitary circuits?

A unitary circuit is an alternative way next to the Schrödinger equation to describe the time evolution of a quantum state. While the Schrödinger equation describes the time evolution in continuous-time, unitary circuits are employed to describe the dynamics in discrete-time. That is, in a unitary circuit, the state evolves according to the unitaries applied at certain timesteps. As a visualization, consider the circuits depicted in Figure 1.2. From an initial product state (represented by the arrows pointing up) we have a discrete-time dynamics to some steady state.

Describing the time evolution of a system with a discretized time dimension can be motivated by a mutlitude of factors. The most straightforward motivation is the discrete time evolution by means of the application of unitaries inherent in quantum computers. Another reason is the *approximation* of continuous time dynamics. When simulating quantum systems on a classical computer, one only has access to a discrete set of timesteps. As such, a unitary circuit can function as an approximation of the continuous-time dynamics. An implementation of the circuit depicted in subfigure (b) of Figure 1.2 could, for instance, approximate the continuous-time dynamics of a one-dimensional spin chain with nearest-neighbor interactions.

We remark that the approximation of continuous-time dynamics can not only be motivated from a technical standpoint, but also from a mathematical one. As a matter of fact, the validity of this approximation is not as simple as it may seem. For some Hamiltonian with nearest-neighbor interaction, we may write $H = \sum_j H_j$ with local interactions H_j. In general, we have[5]

$$e^{-iHt} = e^{-i\sum_j H_j t} \neq \prod_j e^{-iH_j t}. \tag{1.34}$$

If we now consider the limit where $t \to \Delta t \to 0$, we obtain

$$e^{-iH\Delta t} = e^{-i\sum_j H_j \Delta t} = \prod_j e^{-iH_j \Delta t} + \mathcal{O}\left((\Delta t)^2\right). \tag{1.35}$$

[5] $\hbar = 1$

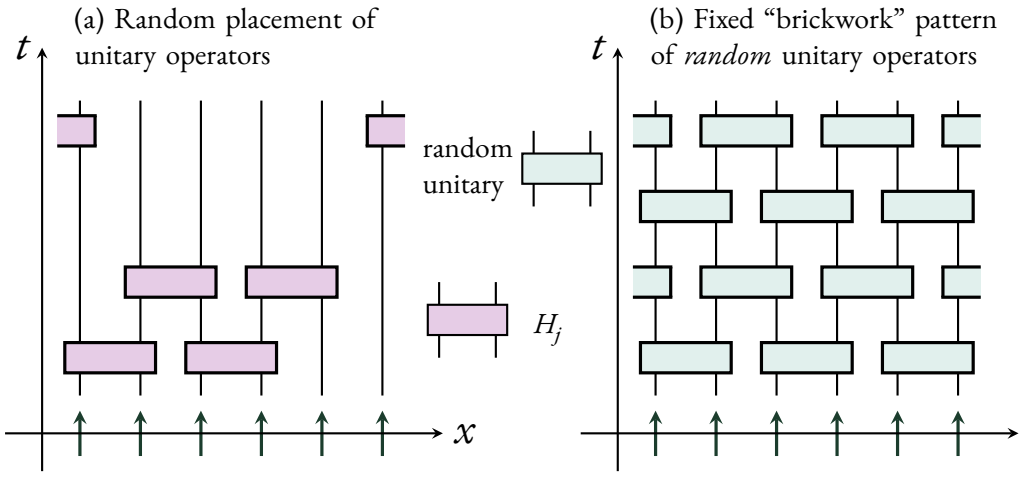


Figure 1.2.: Examples of unitary quantum circuits. Note that in the theory of random circuits and entanglement- or measurement-induced phase transitions, circuit diagrams are conventionally shown in time-position representation, instead of the usual position-time. The arrows on the respective x-axes represent an array of spins—or qubits—initially in a product state. Subfigure (a) shows a random circuit, where "random" refers to the placement of unitary gates in the space-time lattice. Subfigure (b) shows a random circuit with a fixed "brickwork" pattern, where random refers to the choice of unitaries, in that we choose unitaries from a given ensemble. The circuit diagrams are adapted from Fig. 1 of Ref. [44]

Therefore, the total time evolution of the many-body system *can* be represented as a product of local time evolutions, as long as one chooses to discretize the time evolution with sufficiently small Δt. The error of discretizing the time evolution of length T in steps of size Δt is of order $\mathcal{O}(\Delta t/T)$. Both the error and Equation (1.35) follow from the Trotter product formula [49], which is why this process of discretizing the dynamics is often referred to as "Trotterization".

The use of unitary circuits is therefore both motivated by the applicability for prospective quantum computers, as well as the simulation of quantum dynamics on a classical computer by means of Trotterization. Having answered the first question, we can now turn towards the question of why the unitary operations in the circuit should be random.

Why random?

Under the umbrella of "quantum dynamics" does of course fall the dynamics of entanglement, and the question of why the unitary circuit should in any way be random brings us back to exactly that (particularily the growth of entanglement). By choosing a specific

Hamiltonian to dictate the dynamics, we fix the time evolution to be somewhat streamlined to the specific system. (In a classical simulation, we would simulate the circuit in subfigure (b) of Figure 1.2 with fixed unitaries given by the Hamiltonian.) The motivation of randomizing the unitaries—or the placement thereof in the space-time grid (subfigure (a) of Figure 1.2)—is then that one aims to unveil *generic* entanglement dynamics, abstracted away from any particular Hamiltonian.

Furthermore, investigation of entanglement dynamics, in particular entanglement growth, is important for the simulation of quantum states on a *universal* quantum computer. For short-range entangled states there are efficient algorithms to simulate their behavior such as matrix product state simulations. For states with extensive entanglement, i.e. where the entanglement entropy scales with the system's volume, this is no longer the case. (The stabilizer formalism would be agnostic to this, which is why we specified the universal quantum computer.)

With the circuit layout given, e.g. subfigure (b) of Figure 1.2, we are now tasked with selecting for random unitary operators. That is, we need to specify an ensemble of unitaries from which we randomly choose. A sensible first choice would be to select uniformly from the set of all unitary matrices. This type of random unitary is referred to as *Haar-random* matrix. The dynamics of random circuits with Haar-random unitaries is the most generic type of dynamics. However, it is in general exponentially hard to simulate, as the dimensionality goes with 2^N for N qubits. As previously introduced, the Clifford group—generated by the Clifford gate set (see Definition 1.7)—constitutes a narrower ensemble of Restricting the set of operators in this way then allows for efficient classical simulations. As the peculiarities of the Clifford gate set and its connection to classical simulability have been discussed in other sections, we refer the reader to Sections 1.1.2, 1.1.3 and 4.1.1 and Theorem 1.14, and forego additional discussions thereof.

Naturally, there is more to the choice of random unitary circuits for the study of entanglement dynamics and entanglement growth, with excursions to the eigenstate thermalization hypothesis among other things. However, this would go far beyond the scope of this thesis. For our purposes it suffices that we have done our due diligence and convinced ourselves that the framework of random circuits is a practical one for the study of entanglement growth.

1.2.3 Measurement-Induced Phase Transitions

We have now discussed how entanglement can grow and spread in a system by means of random unitary operations. As a concrete example of a circuit that generates entanglement, recall the circuit depicted in Figure 1.1 and consider the first three (or two, if one chooses the middle qubit as control qubit for the second CNOT gate) "timesteps". If we had chosen our random unitaries from the Clifford gate set—or had gotten infinitely lucky with Haar-random matrices—this could have been one possible circuit. This created an entangled state over the three qubits, a 3-qubit Bell cluster.

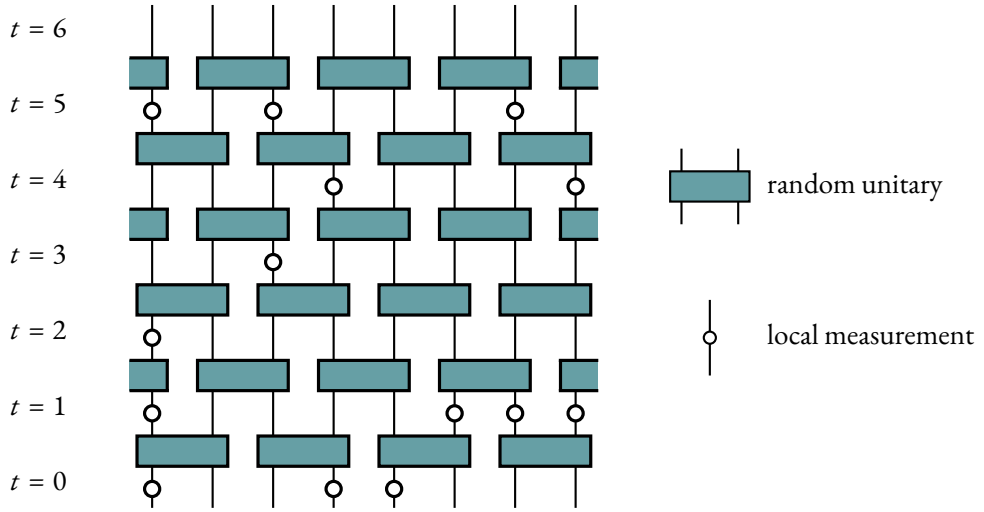


Figure 1.3.: Example of a random unitary circuit model studied in the context of measurement induced phase transitions with measurements performed with probability p at each site for each timestep. In this paradigmatic circuit, two-qubit unitary gates are arranged in a way resembling a brickwork wall, and local single-qubit measurement gates are randomly distributed. The circuit diagram is adapted from Fig. 1 of Ref. [36].

Let us now also consider the operation immediately following the second CNOT in Figure 1.1, a measurement gate on qubit 3. What was merely implied in the previous discussion of the circuit, we now state explicitly: the measurement gate removed the third qubit from the entanglement cluster, thus reducing the degree of entanglement in the system. This is in contrast to the preceeding gates, which created the cluster to begin with. (The fact that the syndrome measurements succeeding the local measurement restore the entanglement becomes important in the projective transverse-field Ising model.)

With these insights we can intuit the idea of the measurement-induced entanglement phase transition (MIPT): Unitary operators tend to increase the degree of entanglement in a system, while local measurements tend to decrease it. Two qubits can be entangled with a Hadamard and CNOT, while measuring either one disentangles the two. A natural question that arises is the following. How, if at all, is the entanglement dynamics altered if the circuit as depicted in subfigure (b) of Figure 1.2, is augmented by sporadic local measurements, such as depicted in Figure 1.3?

The answer to this question is, as is so often the case, "it depends". Specifically, it depends on how frequent the measurements are, that is, it depends on the measurement rate p. If there are no, or only a minimal amount of measurement gates, $p \ll 1$, the entanglement growth remains (qualitatively) unaffected, since we only rarely remove a qubit from an entanglement cluster. However, if there is a measurement gate for each qubit at every timestep, $p \to 1$, we

stun or effectively stop the growth of entanglement.

As it turns out these two extremes cases of the measurement frequency fall into two distinct *phases*, where there is a *phase transition* between them with a critical rate p_c. This is the measurement-induced entanglement phase transition. If measurements are rare, $p < p_c$, the system is in the *entangling phase*. Conversely, if measurements are sufficiently frequent, i.e. more frequent than the critical frequency p_c, the system is in the *disentangling phase* (also sometimes called "weak" and "strong monitoring" phases, respectively). The key features of the two phases are as follows.

Entangling phase: This phase is characterized by extensive entanglement when initializing the system with a (highly separable) pure state. The entanglement then grows linearly in time, proportional to the size of the subsystem.

Furthermore, if one has a mixed state initially, the state will most likely never be purified. Only in timescales exponential in the system size does the mixedness decay.

Disentangling phase: This phase is characterized by the absence of entanglement growth. A pure state will either never develop extensive entanglement if it is a product state initially, or will disentangle if initially entangled. A mixed state will find itself purified over an $\mathcal{O}(1)$ timescale.

The original problem of the MIPT concerned itself with the circuit of the form depicted in Figure 1.3, which is known as a hybrid circuit. The basic layout for L qubits and total time t is a brickwork pattern of Haar-random unitaries, which are followed by measurement gates applied on each site with a classical probability p. This probability is also referred to as the measurement rate. Once the procedure has concluded we quantify entanglement between two partitions of the total system of L qubits. The most relevant partitioning for the MIPT and also in the context of this thesis is the exact halving of the system. That is, one is primarily interested in the half-system entanglement entropy. (For the remainder of this thesis we refer to the half-system entanglement entropy, if not explicitly stated otherwise.) The phenomenology of the phase transition in this setup can be split in two limiting cases: infinite system size $L \to \infty$ with finite time t, and finite system size L in the steady state $t \to \infty$. We have

$$S(t, L \to \infty) \propto \begin{cases} t, & p < p_c \\ \log t, & p = p_c \\ \text{const.}, & p > p_c \end{cases} \tag{1.36}$$

for the limit of large sizes and

$$S(t \to \infty, L) \propto \begin{cases} L, & p < p_c \\ \log L, & p = p_c \\ \text{const.}, & p > p_c \end{cases} \tag{1.37}$$

for the steady-state limit. In the weak monitoring regime, we therefore have volume-law entanglement, since in one dimension the volume of a chain is its lenth. We then have logarithmic scaling at the critical point, where entanglement grows, but much slower than in the weak monitoring case. In the strong monitoring phase, the entanglement entropy is proportional to a constant, not scaling with system size or timescale. This is referred to as an area-law phase. Since the area is the boundary of a volume, and the boundary of a partition is just a single site (or two, in case of periodic boundary conditions), we have a constant.

There are three concluding notes to this section we do not want to fail to bring up. First, although the study of hybrid circuits, such as the one shown in Figure 1.3 was the initially studied setup, entanglement transitions have also been theorized in measurement-only circuits. We can convince ourselves that this theorizing is reasonable by considering the second half of the circuit shown in Figure 1.1. There, the initial entanglement *structure* is restored by a two-qubit projective measurement, after a perturbation in the form of a single-qubit projective measurement disentangled the cluster. While the state is not necessarily the same as before, we have the same distribution of entanglement among the qubits. [6]

Second, the general problem of a measurement-induced phase transition is not *entirely* quantum in its nature. Of course, we already established that entanglement is a quantum phenomenon. However, what we mean by that is that questions concerning the critical behavior of hybrid—or even measurement-only—circuits can oftentimes be mapped to classical problems that have an exact solution. For instance, one can understand the MIPT in hybrid circuits as minimal cut in the circuit diagram, where measurements "cut through" an entanglement structure generated by the application of unitaries. This in turn can be understood by a specific mapping to 2D percolation, which has already been studied with great rigor in the past (for more context on percolation theory, see Refs. [50–53]). These mappings allow for (a) bounds on the critical points of the phase transitions and (b) more substantiated predictions, supporting the ones one arrives at through quantum-mechanical considerations and classical simulations.

This then brings us to the third and final point. Notice that in the previous paragraphs, as well as the entirety of the present section, we have meticulously avoided to use the word "experiment". The reason behind this choice stems not from some grudge against experimentalists. On the contrary! Beyond all the beautiful theory of measurement-induced entanglement phase transitions hides a haunting spectre, known as the *sampling problem*[7]. This problem makes it difficult (not to say impossible) to observe these transitions in an experimental setting. Since one of the goals of this thesis is to circumvent the sampling problem in a specific circuit model (the projective transverse-field Ising model), we dedicate Section 1.4 to the details behind the sampling problem and continue with the introduction of the model system.

[6] For some nice visualizations of entanglement pairing and clustering, we refer to Figures 1 and 11 in Ref. [42].
[7] also called the *postselection problem*

1.3 THE PROJECTIVE TRANSVERSE-FIELD ISING MODEL

In the previous section we gave an overview of the field of entanglement transitions. However, phase transitions are a ubiquitous phenomenon all across theoretical physics. One paradigmatic example of a system exhibiting critical behavior is the Ising model, named after Ernst Ising, who, in his Ph.D. thesis, showed that there is no ferromagnetic ordering (i.e. no phase transition) in one dimension for finite temperatures [54]. Only for higher dimensions does the Ising model exhibit critical behavior. Following that, we have the transverse-field Ising model, which *does* have a phase transition in one dimension [19].

This all culminates into the model studied in this thesis, the *projective transverse-field Ising model* (PTIM) introduced in Ref. [1]. This model also has a phase transition, namely an entanglement transition. In this section we want to motivate the PTIM by going backwards through its name, and discuss the entanglement transition explicitly.

1.3.1 The Ising model

In this subsection, we briefly review the classical Ising model, following Refs. [54, 55]. Although it is often made subject in undergraduate courses on statistical mechanics, its importance for the paradigm of phase transitions cannot be understated. It is one of the simplest systems exhibiting critical behavior. Crucially this is *not* the case in one dimension. We now want to specifically highlight the absence of a phase transition in one dimension. To this end consider a chain of L classical spins. "Classical" in the sense that they are only allowed to point up or down. The Hamiltonian of the system reads[8]

$$H = -2J \sum_{i=1}^{L-1} S_i^z S_{i+1}^z \tag{1.38}$$

with $J > 0$, such that the ground state is obtained by all adjacent spins lining up. The energy of this state is thus $-NJ/2$, since $S_i^z = +1/2$ for all i in the chain. Consider now the case where a "mistake" is added to the system in the form of one spin flipping. This is associated with an energy cost of $\Delta E = J$, since we go from an energy saving $J/2$ contribution to $-J/2$. However, we did not specify where this defect appeared. It could have been on any of the L sites. Thus, there is an entropy gain of[9] $S = \ln L$. In the thermodynamic limit ($L \to \infty$), the cost of a defect is still J, whereas the contribution from the entropy becomes infinite. For our considerations, we have that the properties of the system are determined by the free energy

$$F = E - \beta^{-1} S \tag{1.39}$$

[8] Here, S_i^z is the z-component of the classical spin with $S_i^z = \pm 1/2$.
[9] $k_B = 1$

with $\beta = 1/T$. It is clear that for finite β (or non-zero temperatures) $F \to -\infty$ in the presence of defects. We therefore have that for temperatures above absolute 0, spin-flips will spontaneously occur, and there is no long-range order in one dimension. Equivalently, one can say that the critical temperature is $T = 0$. This consideration is valid for all one-dimensional lattices.

1.3.2 The transverse-field Ising model

The Ising model as conceived by Wilhelm Lenz is a model from classical statistical mechanics. The spins that constitute it can only take on discrete values, yes, but they cannot exist in a coherent superposition of up and down, but only up. Furthermore, it concerns itself with finite temperatures, $T > 0$, and not the quantum limit $T \to 0$. For a quantum mechanical analogue to the classical Ising model, we look towards the *transverse-field* Ising model, which we briefly want to introduce in this section, following its introduction in Ref. [19]. The transverse-field Ising model is especially interesting in the context of quantum phases and quantum phase transitions, where we examine ground states of the Hamiltonian, i.e. at 0 temperature. The phases and phase transitions are then with respect to parameters of the Hamiltonian, or more precisely, the coefficients of non-commuting terms within the Hamiltonian.

In the one-dimensional case for L spin-1/2 degrees of freedom, the Hamiltonian of the transverse-field Ising model (TIM) reads

$$H = -J \sum_{i=1}^{L} \sigma_i^z \sigma_{i+1}^z - h \sum_{i=1}^{L} \sigma_i^x, \tag{1.40}$$

where we choose $J > 0$ (ferromagnetic coupling), periodic boundary conditions ($L + 1 \equiv 1$), and a magnetic field $h > 0$. With positive h, the magnetic field it points in x-direction, which is transverse to the z-direction, hence the name *transverse-field*. Notice that in Equation (1.40), the terms $\sigma_i^z \sigma_{i+1}^z$ and σ_i^x do not commute, $[\sigma_i^z \sigma_{i+1}^z, \sigma_i^x] \neq 0$. Thus, they do not have a shared eigenbasis, and the ground state of the Hamiltonian will in general be a superposition of eigenstates of one of the operators. Conventionally, one chooses the basis of $\sigma_i^z \sigma_{i+1}^z$. Note that the ferromagnetic coupling implies aligned spins, where nearest neighbours tend to share directions, while the transverse magnetic field will flip some spins arbitrarily in the evolution.

Let us now consider the two limits of $J \ll h$, i.e. $J \to 0$, and $h \ll J$, i.e. $h \to 0$. In the former we have a weaker coupling between the spins and thus a higher tendency for individual spins to flip. The ground state is then the product state of local ground states of σ_i^x, $|G_+\rangle = |+ + \cdots +\rangle$. This ground state is unique, since each constituent is locally a unique ground state. One further feature of it is that is possesses no long-range order. The spin on site i is completely agnostic to the happenings of the spin on site j—if i and j are sufficiently

far apart—, i.e. i and j are on average not aligned (similar to the classical ising model for higher temperatures). Mathematically, this expresses as

$$\lim_{|i-j|\to\infty} \langle G_+ | \sigma_i^z \sigma_j^z | G_+ \rangle \to 0. \tag{1.41}$$

The converse case turns out to be rather different. Here, we have that the coupling is strong, and thus the spins tend to align. Crucially however, alignment is the only quality of relevance in the context of the ground state, since the +1 eigenvalue of $\sigma_i^z \sigma_{i+1}^z$ is two-fold degenerate with the eigenstate spanned by $|00\rangle$ and $|11\rangle$. The full Hamiltonian then inherits this degeneracy, such that its ground state space is spanned by two states $|G_0\rangle = |00\cdots0\rangle$ and $|G_1\rangle = |11\cdots1\rangle$. The ground state $|G\rangle$ is then an arbitrary coherent superposition of $|G_0\rangle$ and $|G_1\rangle$,

$$|G\rangle = \alpha\,|G_0\rangle + \beta\,|G_1\rangle\,. \tag{1.42}$$

Note that we here do have long-range order, since we are in the regime, where ferromagnetic coupling is strong. This corresponds to the 0-temperature limit of the classical Ising model. Analogously to the paramagnetic phase we can write

$$\lim_{|i-j|\to\infty} \langle G | \sigma_i^z \sigma_j^z | G \rangle \to 1. \tag{1.43}$$

Note that σ_i^z thus corresponds to a local order parameter for the ferromagnetic phase. Once we were in the paramagnetic regime, we had $\left\langle \sigma_i^z \sigma_j^z \right\rangle \to 0$ in the long-range limit. A natural question to follow is, what happens between the regimes of $h/J \to 0$ and $h/J \to \infty$? The answer is, of course, that there is a phase transition. Remarkably, this phase transition happens already in the one-dimensional case, contrary to the classical Ising model which did not have a phase transition in one dimension. To understand what we mean by a phase transition, consider the symmetry group of the Hamiltonian, $\mathcal{G}_H = \{I, X\} \cong \mathbb{Z}_2$ with $X = \prod_i \sigma_i^x$, which corresponds to flipping all spins. In the paramagnetic phase, the ground state shares this symmetry, since $X\,|G_+\rangle = |G_+\rangle$. In the ferromagnetic phase, we do not have this symmetry in general, since flipping all spins would transform one ground state into the other. Thus, the symmetry of the Hamiltonian is spontaneously broken. This breaking of a symmetry corresponds to a phase transition according to the Landau paradigm of phase transitions [18].

1.3.3 The projective transverse-field Ising model

We now have the baseline set to introduce the *projective* transverse-field Ising model. Our detailed introduction follows Ref. [1]. The initial setup of the model is similar to the TIM, in that we have a linear chain of L spin-1/2 degrees of freedom on sites $i \leq L$. However, in the PTIM we do not consider time evolution generated by a Hamiltonian, such as the

one in Equation (1.40), but rather as time evolution in discrete steps of a quantum circuit consisting of projective measurements, as discussed in Section 1.2.2. In fact, the PTIM is an example of a circuit as depicted in subfigure (a) of Figure 1.2, with projective measurements instead of unitary measurements. The action of a measurement of observable O on a state $|\phi\rangle$ is given by

$$\mathcal{M}[O](|\phi\rangle) = \frac{\mathbb{P}_\lambda |\phi\rangle}{\sqrt{\langle\phi|\mathbb{P}_\lambda|\phi\rangle}} \tag{1.44}$$

with the eigenvalue λ of O and probability $P(\lambda; |\phi\rangle) = \langle\phi|\mathbb{P}_\lambda|\phi\rangle$.

The connection to the TIM is through the choice of observables O and the measurement scheme; by choosing observables σ_i^x and $\sigma_i^z\sigma_{i+1}^z$ for each site i and edge between adjacent sites $e = (i, i+1)$, we have the same non-commuting observables that constitute the Hamiltonian in Equation (1.40). Repeated measurements on the same site thus lead to nontrivial dynamics.

As initial state, we choose the product state of $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$,

$$|\phi(0)\rangle = |+ + \cdots +\rangle\,. \tag{1.45}$$

The circuit governing the (discrete) time evolution is then constructed as follows. (This is also the protocol for circuit generation in upcoming numerical experiments, i.e. simulations.) For each time step, we go through each site and set the site variable x_i to 1 with probability p, and 0 otherwise. Independently from this we set the edge variable $z_e = 1$ with probability $1 - p$, and 0 otherwise. These vectors correspond to the sites and edges where X and ZZ measurements will be performed in the circuit. In each timestep we perform the corresponding measurements, thus evolving the state. An example of a circuit generated by such a protocol is shown in Figure 1.4 for $p = 1/2$. The full circuit defines a (discrete time) quantum trajectory $|\phi(t)\rangle$.

With this dynamics of the system, we can probe different quantities, such as the entanglement entropy $S_E(A)$ of a subsystem A, over many trajectories. Averaging them defines the sample average,

$$X = \frac{1}{M}\sum_{|\phi\rangle\in N} \mathcal{X}(|\phi\rangle), \tag{1.46}$$

where N is a set of M randomly generated trajectories. Note that we typically fix a parameter p and a time t when sampling specific quantities, where the circuit generation and application defines a classical probability distribution.

In spirit of the TIM, we are interested in (sample averaged) quantities as a function of the probability parameter p characterizing the relative strength between the non-commuting observables. Varying p in the interval $[0, 1]$ is thus analogous to the limits of $h/J \to 0$

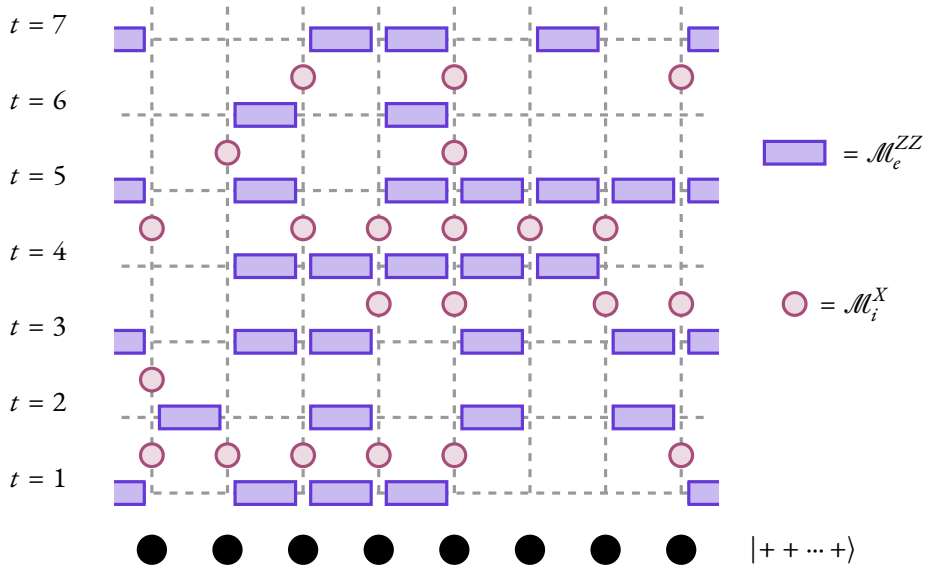


Figure 1.4.: Example of a PTIM circuit for $p \approx 0.5$ for a chain of $L = 8$ spins with periodic boundary conditions. The rectangles on horizontal edges of the grid, $e = (i, i+1)$, denote ZZ measurements, $\mathcal{M}_e^{ZZ}$, while the circles on vertical edges of the grid, i, denote X measurements, $\mathcal{M}_i^X$.

or $h/J \to \infty$ in the TIM. As the example most relevant to entanglement transitions, we highlight the behavior of the entanglement entropy as a function of p. This is shown in Figure 1.5.

Comparing Figure 1.5 to the phenomenology discussed in Section 1.2.3, there are two conspicuous features of the entanglement entropy in the PTIM we want to highlight in particular. First, we can see that the entropy in the disentangling phase, where X measurements are more frequent, has an area-law scaling, manifest in the fact that it is independent of the chain length, i.e. the volume. Additionally, the scaling at the critical point is logarithmic. This fact is highlighted by the choice of chain lengths. Doubling the length of a chain scales the entanglement entropy linearly at the critical point.

Secondly, we observe that the entangling phase now also has an area-law scaling as well. This can be explained by the fact that in the $p \to 0$ limit, the initial product state tends to get entangled in a singular cluster, characterized by unit entropy.

As a final note we interpret the critical behavior of the entanglement entropy shown in Figure 1.5 in the context of quantum error correction. The specific choice of observables is not only interesting in their connection to the TIM. Recall the circuit shown in Figure 1.1. If we have an entangled state over multiple qubits, we can think of ZZ measurements as stabilizer (syndrome) measurements and projective X erors in the circuit. Due to the projective

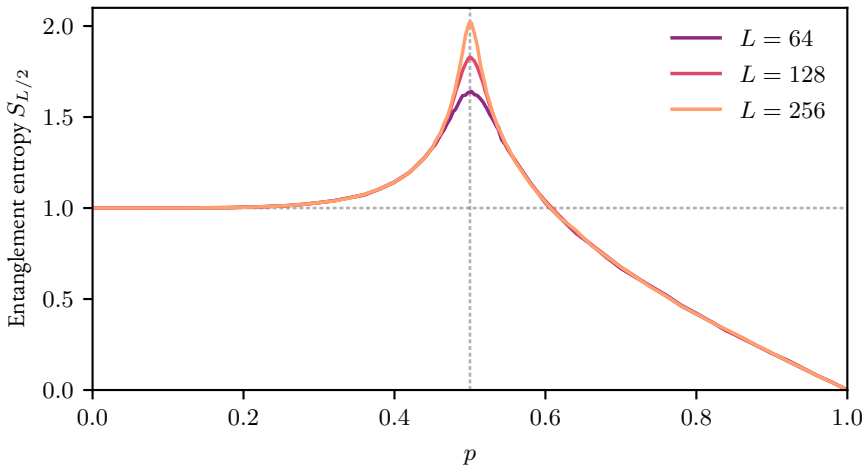


Figure 1.5.: Half-system entanglement entropy in the steady-state limit, $S(L/2, t \to \infty)$, shown as a function of the probability parameter p for systems of lengths $L \in \{64, 128, 256\}$. The dashed horizontal line corresponds to the area-law entanglement in the regime of $p \ll p_c$. The dashed vertical line corresponds to the critical probability $p_c = 1/2$. At the critical point, the entanglement grows logarithmically with the system size, analogous to the MIPT in hybrid circuits (see Equation (1.37))

dynamics, we not only detect, but also sometimes correct the error. More importantly, the entanglement structure stays the same no matter what. (This is also manifest in the fact that the local observables commute with the symmetry operator $U = \prod_i X_i$, which generates a global $\mathbb{Z}_2$ symmetry.) For $p \to 0$, we thus have a lot of effective entangling gates, which are stabilized by sufficiently frequent ZZ measurements. As a consequence, each qubit is entangled with every other qubit, giving a half-system entanglement entropy of 1 (as there is only one independent Bell pair).

For $p \to 1/2$ we approach the critical point, where entanglement clusters are sporadically created and merged together. The dynamics at this point is nontrivial, and can be characterized by a mapping to a conformal field theory. Interestingly, the PTIM and its phase transition is identical to two-dimensional percolation, which is analytically solvable and has a well-known critical point of $p_c = 1/2$. The interested reader is invited to consult Refs. [53, 56] for further details on percolation and conformal field theories, which go beyond the scope of this work. For now, we can see that the emergence of new entanglement clusters peaks at $p = 1/2$.

In the $p \to 1$ regime, we do not stabilize any entanglement, but rather have a product state of X eigenstates. The more frequent X measurements are (and conversely, less frequent ZZ measurements), the more we stabilize a product state.

1.4 SAMPLING PROBLEM

The metaphysics of this endeavour (i.e. this thesis) can be condensed in the following way; we (a) know from experiments how quantum systems behave under certain conditions, and (b) predict through theoretical calculations what these systems might do in another experimental setting. In the latter case however, there is an uncanny regime of utility, where we either (a) cannot precisely pass predictions, (b) cannot perform the experiment on the grounds of hardware limitations[10], or (c) try to predict the behavior of quantities not directly measurable. In the field of quantum information, and especially in entanglement transitions, we face the latter two of these bottlenecks in increasing severity. The main focus of this thesis, however, is the overcoming of the last problem, which lies at the heart of the so-called *sampling problem*.

The measurement-induced phase transitions theorized in random circuits, in particular also the projective transverse-field Ising model, is only a feature of the average over many post-selected pure states. That is, from a given measurement record $\mathbf{m}$, we can construct a density matrix

$$\rho_{\mathbf{m}} = \left|\phi_{\mathbf{m}}\right\rangle\!\left\langle\phi_{\mathbf{m}}\right| . \tag{1.47}$$

The entanglement entropy of this state is then given by Equation (1.33). In principle, determining a density matrix in an experiment can be performed by tomographical means, such as shadow tomography [58]. However, tomographical methods require the same state to be prepared—otherwise one would obviously probe an entirely different state. On an entirely classical platform[11], this would merely be a technical issue, since we could then, by fixing the positions of measurement gates, eliminate the element of stochasticity under our control, and prepare the state deterministically.

Unfortunately, quantum mechanics is an intrinsically probabilistic theory: the outcome to each measurement of a measurement gate is given by the Born rule. Thus, in a circuit with N measurement gates, the probability of obtaining a specific measurement record is $\mathcal{O}\left(2^N\right)$.[12] Therefore, even with the measurement gates fixed in place one would—on average—require exponentially many runs of the same circuit. This, in conjunction with the tomography problem, makes detecting an entanglement *transition* almost insurmountably difficult to detect.

Not that this has not been tried. See Refs. [60–62] for attempts to observe the phase transition in an experimental setting. However, these experiments are not even remotely realizing the steady-state limit, but are restricted to short times and small systems (fewer lattice sites in the space-time grid imply a lower number of measurement gates). Thus, one

[10] The Higgs particle was predicted 40 years before it was discovered [57]

[11] such as a stabilizer simulator

[12] One could say that the problem of preparing the same quantum state twice in a random hybrid or measurement-only circuit is in EXPTIME (see [59] and the beginning of Chapter 4)

of the greatest problems in the field of entanglement transitions is to find a way around the sampling problem to detect the phase transition.

While there are other measures for probing entanglement, also employed in an experimental context [63, 64], we specifically investigate two approaches in the context of the entanglement transition in the PTIM: the linear cross entropy as introduced in [2], as well as the approach of Garratt and Altman in [3], which is based on Klein's inequality, bounding the entanglement entropy from above by the cross entropy. In the following we will briefly summarize the basic ideas behind the approaches.

Linear Cross Entropy

The linear cross entropy was originally introduced to act as an order parameter to distinguish between the volume law and area law phases. The idea is to compare two distinct initial states for the same circuit. In the volume law phase, they cannot be distinguished, while in the area law phase they can. The linear cross entropy is then a quantifier of the distinguishability of the two initial states, where one has unit linear cross entropy in the volume law phase and a linear cross entropy smaller than 1 in the area law phase.

The advantage of this is that one does not need a post-processing step. If we perform an experiment on a quantum simulator, then compare it with a classical simulation, where we pipe the measurement outcomes to the classical simulator, we do not require a post-processing step on the quantum level.

The precise mathematical definitions and properties of the linear cross entropy are given in Section 2.1.

Upper Bound

Contrary to the previous idea, the approach suggested in Ref. [3] is not a precise order parameter for the phase transition, but rather relies on bounding the entanglement entropy from above. To this end, they derive a relation incorporating shadow tomography that relates the average over many runs of an experiment to the quantum mechanical expectation value. For a set of matrices dependent on the measurements, W_m, and the shadow of a density matrix ρ_r^S they obtain

$$\mathbb{E}_r[W_{m,r}\rho_r^S] = \mathbb{E}_m[W_m\rho_m], \tag{1.48}$$

with the average over experimental realizations $\mathbb{E}_r[\bullet]$ and the average over all measurement outcomes $\mathbb{E}_m[\bullet]$, weighted by the Born probabilities P_m. For the detailed derivation of Equation (1.48), see the original work, Ref. [3].

If one now chooses $W_m = -\log \sigma_m$, where σ_m is a classical estimate for the post-measurement density matrix, one can employ Klein's inequality to bound the measurement-averaged entanglement entropy,

$$\mathbb{E}_m[S_{E,m}] \leq \mathbb{E}_r[S_r^{SC}], \tag{1.49}$$

where S_r^{SC} is the cross entropy between the shadow and the classical estimate.

2

LINEAR CROSS ENTROPY

"Check the circuit"
Spock to helmsman; Star Trek: The Original Series — Season 1, Episode 1

In this chapter we examine a measure known as the linear cross entropy proposed by Li et al. in [2] to experimentally access measurement-induced phase transitions. Despite its novelty, it has already been employed in experiments to probe hybrid quantum circuits [62], and other theoretical works are based on it. Even for our model system, the projective transverse-field Ising model (PTIM), a study has been conducted (see Ref. [4][1]). With this in mind we thus build on the work of Tikhanovskaya et al. in [4]. In particular, we reproduce their findings, provide a more detailed insight into the features and technicalities of the linear cross entropy, and further expand on their work by introducing more general types of noise. We thereby aim to detect the theorized measurement-induced phase transition in the projective transverse-field Ising model in the best case, or gain a new understanding from failure in the worst case.

The chapter is structured as follows. We will first introduce the quantity itself in Section 2.1, where we also provide a shortcut for computing the LXE in clifford circuits. We then investigate the behavior of it in the PTIM, comparing it to the entanglement entropy of an ancilla qubit attached to the system. Next, we introduce faulty gates to the process and examine the LXE for a system with errors. Lastly, we motivate the discarding of either X or ZZ measurement results with probability-theoretic arguments and evaluate the utility of the LXE for the different cases.

[1] While this reference points to a preprint on arXiv, we remark that the cited work has since been published in Physical Review B with no significant alterations (see Ref. [66]).

2.1 DEFINITION AND PROPERTIES

In this section we provide a definition for the linear cross entropy χ (also abbreviated as LXE) and a detailed derivation for its computation in Clifford circuits. To this end, we follow the results of Ref. [2, 4] for the generalities, and provide our own derivations and proofs for the particularities surrounding the projective transverse-field Ising model.

Before introducing all the technical details surrounding the linear cross entropy, we want to give a quick remark on the motivation behind the linear cross entropy. In Ref. [2], they mention that eariler studies found a quantity closely related to $-\ln\chi$ corresponding to the free energy cost after fixing a boundary condition in a replicated spin model [2, 37, 67–69]. The linear cross entropy is thus also motivated from classical statistical physics.

2.1.1 General random circuits

We first introduce the linear cross entropy (also abbreviated as LXE) in the context of general random quantum circuits. We then provide a concrete description and expression of the LXE for our system, the projective transverse-field Ising model.

Consider a hybrid circuit with some cicruit layout C, consisting of unitary and measurement gates. The measurement gates within the circuit yield results m_i, which are collected in the sequence $\mathbf{m} = \{m_1, m_2, \dots, m_N\}$, where N is the number of measurement gates. We call this sequence of measurement outcomes the *measurement record*. For some initial state ρ of our hybrid circuit, we can compute the *unnormalized* output state as

$$\rho_{\mathbf{m}} \equiv C_{\mathbf{m}} \rho C_{\mathbf{m}}^{\dagger}. \tag{2.1}$$

Here, $C_{\mathbf{m}}$ is the time-ordered product of all gates in the circuit, unitary and measurement, and can be written as [2]

$$C_{\mathbf{m}} = \mathbb{P}_{m_N} \mathbb{P}_{m_{N-1}} \cdots \mathbb{P}_{m_{N-N_T+1}} U_T \mathbb{P}_{m_{N-N_T}} \cdots \mathbb{P}_{m_{N-N_T-N_{T-1}+1}} U_{T-1} \cdots . \tag{2.2}$$

The action of $C_{\mathbf{m}}$ on the initial state ρ is the mathematical object representing one possible trajectory with measurement outcomes $\mathbf{m}$. Leaving the state unnormalized after projection operations is a deliberate choice; it allows us to define the probability of a certian measurement record.

Definition 2.1 (Probability of a trajectory)**.** Given a random circuit C and an initial state ρ with corresponding measurement record $\mathbf{m}$, the probability of $\mathbf{m}$ is denoted by

$$P\left(\mathbf{m} \mid C, \rho\right) \equiv \mathrm{Tr}\left[\rho_{\mathbf{m}}\right] = \mathrm{Tr}\left[C_{\mathbf{m}} \rho C_{\mathbf{m}}^{\dagger}\right]. \tag{2.3}$$

Although we do not want to delve too deep into probability theory at this point, let us nonetheless first convince ourselves that Definition 2.1 really does define a proper probability measure. To this end, consider the simplest example of a circuit with initial state $\rho = |+\rangle\langle+|$ and a single measurement in the Pauli-Z basis. By Equation (2.1) our unnormalized output state reads

$$\rho_{\mathbf{m}} = \mathbb{P}_m |+\rangle\langle+| \mathbb{P}_m^\dagger = \begin{cases} \frac{1}{2} |0\rangle\langle 0| , & m = 1 \\ \frac{1}{2} |1\rangle\langle 1| , & m = -1 \end{cases} \tag{2.4}$$

with $\mathbb{P}_m = \frac{1}{2}(\mathbb{1} + mZ)$. Computing the trace then yields $\frac{1}{2}$ for the probability of each trajectory. This is also consistent with our expectation from elementary quantum mechanics. As projectors sum to the identity (completeness), all the probabilities sum to 1, as they should. Furthermore, since unitaries leave the norm invariant, the trace only picks up projections. However, one needs to be careful here. Applying a Hadamard gate to $|+\rangle\langle+|$ transforms the state, and the probabilities change.[2]. Conversely, applying a Hadamard gate to any of the output states in Equation (2.4) does not alter the trace, but rather the output state itself.

We will later continue this discussion for the measurement-only circuit of the PTIM. For now, we employ Definition 2.1 and define the circuit-level linear cross entropy $\chi_C(\rho, \sigma)$.

Definition 2.2 (Linear cross entropy)**.** Given a random circuit C with two distinct initial states ρ and σ, and measurement records $\mathbf{m}$, the normalized linear cross entropy is defined as

$$\chi_C = \sum_{\mathbf{m}} P(\mathbf{m} \mid C, \rho) \frac{P(\mathbf{m} \mid C, \sigma)}{\sum_{\mathbf{m}'} \left(P(\mathbf{m}' \mid C, \sigma)\right)^2}, \tag{2.5}$$

where the sums $\sum_{\mathbf{m}}$ go over all possible measurement outcomes of the measurement gates in the circuit.

Most times when dealing with random circuits, we want to investigate numerous different circuit realizations for a given parameter (usually probability). Hence, when referring to the linear cross entropy, we will mostly refer to the circuit-averaged linear cross entropy,

$$\chi \equiv \langle \chi_C \rangle_C . \tag{2.6}$$

We will later also see the utility of defining χ and χ_C this way, especially concerning the denominator in Equation (2.5). For now it suffices to remark that it is a normalizing factor such that the linear cross entropy is bounded from above by 1, $0 \leq \chi \leq 1$.

We can interpret this quantity like a classical fidelity between two probability distributions. For two given initial states, we have a probability distribution over measurement records.

[2]The probabilities for the outcomes of a Pauli-Z measurement become 1 and 0

For each measurement record, Definition 2.1 provides a way to quantify the probability. For these probabilities, the linear cross entropy quantifies the overlap of one probability distribution with the other. If $\chi = 1$, the probability distributions are identical, and if we chose different initial states, a unit linear cross entropy tells us that this circuit (or parameter p) is not suited to distinguish between the initial states ρ and σ. On the other hand, if $\chi = 0$, we have no measurement record, which is compatible with ρ and σ simultaneously.

2.1.2 Clifford circuits and PTIM

Now that we have introduced the general concept of the linear cross entropy and defined the relevant quantities formally, we will examine it more closely in the context of the projective transverse-field Ising model. As we are able to simulate the PTIM with a stabilizer simulator, we will also introduce a way to compute the LXE in Clifford circuits, which will be implemented in the simulator.

In Equation (2.5), the choice of placement of the individual probabilities is a deliberate one. It is to highlight the fact that we are averaging a quantity depending on σ over a probability distribution that depends on ρ. As a means of cirumventing the sampling problem, we'd like σ to correspond to a classical simulator and ρ to the experiment. It will therefore prove useful to examine the part dependent on σ more closely, as it will later enable us to efficiently compute the linear cross entropy. As intermediate step we first derive an expression for the probabilities $P\left(\mathbf{m} \mid C, \rho\right)$, laid out in Lemma 2.3.

Lemma 2.3 (Probability of a trajectory in measurement-only Clifford circuits)**.** Given a Clifford circuit C with N measurement gates and no other gates, with initial stabilizer state ρ, the probability of a measurement record is

$$P\left(\mathbf{m} \mid C, \rho\right) = 2^{-N_{\text{rand}}}, \tag{2.7}$$

where $N_{\text{rand}} \leq N$ is the number of random outcomes for the measurement gates.

Proof. Suppose we have a Clifford circuit C with N measurement gates and an n-qubit initial stabilizer state ρ. We already know from Section 1.1.3 that there are two types of measurement outcomes, random or deterministic. Each measurement potentially has 2 outcomes, and in the former case each outcome has probability $1/2$ of occuring. (The other outcomes, the deterministic ones, trivially have unit probability.) Furthermore, we know that ρ can be expressed as the product of projectors (see Definition 1.11)

$$\rho = \frac{1}{2^n} \prod_{i=1}^{n} (I + g_i) \tag{2.8}$$

with generators g_i, or equivalently as a sum of *all* stabilizer group elements

$$\rho = \frac{1}{2^n} \sum_{g \in \mathcal{S}(\rho)} g. \tag{2.9}$$

If we now subject ρ to the circuit C, we obtain a measurement record $\mathbf{m}$. By Definition 2.1 we compute the probability by tracing the unnormalized output state (see Equation (2.1)), which in turn is obtained by application of the respective projectors on the input state, i.e.

$$\rho_{\mathbf{m}} = C_{\mathbf{m}} \rho C_{\mathbf{m}}^{\dagger} = T\left\{\prod_i^N \mathbb{P}_i\right\} \rho \, T\left\{\prod_i^N \mathbb{P}_i\right\}^{\dagger}. \tag{2.10}$$

Here, $T\{\bullet\}$ is the time-ordering operator and $\mathbb{P}_i$ is the projection operator on measurement outcome i. The probability of some measurement record is then given by

$$P\left(\mathbf{m} \mid C, \rho\right) = \mathrm{Tr}\left[C_{\mathbf{m}} \rho C_{\mathbf{m}}^{\dagger}\right], \tag{2.11}$$

which can be written as

$$P\left(\mathbf{m} \mid C, \rho\right) = \mathrm{Tr}\left[T\left\{\prod_i^N \mathbb{P}_i\right\} \rho\right], \tag{2.12}$$

with the cyclic property of the trace and $\mathbb{P}^2 = \mathbb{P}$. We can now consider the successive application of measurement gates. We know that each of the N measurements is either random or deterministic. Let us therefore consider these two cases separately.

Case 1 – Deterministic outcome: If a measurement has a deterministic outcome, the measurement operator was already part of the stabilizer group. We can therefore construct a set of commuting operators $\langle g_i \rangle$, containing the measurement operator M, which generate the stabilizer group. If we let $g_1 = M$ w.l.o.g., we have

$$\begin{aligned} P\left(\mathbf{m} \mid C, \rho\right) &= \mathrm{Tr}\left[\mathbb{P}_N \cdots \mathbb{P}_i \rho\right] \\ &= \frac{1}{2^{n-1}} \mathrm{Tr}\left[\mathbb{P}_N \cdots \frac{1}{2}(I + M) \frac{1}{2}(I + g_1) \prod_{i=2}^{n} (I + g_i)\right] \\ &= \frac{1}{2^{n-1}} \mathrm{Tr}\left[\mathbb{P}_N \cdots \frac{1}{2}(I + M) \prod_{i=2}^{n} (I + g_i)\right] \\ &= \frac{1}{2^n} [\mathbb{P}_N \cdots \mathbb{P}_{i+1} \prod_{i=1}^{n} (I + g_i)], \end{aligned} \tag{2.13}$$

where we used the fact that $\mathbb{P}^2 = \mathbb{P}$. The last line of Equation (2.13) tells us that deterministic measurements do nothing on the state and the probability.

Case 2 – Random outcome: In the case of a random outcome, the operator to be measured was not in the stabilizer group. Technically, it should be replaced, which is done by

conjugation with the projector. However, it turns out that for the probability, we only need to consider a single projection. It is

$$P\left(\mathbf{m} \mid C, \rho\right) = \mathrm{Tr}\left[\mathbb{P}_N \cdots \mathbb{P}_i \frac{1}{2^n} \sum_{g \in \mathcal{S}(\rho)} g\right]$$
$$= \mathrm{Tr}\left[\mathbb{P}_N \cdots \frac{1}{2}\left(\frac{1}{2^n} \sum_{g \in \mathcal{S}(\rho)} g + \sum_{g \in \mathrm{Stab}(\rho)} Mg\right)\right]. \tag{2.14}$$

Since the second sum is exclusively one over Pauli matrices, which are traceless, they don't contribute to the total. We thus have

$$P\left(\mathbf{m} \mid C, \rho\right) = \mathrm{Tr}\left[\mathbb{P}_N \cdots \mathbb{P}_{i+1} \frac{1}{2^{n+1}} \sum_{g \in \mathcal{S}(\rho)} g\right]. \tag{2.15}$$

Combining cases 1 and 2, knowing that measurements with deterministic outcomes don't alter the state at all, we have

$$P\left(\mathbf{m} \mid C, \rho\right) = \mathrm{Tr}\left[\frac{1}{2^{n+N_{\mathrm{rand}}}} \sum_{g \in \mathcal{S}(\rho)} g\right]$$
$$= \frac{1}{2^{N_{\mathrm{rand}}}} = 2^{-N_{\mathrm{rand}}}. \tag{2.16}$$

□

We can now use this result to find a nice expression for the linear cross entropy in Clifford circuits, and in particular the projective transverse-field Ising model.

Theorem 2.4 (LXE for Clifford circuits)**.** Let C be a Clifford circuit with N measurement gates and a measurement record $\mathbf{m}$ obtained from a realization of C with some initial state ρ. Furthermore, let $\sigma \neq \rho$ be a different state, where $\mathbf{m}'$ are all possible measurement records from applying C on σ. Then

$$\frac{P(\mathbf{m}|C,\sigma)}{\sum_{\mathbf{m}'}\left(P(\mathbf{m}'|C,\sigma)\right)^2} = \begin{cases} 1 & \sigma \text{ is compatible with } C \text{ and } \mathbf{m} \\ 0 & \sigma \text{ is not compatible with } C \text{ and } \mathbf{m} \end{cases}. \tag{2.17}$$

Proof. We can consider two (disjoint) cases:

1. No projection in $C_{\mathbf{m}}$ is orthogonal to σ, meaning that σ is compatible with C and $\mathbf{m}$.

This implies that the trace in the computation of $P(\mathbf{m} \mid C, \sigma)$ will always be strictly above 0. In fact, the probability will be $2^{N_{\text{rand}}}$, since each projection was successful. Together with the fact that there are $2^{N_{\text{rand}}}$ measurement gates we have

$$\frac{P(\mathbf{m}|C,\sigma)}{\sum_{\mathbf{m}'}\left(P(\mathbf{m}'|C,\sigma)\right)^2} = \frac{2^{-N_{\text{rand}}}}{\sum_{\mathbf{m}'} 2^{-2N_{\text{rand}}}} = \frac{2^{-N_{\text{rand}}}}{2^{-N_{\text{rand}}}} = 1. \tag{2.18}$$

2. At least one projection in $C_{\mathbf{m}}$ is orthogonal to σ, meaning that σ is not compatible with C and $\mathbf{m}$.

   Here we have at least one projection in $C_{\mathbf{m}}$, which projects σ onto the 0-vector. Incidentally, we require *exactly* one projection to be orthogonal, since we then have no object that can be projected to anything. Thus, the trace becomes 0, and as a consequence we have

$$\frac{P(\mathbf{m}|C,\sigma)}{\sum_{\mathbf{m}'}\left(P(\mathbf{m}'|C,\sigma)\right)^2} = 0. \tag{2.19}$$

□

The second case in the proof of Theorem 2.4 is an interesting one to consider, especially in the context of the PTIM. Suppose we run a PTIM circuit on some n-qubit platform with an initial state of $|\phi_0\rangle = (|0\cdots 0\rangle + |1\cdots 1\rangle)/\sqrt{2}$ or $\rho = |\phi_0\rangle\langle\phi_0|$. This produces a measurement record, which we then feed into a classical simulator with an orthogonal Bell state $|\varphi_0\rangle = (|0\cdots 0\rangle - |1\cdots 1\rangle)/\sqrt{2}$ or $\sigma = |\varphi_0\rangle\langle\varphi_0|$. There are of course other orthogonal Bell states we could choose as initial state for our simulator. However, these two states are additionally orthogonal in the logical basis. That is, if we encode information in $|\phi_0\rangle$, the opposite information is encoded in $|\varphi_0\rangle$. That this choice is a reasonable one gets even more apparent when considering generating sets of the respective stabilizers,

$$\mathcal{S}_\rho = \langle X\ldots X, Z_1Z_2, \ldots, Z_{n-1}Z_n\rangle \quad \text{and} \quad \mathcal{S}_\sigma = \langle -X\ldots X, Z_1Z_2, \ldots, Z_{n-1}Z_n\rangle. \tag{2.20}$$

Notice that they are almost identical with a different sign in front of the global X stabilizer, implying orthogonality in the logical basis. In the hypothetical, measurement outcomes of ρ are projected onto σ. Let us consider the two types of measurement, pairwise ZZ and single-site X. Performing a pairwise ZZ measurement is a stabilizer measurement, apparent from the generating sets given in Equation (2.20). A measurement thereof will (at least initially) produce deterministic measurement outcomes with $m = +1$. This is different for the X measurements. Initially, an X measurement will not have an outcome we can infer beforehand. This, however, is the case for both of them, up to a certain degree, since the pairwise Z stabilizers anticommute with the single-site X measurement. This also provides

an explanation on why this is not always the case. If X measurements are sufficiently frequent, there will be a point where there are no ZZ generators left, as all of them were replaced by X by means of measuring. Consequently, subsequent measurements of X are deterministic, which will detect the sign difference of the global X stabilizer. This detection would correspond to the point where the initial cluster dies in the colored cluster model.

It turns out that this is the only case for the circuit-level linear cross entropy to be 0, i.e. $\chi_C = 0$. One can argue from the structure of the stabilizer group, or just the generating set, that any measurements producing a random outcome will produce a random outcome no matter what the previous results were.[3] For an outcome to be random, the measurement operator may not be contained in the stabilizer of the state. Conversely, for an outcome to be deterministic, the operator is a stabilizer. After measuring, the operator is guaranteed to be in the stabilizer group of the state and subsequent measurements of this operator produce the same result with unit probability. With the structure of the group changed, we also have new anticommuting measurement operators, and so on. This boils down to the conclusion that once the circuit C is fixed, the *type* of measurement outcome is fixed. One could tell, without knowing the actual outcomes to measurements, which of them were random and which were deterministic. As such, a given circuit, which detects the sign difference between σ and ρ, does so in all of the runs, as deterministic measurement outcomes stay so regardless of the previous history. Consequently, the circuit-level linear cross entropy is 0 regardless of measurement record $\mathbf{m}$, if C contains a deterministic measurement in X, where the outcome is not given by a previous measurement, but by the sign of the initial global X stabilizer.

Figure 2.1 shows a minimal example of vanishing linear cross entropy on the circuit level with initial states chosen as orthogonal Bell states.

2.2 A FIRST PTIM SIMULATION

In this section we will examine the linear cross entropy in the context of our model system, the projective transverse-field Ising model. In particular, we will perform stabilizer simulations, as we have now worked out how the LXE can be computed in the context of Clifford circuits. As we also know an efficient method of computing the (experimentally inaccessible) entanglement entropy, we will also compare the LXE with the entropy of entanglement.

2.2.1 Methods

Before we discuss the linear cross entropy in detail, we introduce the setup for our simulations. If not explicitly stated otherwise, the simulations were performed with a stabilizer simulator written in C++ using the algorithm detailed in Chapter 4 and Section 4.1.1 [70]. Previous numerical investigations of the projective transverse-field Ising model have that the sample

[3] They are "Markovian" in that sense.

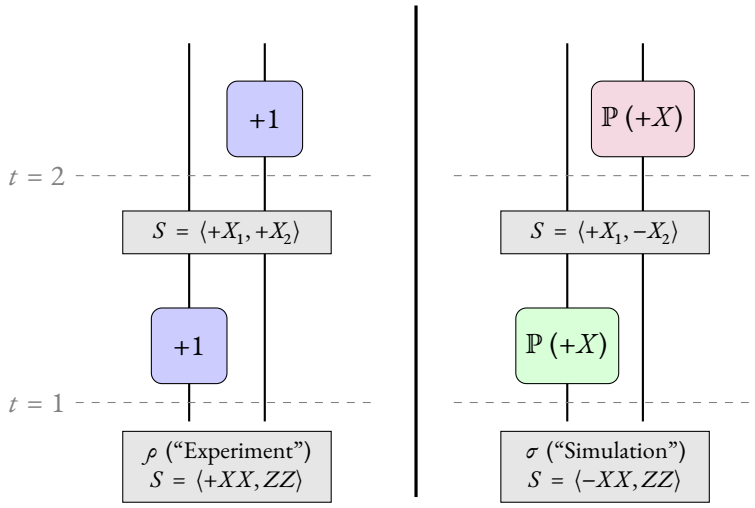


Figure 2.1.: Minimal example of a PTIM setup showcasing the mechanism behind vanishing LXE with $L = 2$, $T = 2$, and $p \neq 0$. The experimental circuit is shown on the left side. In this particular circuit the initial Bell cluster dies in the first measurement layer by measuring X, and thus removing the singular ZZ stabilizers. The +1 in the box refers to the actual outcome of the result. The circuit on the right corresponds to the post-processing algorithm with orthogonal initial state in the logical basis. Here we project onto the measurement results given by the measurement record, represented by $\mathbb{P}(+X)$. The second projection is unsuccessful, since we had a sign difference initially.

average of a system is [71]

$$\left\langle\left\langle f_D \right\rangle\right\rangle = \sum_{\mathcal{T} \in \{\mathcal{T}\}} P(\mathcal{T}) \cdot f(\mathcal{T}; D(S)), \tag{2.21}$$

where $\mathcal{T}$ indicates a trajectory in the system and $\{\mathcal{T}\}$ is the set of all possible trajectories for a given initial state. The double angled braces denote the average over both measurement records within one circuit as well as circuits of a given probability. In the following we discuss how we should go about sampling the circuits in order to get a sensible ensemble average, such that we arrive at something of the form in Equation (2.21).

In Equation (2.21), a trajectory is defined as the circuit C, i.e. the measurement pattern, and the outcomes of said measurements, $\mathbf{m}$, including the initial state. A reasonable place to start is to find a probability we average over. If we split the measurement record into disjoint records of X measurements $\mathbb{m}_X$ and ZZ measurements $\mathbb{m}_Z$, we have that the probability of a trajectory is

$$\begin{aligned} P(\mathcal{T}) &= p^{|\mathbf{m}_X|}(1-p)^{LT-|\mathbf{m}_X|} \cdot p^{(L-1)T-|\mathbf{m}_Z|}(1-p)^{|\mathbf{m}_Z|} \cdot 2^{-N_{\text{rand}}} \\ &= p^{|\mathbf{m}_X|+(L-1)T-|\mathbf{m}_Z|}(1-p)^{LT-|\mathbf{m}_X|+|\mathbf{m}_Z|} \cdot 2^{-N_{\text{rand}}}, \end{aligned} \tag{2.22}$$

where L is the number of qubits (without periodic boundary conditions) and T the number of timesteps. This follows from the fact that we have probability p of X measurements occuring and probability $1-(1-p)=p$ of ZZ no measurements occuring, with the converse for the exponent of $(1-p)$. The factor $2^{-N_{\text{rand}}}$ is from the fact that N_{rand} of the occuring measurements are random in nature and thus contribute a probability of $1/2$ each.

However, recall that the quantity we really want to average (first over the measurement records, then over circuits), is the quantity defined in Equation (2.17). From Theorem 2.4 we know that we can consider each circuit exactly once, since for each circuit it will either give 0 or 1. We should therefore think about how to average the quantity

$$f(\mathbf{m}, C_p, \sigma) = \frac{P(\mathbf{m}|C_p, \sigma)}{\sum_{\mathbf{m}'} \left(P(\mathbf{m}'|C_p, \sigma)\right)^2} \tag{2.23}$$

for a PTIM circuit C_p with probability parameter p, giving the probability of X measurements being performed on each site. We thus have a system average for the PTIM with initial state ρ and probability parameter p,

$$\langle\langle f \rangle\rangle_{p,\rho} = \sum_{C_p} P(C_p) \sum_{\mathbf{m}} P(\mathbf{m}|C_p, \rho) \cdot f(\mathbf{m}, C_p, \sigma). \tag{2.24}$$

In principle, the sums over C_p and $\mathbf{m}$, i.e. the sum over $\mathcal{T}$, include *all possible* circuits C_p and corresponding measurement outcomes $\mathbf{m}$. However, as already discussed, this creates a lot of unnecessary redundancy; with stabilizer circuits, each $\mathbf{m}$ contributes the same factor of $2^{-|N_{\text{rand}}|}$ to the total probability, it doesn't change anything to sample over it. Thus, we can instead opt for the generation of random circuits and subjecting ρ to them once. We then use a project function (see Algorithm 4.9) to verify if a different initial state σ is compatible with the measurement record obtained from applying C on ρ.

In the stabilizer formalism, projections can be done as efficiently as applying measurement gates, since the mechanism behind it is similar. However, we do not get a measurement result, but rather obtain feedback on the success of the projection. For a detailed introduction into the simulation algorithm used in the numerical experiments, consult Chapter 4 and in particular Section 4.2, as well as Algorithm 4.9.

The actual simulation of the PTIM with the computation of the linear cross entropy is then done as follows. First, as outlined in Section 1.3.3, we generate two binary matrices of size $T \times L$, representing the locations of X and ZZ measurements, respectively. We also initialize two additional matrices of the same size to 0, where the measurement outcomes of the respective run is stored. We then simulate a PTIM with a stabilizer simulator (see Ref. [70] and Chapter 4). Afterwards we repeat the circuit for the other initial state, but use the measurement results in the "outcome matrices" as argument for the project function. If at any point during the simulation we encounter a failed projection, the loop over the circuit

stops and continue with the next one. If it successfully completes, we add 1 to the total and repeat the procedure for the next circuit.

Consequently, we technically compute the quantity

$$\langle\langle f \rangle\rangle_{p,\rho} = \sum_{C_p} P(C_p) \cdot f(\mathbf{m}, C_p, \sigma), \tag{2.25}$$

when sampling for the linear cross entropy numerically, where we generate C_p as a random circuit in our simulator.

In the following, we will drop the index p from the labelling of the circuit C_p for the sake of readability. It is nonetheless implied, and should be clear from context that C is a circuit randomly generated from a probability parameter p.

2.2.2 Results

As [4] already explored the linear cross entropy on the projective transverse-field Ising model, we choose the identical setup with the maximally entangled states $\rho = |GHZ+\rangle\langle GHZ+|$ and $\sigma = |GHZ-\rangle\langle GHZ-|$ as initial states with $L = T$ for better comparability with their results in this and the upcoming sections of this chapter.

Figure 2.2 shows the linear cross entropy and the entanglement entropy for different system sizes.

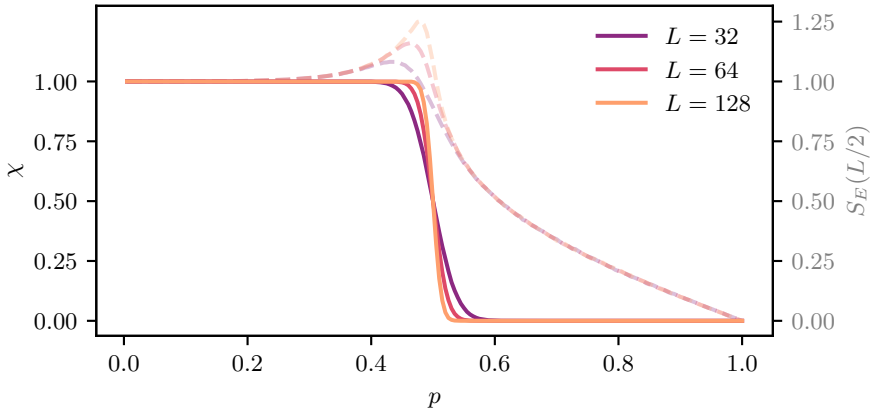


Figure 2.2.: Linear cross entropy and entanglement entropy in the projective transverse-field Ising model and for $L = T$ and initial states $\rho = |GHZ+\rangle\langle GHZ+|$ for the actual system and $\sigma = |GHZ-\rangle\langle GHZ-|$ for the classical replica. Note that we here use open boundary conditions. For each probability parameter p, we sampled over $\sim 10^5$ circuit realizations.

From Figure 2.2 one can see that the linear cross entropy acts as an order parameter for the phase transition, with every line coinciding at the critical point. One can also see that

the linear cross entropy converges faster towards the critical point than the entanglement entropy. The entanglement entropy is still somewhat smooth at the critical point, while the linear cross entropy is already smoothed out. This is due to the fact that the LXE probes the survival of an initial entanglement cluster, while the entanglement entropy quantifies independent entanglement pairs at the very end of the circuit. As such, the linear cross entropy in this form is equivalent to the entanglement entropy of an ancilla qubit entangled initially to the rest of the system. We will make use of this fact later to test the reliability of the LXE in the presence of noise.

A more subtle fact we do not want to fail to mention is that the entanglement entropy is off-center in the case of open boundary conditions. In case of periodic boundary conditions, we have the transition in the entanglement entropy centered at the critical point. We highlight both the convergence behavior and the shift of the critical point in Figure 2.3. Note that we also use larger system sizes compared to Figure 2.2.

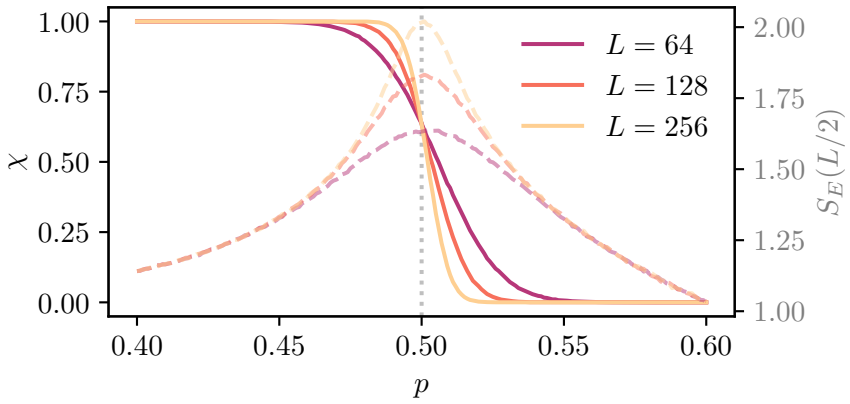


Figure 2.3.: Linear cross entropy and entanglement entropy zoomed in at the critical point for periodic boundary conditions. The dotted vertical line corresponds to the theoretical critical point of the PTIM in the steady state. Each datapoint corresponds to $\sim 10^5$ circuit realizations per probability parameter.

What Figure 2.3 also remarkably shows is that with periodic boundary conditions $\chi(p = 1/2) > 1/2$, whereas with open boundary conditions it was $\chi(p = 1/2) = 1/2$. The latter implies that half of the initial clusters do not survive at the critical point, while the other half does. With periodic boundary conditions, we add another source for stabilizer measurements, thus leading to a more stable cluster towards the critical point, which is still $p_c = 1/2$.

2.3 MARINALIZING OVER PROBABILITIES

In [4] they argue that by tracking only the outcomes of X measurements, one obtains the identical results for the linear cross entropy compared to tracking all results. In this section, we will prove this statement with arguments from classical probability theory, using that the probability $P(\mathbf{m} \mid C, \bullet)$ is a joint probability distribution of two discrete random variables $\mathbf{m}_X$ and $\mathbf{m}_Z$.

The group theoretic arguments leading to vanishing linear cross entropy also give rise to another insight into a more subtle property thereof. Since the sign difference in the initial states is in the global X stabilizer, we can be certain of the fact that it must be an X measurement, which failed to project, thus leading to a vanishing linear cross entropy. Furthermore, we argued that the type of a measurement (random or deterministic) is not altered by previous measurements. (Note that this fact is completely independent of the initial state!) It will therefore prove useful to write the entire measurement record as $\mathbf{m} = \mathbf{m}_X \cap \mathbf{m}_Z$, where $\mathbf{m}_{X/Z}$ are the measurement records of X and ZZ. The individual outcomes in $\mathbf{m}$ are independent of their preceeding outcomes. It follows that $\mathbf{m}_X$ and $\mathbf{m}_Z$ are independent random variables. Starting with orthogonal GHZ states, we could then, theoretically, marginalize over $\mathbf{m}_Z$ and should arrive at the same linear cross entropy.

Let us consider what it implies to have independent random variables. Probability theory tells us that for independent random variables A and B we have $P(A, B) = P(A)P(B)$, which gives us

$$P(\mathbf{m}|C, \rho) = P(\mathbf{m}_X \cap \mathbf{m}_Z|C, \rho) = P(\mathbf{m}_X|C, \rho) \cdot P(\mathbf{m}_Z|C, \rho). \tag{2.26}$$

Note that this argument relies on the fact that in Equation (2.3) we replace each measurement we want to marginalize by an identity of the form

$$\mathbb{1} = \mathbb{P}_+ + \mathbb{P}_-, \tag{2.27}$$

with the projectors onto measurement results $m_{X/Z} = \pm 1$. The sum that emerges when all X (ZZ) projections are replaced by this identity is then the marginalized probability distribution of ZZ (X).

Since the linear cross entropy solely consists of probabilities of this form, we can insert

above separation into Equation (2.5) to yield

$$\begin{aligned}
\chi_C &= \sum_{\mathbf{m}} P(\mathbf{m} \mid C,\rho)\frac{P(\mathbf{m} \mid C,\sigma)}{\sum_{\mathbf{m}'}\left(P(\mathbf{m}' \mid C,\sigma)\right)^2} \\
&= \sum_{\mathbf{m}_X\cap\mathbf{m}_Z} P(\mathbf{m}_X\cap\mathbf{m}_Z|C,\rho)\frac{P(\mathbf{m}_X\cap\mathbf{m}_Z|C,\sigma)}{\sum_{\mathbf{m}'_X\cap\mathbf{m}'_Z}\left(P(\mathbf{m}'_X\cap\mathbf{m}'_Z|C,\sigma)\right)^2} \\
&= \sum_{\mathbf{m}_X\cap\mathbf{m}_Z} P(\mathbf{m}_X|C,\rho)P(\mathbf{m}_Z|C,\rho)\frac{P(\mathbf{m}_X|C,\sigma)P(\mathbf{m}_Z|C,\sigma)}{\sum_{\mathbf{m}'_X\cap\mathbf{m}'_Z}\left(P(\mathbf{m}'_X|C,\sigma)P(\mathbf{m}'_Z|C,\sigma)\right)^2} \\
&= \underbrace{\frac{\sum_{\mathbf{m}_Z} P(\mathbf{m}_Z|C,\rho)P(\mathbf{m}_Z|C,\sigma)}{\sum_{\mathbf{m}'_Z}\left(P(\mathbf{m}'_Z|C,\sigma)\right)^2}}_{\text{Subset ZZ}}\ \underbrace{\frac{\sum_{\mathbf{m}_X} P(\mathbf{m}_X|C,\rho)P(\mathbf{m}_X|C,\sigma)}{\sum_{\mathbf{m}'_X}\left(P(\mathbf{m}'_X|C,\sigma)\right)^2}}_{\text{Subset X}}
\end{aligned} \tag{2.28}$$

with "Subset ZZ" and "Subset X" being the marginalized versions of the linear cross entropy, where the other type of measurement is traced out.

As of yet, we have not proven the statement that tracking the outcomes of X measurements yields the same linear cross entropy as tracking all outcomes of all measurements. We have only shown that when tracking only a subset of measurements, we can multiply it by the complement to obtain the full linear cross entropy.

However, if we paradigmatically start the circuit in orthogonal GHZ states, where the only difference between the two initial states (the one of the quantum and the one of the classical simulator) is the sign of the global X stabilizer, we can deduce that there will never be a difference in the "Subset ZZ" linear cross entropy, since replacing the projectors onto X measurement results by an identity will never project onto 0. Therefore, it should be identically 1.

2.3.1 Methods and results

To implement a marginalized version of the linear cross entropy, we need to replace each projection we would perform in the replicated circuit by an identity operation. Naively, we could then just remove all complementary measurement gates in the numerical implementation. For a more accurate description, or simulation, of what we actually want to achieve, we replace each projection we would do when tracking all outcomes with a measurement gate, where we choose to be agnostic to the outcome. This is done to keep the underlying group structure intact. Note that contrary to the projection operation we implemented, a measurement can always be performed and always has an outcome.

For the results of a simulation implemented this way, consider the first row of subplots in Figure 2.7. In this figure, we show the linear cross entropy as a function of the probability parameter p, where the different columns correspond to different marginalizations. We additionally plot the entanglement entropy of an ancilla qubit entangled to the initial Bell cluster of the system, as we already noted that this is a physically equivalent quantity in this case. In column "Track X (ZZ)" we only store the measurement outcomes of X (ZZ) measurements and perform a measurement of ZZ (X) instead of a projection in the "classical version" of the circuit.

Note that the label A in the subplot of columns X and all correspond to the mechanism, which leads to a vanishing cross entropy we discussed earlier and which is depicted schematically in Figure 2.1. The reason we introduce this becomes apparent in the next section, where we examine the behavior of the linear cross entropy in the presence of symmetric (projective) noise. For the noiseless case, the previously introduced mechanism, which we will refer to as mechanism A, is the only way where the linear cross entropy vanishes.

The first row in Figure 2.7 clearly shows that only tracking the outcomes of ZZ measurements yields a (marginalized) linear cross entropy that is identically 1. Conversely, the tracking of only the outcomes of X measurements yields the same curve as the one where we track everything. This is an indicator that the probability theoretic argument that $\mathbf{m}_X$ and $\mathbf{m}_Z$ are independent random variables holds water. By our choice of initial states, there is no mechanism present in the circuit, where the marginalized cross entropy of ZZ goes to 0. We will take special note of this fact when we introduce noise to the circuit.

2.4 PTIM WITH FAULTY GATES

So far, the LXE appears to be a promising candidate for the order parameter of the phase transition in the projective transverse-field Ising model. Nevertheless, it is a well known fact that the world is not perfect. It is utopian to imagine a quantum simulator going through a circuit without any errors. Hence it seems a worthwile endeavor to investigate the robustness of the linear cross entropy when the "experimentally realized" circuit with initial state ρ is subjected to noise.

In this section and the remainder of this chapter we investigate the impact of a type of error model on the linear cross entropy and verify its robustness to noise. We build on and critique the work of Ref. [4] insofar as we implement their error model in our simulations, expand it to include also ZZ noise, as well as the marginalized linear cross entropies, and contextualize the results.

2.4.1 Error model

As error model, we implement the model from [4], where they considered symmetric noise in X. The protocol to introduce errors is as follows. Between each timestep, apply a quantum channel of the form

$$\varepsilon(\rho) = \frac{1}{2}\rho + \frac{1}{2}X\rho X \tag{2.29}$$

on each qubit with an error rate q.[4] That is, after each timestep, before applying the measurement gates of the next one, we construct an additional measurement layer, where measurements of certain types are performed with probability q. This scheme is generic insofar as we can also imagine ZZ errors happening in this way, as well as errors on both observables.

This type of error can be interpreted as additional measurements we failed to keep track of in general, and not by choice. Another way of interpreting it is that they constitute projective errors, akin to the one shown in a minimal example in Figure 1.1. Our expectation from the interpretation of the survival rate of the initial cluster would be that for X noise, we have the phase transition at a smaller probability, since the initial cluster dies earlier due to the presence of more frequent X measurements. In the converse we expect the cluster to survive longer, since we artificially introduce stabilizer measurements at a rate q.

Before we put this expectation to the test, we need to consider how we go about sampling this quantity. We implement this noise in a way closest to what one could consider noise in a real experiment starting with our usual scheme of designing a random circuit C and measuring accordingly. However, this time we measure additionally on each qubit after each timestep with an error rate q. These measurements we don't keep track of and seed randomly in the simulation of ρ.

By introducing errors to the circuit, we subject it to quite impactful alterations. Previously we could consider the reduced measurement pattern—which we had access to—and apply it to both initial states, with mechanism A being the only source of vanishing terms in the sampling. Now we need to consider the designed, albeit still randomly generated, circuit C and the faulty pattern $\tilde{C}$, where $\tilde{C}$ is parametrized by both p and q. Crucially, the new faulty circuit $\tilde{C}$ constitutes a superset of C, since each measurement of C is performed regardless. That is, we have $C \subseteq \tilde{C}$.

Note that this leaves the fraction in Equations (2.5) and (2.17), that is,

$$f(\mathbf{m}, C, \sigma) = \frac{P(\mathbf{m}|C,\sigma)}{\sum_{\mathbf{m}'} \left(P(\mathbf{m}'|C,\sigma)\right)^2} \tag{2.30}$$

invariant; we are still trying to find the compatibility between classical simulation (C, σ) and "quantum" experiment $(\tilde{C}, \rho)$. However, we will obtain a different expression for the

[4] We refer to this quantum channel as "error" and "noise" interchangeably.

sample average, Equation (2.25). As we are trying to realistically model errors, we should be unaware of the location they happen in, but assume that they happened. As such, $\tilde{C}$ is a random circuit with probability parameters p and q for measurements we do and do not have control over, respectively. Equation (2.25) then becomes a sum over $\tilde{C}_{p,q}$,

$$\left\langle\left\langle f \right\rangle\right\rangle_{p,q,\rho} = \sum_{\tilde{C}_{p,q}} P(\tilde{C}_{p,q}) \cdot f(\mathbf{m}, C_p, \sigma). \tag{2.31}$$

With the sampling scheme defined in a more precise way, we can refine our predictions on the results in the simulation. Although the f in Equation (2.31) and Equation (2.25) are identical, the measurement record we take as input is not generated from the same circuit. Consequently, we can now identify other causes of the function going to 0.

Previously we argued that the probability of a measurement outcome is not influenced by preceeding measurements, that is, they are independent random variables. While this is still the case, the argument only holds on a circuit-level. The only way where an error does not alter the probability of the succeeding measurements is, if it commutes with the measurement operators on the same site that directly preceed or succeed the error. For instance, if the circuit dictates a measurement of X, which is then followed by the noise channel, the state is not altered and nothing is affected.

Thus, we argue that there is an additional mechanism for vanishing circuit-level linear cross entropy, where an error is not bypassed by the mechanism described above, but is entrapped by the competing measurement.

Take, for instance, the minimal example of two qubits with X-Errors. A valid measurement pattern would be (Z_1Z_2, Z_1Z_2). Starting with a Bell state would yield the outcomes $(+1, +1)$ deterministically. If we now squeeze an error inbetween the two measurements, we have halved the probability of getting $+1$ at the second timestep. Thus, for half of the runs we would get an unsuccessful projection.

The mechanism of the circuit-level linear cross entropy going to 0 due to errors which fail to not get noticed will henceforth be denoted with B1 and B2 for X-errors and ZZ-errors respectively. Possible examples of them occuring in a PTIM experiment with the corresponding simulation are depicted schematically in Figures 2.4 and 2.5.

It is not hard to convince oneself that the linear cross entropy sampled with faulty circuits is less than or equal to the original cross entropy, since the original source of it going to 0 is still present nonetheless. We can do a simple estimation on the probability of one of the above introduced mechanisms to bound the error-influenced linear cross entropy from above. These considerations also serve to better understand them.

Let us consider mechanism B1, where X errors occur with probability q. The measurement record $\mathbf{m}$ obtained from $\tilde{C}$ applied on ρ is incompatible with σ if there is a projection onto the zero-vector. Thus, for a non-zero LXE we want this *not* to happen. One way for this to happen is if mechanism B1 takes effect. To derive the probability of B1 occuring, consider the following train of thought.

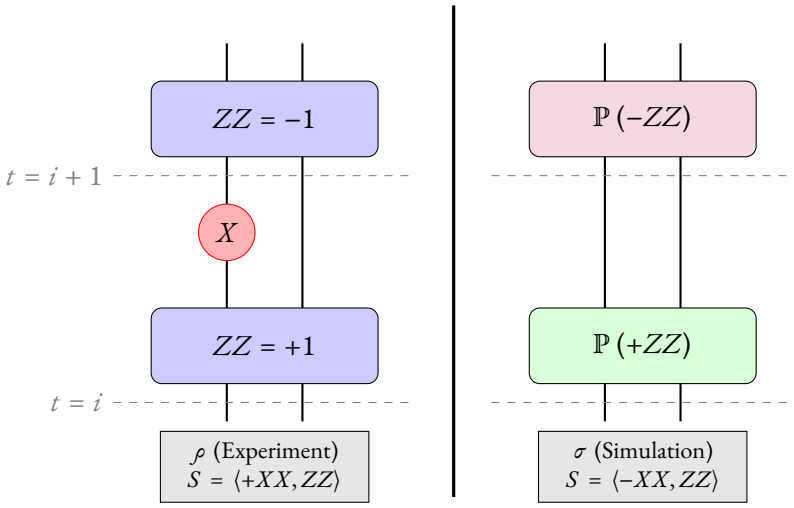


Figure 2.4.: Excerpt of a possible PTIM circuit with non-zero probability of X-errors occuring showcasing mechanism B1. At timestep $t = i$, we perform a ZZ measurement on two neighboring qubits with result +1, and an X-error occurs on the first qubit after the measurement. Then in the next timestep, we measure ZZ once more. As a consequence of the error, this ZZ measurement is not deterministically +1, but randomly ±1 with probability $\frac{1}{2}$. When trying to project in the classical simulation, this discrepancy gets noticed, since we do not have access to the precise nature of the errors. Upon failed projection we have $\chi = 0$.

1. Two successive ZZ measurements must occur in C on the same edge. This event has probability $(1-p)\cdot(1-p) = (1-p)^2$ by the construction rules of the PTIM circuit.

2. No X measurements on the site where the error occurs. This event also has probability $(1-p)^2$, as p is the probability of X measurements, again by the rules on how we construct a PTIM circuit.

3. An error must occur on one of the sites, which gets detected. The error rate is q, with an effective probability of $q/2$, since the "correct" measurement result is still a valid outcome in the circuit.

4. The mechanism is symmetric in the two sites encompassed in the edge the ZZ measurements occur, which multiplies the above points by 2.

Thus, the probability of B1 is

$$P(\mathrm{B1}; p, q) = \frac{2}{2} q \left(1-p\right)^2 \left(1-p\right)^2 = q\left(1-p\right)^4 . \tag{2.32}$$

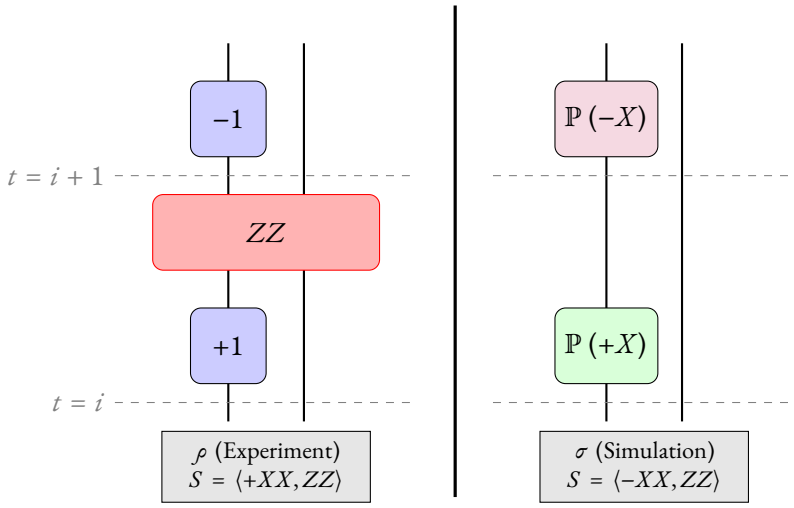


Figure 2.5.: Excerpt of a possible PTIM circuit with non-zero probability of ZZ-errors occuring showcasing mechanism B2. At timestep $t = i$, we perform an X measurement on the left qubit with result +1, and a ZZ error occurs on the shown pair after the measurement. Then in the next timestep, we measure X on the left qubit once more. As a consequence of the error, this X measurement does not yield +1 deterministically, but ±1 with probability $\frac{1}{2}$ for either result. When trying to project in the classical simulation, this discrepancy gets noticed, since we do not have access to the precise nature of the errors. Upon failed projection we have $\chi = 0$.

Since this should not happen anywhere in the space-time lattice, we have

$$\chi \leq \left(1 - q\left(1 - p\right)^4\right)^{(L-1)(T-1)} \tag{2.33}$$

as an estimation for the linear cross entropy for X-errors. The LXE is thus exponentially suppressed for $p < 1$ in the thermodynamic limit of $L \to \infty$. We should also not confuse the facts here; this estimate does not give the actual behavior of the linear cross entropy. It is only a heuristically derived upper bound for the rate of occurence of mechanism B1. In principle, this probability would need to be multiplied by the probability of the other mechanisms to get a tighter upper bound. However, it stands to reason that even if this were the only mechanism (and we will come across an example of exactly this later), the linear cross entropy would still go to 0 for large systems, with a singular value of $\chi = 1$ at $p = 1$.[5] Note that the converse is the case for B2 by an analogous line of reasoning, where it is $\chi = 1$ at $p = 0$ and $\chi = 0$ otherwise.

[5] This follows from the sandwich lemma.

2.4.2 Tracking all measurement outcomes

We first discuss the results for tracking every measurement outcome. For a visualization, consider the "Track all" column of Figure 2.7. The annotations in the individual subplots highlight the predominant mechanism leading to a vanishing cross entropy, referring to the situations depicted in Figures 2.4 and 2.5. We set up the system the same way as before with $\rho = |GHZ+\rangle\langle GHZ+|$ and $\sigma = |GHZ-\rangle\langle GHZ-|$, and an additional error rate of $q = 0.01$, consistent with the choice in Ref. [4]. Note that in the cases where X errors are present, we show the plots of systems with fewer qubits. As we can infer from Equation (2.33) the linear cross entropy is exponentially suppressed with larger system size. We therefore obtain $\chi \equiv 0$ for sufficiently large systems, which would be the case for $L = 64$ and $L = 128$, as shown in the other subplots. Shown additionally is the entanglement entropy of an ancilla qubit, S_{anc}, which was the initial interpretation of the linear cross entropy in case of no noise. One can clearly see that this interpretation is no longer a valid one.

To highlight the shift of the critical point in case of noise, consider Figure 2.6, where the entanglement entropy of an ancilla qubit entangled to the initial cluster is shown with the corresponding linear cross entropy. For the simulation we probed a region of p close to the critical point with an error rate of $q = 0.1$, higher than the one in Figure 2.7. We here chose smaller systems as well. Notice that the critical point moves as expected, where the linear cross entropy leaves no possibility for inference thereof. Its behavior is seemingly decoupled from the actual dynamics of the entanglement cluster, and is dominated by the other mechanisms leading to a vanishing cross entropy. Even for relatively small systems of $L = 16$, the linear cross entropy is not robust to the influence of noise.

Our results thus show that the linear cross entropy in the form defined in Definition 2.2 is not a sensible choice for an order parameter of the phase transition, as experimental realizations of the projective transverse-field Ising model will inevitably include noise, exponentially suppressing the linear cross entropy. Its utility with regards to the phase transition in the PTIM is therefore rather limited.

As an aside, we want to bring attention to the fact that our results differ from the results obtained by [4]. Their results for the linear cross entropy in a noisy circuit show little deviation from the noiseless behavior, seemingly only scaled down by some factor. However, by the provided upper bound, we know the LXE to be exponentially suppressed. This discrepancy between our results and theirs could be caused by a multitude of different issues. First, the source of the discrepancy could be statistical in nature, due to different sample sizes. Additionally, they could also process their data with alternative methods. However, from the material available to us, it remains unclear what causes this difference in results.

2.4.3 Marginalization

Notice that if, instead of applying projection operations, we applied measurement gates in the right hand side circuits depicted in Figures 2.4 and 2.5, we would – obviously – not

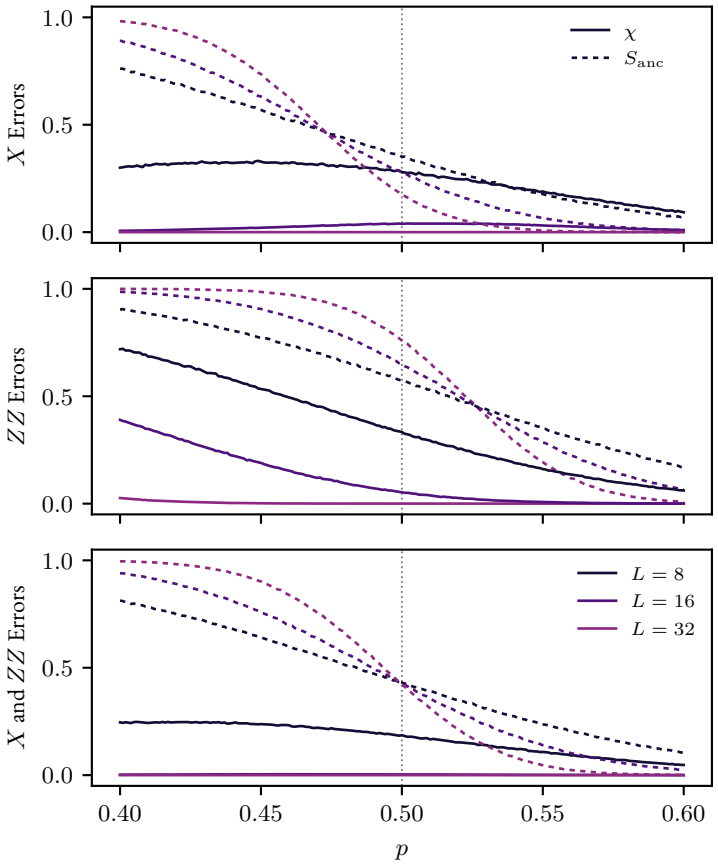


Figure 2.6.: Ancilla entanglement entropy and linear cross entropy for an error rate of $q = 0.1$ to highlight the behavior of S_{anc}. The system sizes are chosen smaller compared to Figure 2.7 since χ would be 0 for larger systems. Note that this is shown for the region around the critical point $p = 0.5$ in the ideal case. This was done to make the shift of S_{anc} in p more noticeable without having the cross entropy be 0 for small system sizes. Grey vertical dots indicate the critical point in the ideal case of no errors at $p = 0.5$. Each datapoint corresponds to the sampling of $\sim 10^5$ circuit realizations.

project onto the 0-vector, and thus the circuit would continue. This is exactly what happens when one marginalizes out the measurement results of ZZ or X measurements, as discussed in Section 2.3. It thus stands to reason that we should entertain this probability theoretic trick for the noisy circuits.

Within $\tilde{C}$ and C the same argument as in Section 2.3 holds. The outcomes we track are still independent random variables, now only with a different circuit that produces them, i.e.

$$P(\mathbf{m}_X \cap \mathbf{m}_Z|\tilde{C},\rho) = P(\mathbf{m}_X|\tilde{C},\rho)\cdot P(\mathbf{m}_Z|\tilde{C},\rho). \tag{2.34}$$

Therefore, despite the fact that ρ and σ are being subjected to physically different circuits, we can still separate the respective probabilities as we did before. We just need to be careful with the interpretation. That is, we can follow the derivation in Equation (2.28), where we replace C in the probabilities conditioned on initial state ρ with $\tilde{C}$,

$$\begin{aligned}
\chi_{\tilde{C},C} &= \sum_{\mathbf{m}} P(\mathbf{m}\mid\tilde{C},\rho)\frac{P(\mathbf{m}\mid C,\sigma)}{\sum_{\mathbf{m}'}\left(P(\mathbf{m}'\mid C,\sigma)\right)^2}\\
&= \sum_{\mathbf{m}_X\cap\mathbf{m}_Z} P(\mathbf{m}_X\cap\mathbf{m}_Z|\tilde{C},\rho)\frac{P(\mathbf{m}_X\cap\mathbf{m}_Z|C,\sigma)}{\sum_{\mathbf{m}'_X\cap\mathbf{m}'_Z}\left(P(\mathbf{m}'_X\cap\mathbf{m}'_Z|C,\sigma)\right)^2}\\
&= \sum_{\mathbf{m}_X\cap\mathbf{m}_Z} P(\mathbf{m}_X|\tilde{C},\rho)P(\mathbf{m}_Z|\tilde{C},\rho)\frac{P(\mathbf{m}_X|C,\sigma)P(\mathbf{m}_Z|C,\sigma)}{\sum_{\mathbf{m}'_X\cap\mathbf{m}'_Z}\left(P(\mathbf{m}'_X|C,\sigma)P(\mathbf{m}'_Z|C,\sigma)\right)^2}\\
&= \underbrace{\frac{\sum_{\mathbf{m}_Z}P(\mathbf{m}_Z|\tilde{C},\rho)P(\mathbf{m}_Z|C,\sigma)}{\sum_{\mathbf{m}'_Z}\left(P(\mathbf{m}'_Z|C,\sigma)\right)^2}}_{\text{Subset ZZ}}\underbrace{\frac{\sum_{\mathbf{m}_X}P(\mathbf{m}_X|\tilde{C},\rho)P(\mathbf{m}_X|C,\sigma)}{\sum_{\mathbf{m}'_X}\left(P(\mathbf{m}'_X|C,\sigma)\right)^2}}_{\text{Subset X}}.
\end{aligned} \tag{2.35}$$

As discussed previously, by tracing out the other measurement outcomes, we effectively replace them by an identity in the calculation of the probability. It therefore turns out that tracking only the outcomes $\mathcal{O}$-measurements prevents us from seeing $\mathcal{O}$-errors. Another way one can convince oneself of this fact is the following. If we do not care about the outcomes of the observable, where no errors happen, then every error gets necessarily bypassed as described above. We can no longer distinguish if a measurement turned from random to deterministic due to a faulty random measurement. The outcome is still a valid one with respect to C (and possibly σ, for that matter). On the other hand, it will be impossible to tell if the initial entanglement cluster died because of a faulty measurement or a native one.

In particular will marginalizing over ZZ, i.e. tracking X, with X-errors yield the identical linear cross entropy as in the noiseless case. With the results of Section 2.3 we also know that this is then equivalent to tracking every outcome in the noiseless circuit. Furthermore will tracking ZZ while having ZZ errors still be identically 1. Note that this implies that the different mechanisms for the different error types get absorbed depending on which observable we choose to keep track of.

Results

The results of the numerical analysis are shown in Figure 2.7. The linear cross entropy and the entanglement entropy of an ancilla qubit are shown, where every possible combination of error type and marginalization is represented in a separate subplot. Although we previously made reference to the figure, we will clarify how the tableau presentation is to be read, as we will employ it again for our numerical results in Chapter 3.

Along the rows of the tableau of subplots we simulate the PTIM with noise represented by different observables. In the first row, there are no errors, $q = 0$. In the second row, we introduce symmetric noise in X as defined in Section 2.4.1 and Equation (2.29). As error rate we choose $q = 0.01$ in agreement with Ref. [4]. In the third row we proceed analogously with ZZ errors, also at a rate of $q = 0.01$. In the last row, we combine the two, where first the noise channel of X and then the one of ZZ is applied.

Along the columns of Figure 2.7, we use a different subset of measurement outcomes. In the first column, titled "Track X", we track the outcomes of X measurements, and replace projections onto ZZ results with measurement operations. The second column, titled "Track ZZ" is the converse of the first, with ZZ outcomes tracked and X outcomes marginalized. The last column is where we track everything.

The solid line denotes the linear cross entropy, while the dashed line is the ancilla entanglement entropy. Note that each simulation was done twice: once for an isolated system, which we then compared with σ in order to compute the linear cross entropy, and another one entangled to an ancilla qubit. We want to emphasize that the different combinations of errors and marginalizations are qualitatively different in that some system sizes need not be shown for one or the other. For instance, in the "X Errors, Track all" case, the linear cross entropy for $L = 64$ would be 0 almost everywhere, with miniscule deviations. We thus omit some system sizes in some subplots, also in an effort to retain legibility.

Within each subplot we annotated each curve to qualitatively indicate the predominant mechanism that sends χ to 0, refering to the previously defined denotions of said mechanism. For the "X Errors – Track X", "No Errors – Track X", and "No Errors – Track all" situations, the only mechanism is the one where the initial cluster dies by means of an X measurement. As such, mechanism A is the predominant one. For the "X Errors – Track all", we have A, as well as B1, the latter of which is predominant in the regime $p < p_c$, whereas the former is predominant for $p > p_c$. Notice that this can be seen in the combination of the subplots to

the left of it. In the "No Errors – Track ZZ" case, the linear cross entropy is identically 1, which is no longer the case for X errors. Here, we have B1, which is the only mechanism by which the ZZ linear cross entropy goes to 0.

Since we also know that the "Track all" linear cross entropy is the product of the marginalized ones, we can read each line as the product of the first two plots equaling the third one. For rows 1 and 3, this is rather trivial. For the others, it offers an explanation as to why it was necessary to lower the system size in the last column.[6] This shows that the simulations agree with the prediction.

These remarkable results notwithstanding, we would still be hard-pressed to find any evidence of the *real*, i.e. *measureable*, phase transition of the projective transverse-field Ising model in noisy systems. This is highlighted by the fact that the "true" behavior of the system is simulated as well in the form of the ancilla entanglement entropy. For this we especially focus on the last row of the tableau, which would be the noisiest system, where the behavior of the linear cross entropy is seemingly decoupled from the dynamics of the cluster. In particular, due to the exponential suppression in the thermodynamic limit the mechanisms that are *not* A dominate.

However, this exponential suppression can be synthesized into an advantage. Notice that as a consequence of this suppression, the "Track ZZ" column is highly sensitive to X noise. Where it is identically 1 in the ideal case, even small error rates in small systems lead to the deviation from the ideal behavior. We can therefore use this fact, as well as Equation (2.33), to estimate how noisy our system is. As such, one *could* still draw some utility from the linear cross entropy, in that the "Track ZZ" column could function as an indicator of noise in the system, albeit only noise in X, i.e. bitflips.

[6] The last row offers the better visualization of this phenomenon in the nontrivial case compared to the second.

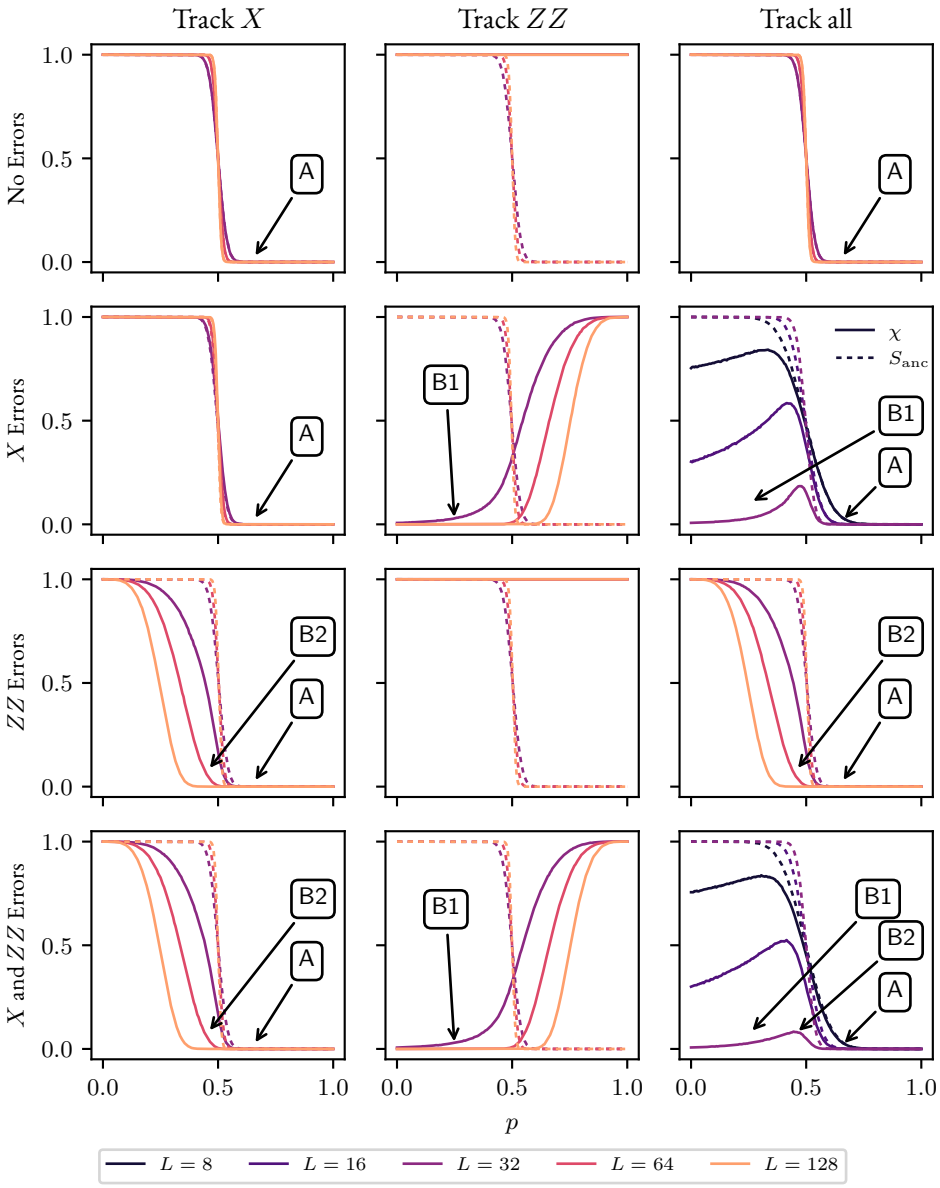


Figure 2.7.: Linear cross entropy (solid) and entanglement entropy of an ancilla qubit entangled to the initial cluster (dashed) for $p \in [0, 1]$ for all combinations of errors and marginalizations and different system sizes. Note that the ancilla entropy is unaffected by the marginalizations. Annotations within the subfigures reference the predominant mechanisms of the linear cross entropy going to 0. A corresponds to Figure 2.1, B1 and B2 correspond to Figures 2.4 and 2.5, respectively.

2.5 SUMMARY

In this chapter we examined the linear cross entropy in the projective transverse-field Ising model. The linear cross entropy was introduced by Li et al. in [2], and first investigated for the projective transverse-field Ising model by Tikhanovskaya et al. in [4]. In search of a promising candidate for an order parameter of the phase transition that is measureable in an experimental setting, we picked up where these previous works left off. Our investigations showed that while it is a fitting quantity for noiseless systems, the linear cross entropy is not at all robust to noise in the system, as it gets exponentially suppressed with larger system size. We additionally showed that we can marginalize over measurement outcomes of one operator and still find some utility in the linear cross entropy. In an experimental setting, the marginalized linear cross entropy could be employed in attempts to minimize the influence of noise, as it is highly sensitive to noise.

3

UPPER BOUND

"To Infinity And Beyond"

Buzz Lightyear

In this chapter we explore the approach of Garratt and Altman presented in [3][1]. They introduce a method to probe the critical point of the phase transition of a hybrid circuit. In particular, they make use of Klein's inequality, bounding the entanglement entropy from above. In the present chapter we introduce the idea behind this approach, translate it to our system—the projective transverse-field Ising model (PTIM)—, and evaluate the quality of the resulting estimate.

The chapter is structured as follows. In Section 3.1 we state the principles behind the idea and derive the upper bound. Moreover, we state and prove that for Clifford circuits, i.e. in the stabilizer formalism, there is only a limited class of cases, where the upper bound is non-trivial. In Section 3.3 we introduce different numerical post-processing algorithms, trying to combat infinities appearing, when introducing an error model akin to the one in Section 2.4.1. The results of which are shown and discussed in **??**. Finally, we provide and discuss regularizations of divergences, weighing out utility and computational efficiency.

[1] While this reference points to a preprint on arXiv, we remark that the cited work has since been published in PRX Quantum with no significant alterations (see Ref. [72])

3.1 THE IDEA & STABILIZERS

In this section we will introduce the upper bound on entanglement entropy, provide an information-theoretic interpretation and derive an expression for the quantity of interest in case of stabilizer states. First however, we will recall the definitions of some important quantities.

Entropy of entanglement

The main quantity of interest in the whole field of entanglement transitions is the entropy of entanglement. It is a measure for how entangled one subsystem of a bipartite state $|\phi\rangle_{AB}$ is with the other. We recall from Section 1.2 that its definition can be stated as follows [27].

Definition 3.1 (Entanglement entropy)**.** Let $|\phi\rangle \in H^{\otimes N}$ be a bipartite pure state with subsystems A and B. The entropy of entanglement of $|\phi\rangle$ then reads

$$S_{\mathrm{E}}\left(|\phi\rangle\right) \equiv -\operatorname{Tr}\left[\rho_B \log \rho_B\right], \tag{3.1}$$

where $\rho_B = \operatorname{Tr}_A[|\phi\rangle\langle\phi|]$ is the reduced density matrix of subsystem B. Conventionally, one uses the logarithm of base 2.

This quantity can be efficiently computed in clifford circuits via the stabilizer formalism. It is important to note that the density matrix of the whole system $\rho = |\phi\rangle\langle\phi|$ describes a pure state. For a more universal measure of entropy, where the entropy of entanglement is a special case, we need to consider the more general *von Neumann entropy*.

Von Neumann entropy

The von Neumann entropy lets us quantify the average information content in a mixture of quantum states. Originally, von Neumann introduced it as an extension of the classical Shannon entropy to quantum systems, as density matrices serve as extension of the classical notion of (discrete) probability distributions [47].[2] Consequently, we can write down a definition of the von Neumann entropy.

Definition 3.2 (Von Neumann entropy)**.** Let ρ be an N-qubit density matrix. The von Neumann entropy of ρ is given by

$$S\left(\rho\right) \equiv \left\langle -\log \rho \right\rangle = -\operatorname{Tr}\left[\rho \log \rho\right]. \tag{3.2}$$

[2] Shannon entropy is a measure of the average information of probabilistic events. Since Information is defined as $I = -\log p_x$ for some event x with probability p_x, we can write down an average, $S = \left\langle -\log p_x \right\rangle_x = -\sum_x p_x \log p_x$, which defines the Shannon entropy [48].

For diagonalizable matrices (as we expect density matrices to be) we can also express Equation (3.2) as a sum over eigenvalues λ_k of ρ, i.e.

$$S(\rho) = -\sum_k \lambda_k \log \lambda_k, \tag{3.3}$$

where $0 \leq \lambda_k \leq 1$. This can be shown by diagonalizing ρ and using the fact that the matrix logarithm of a diagonal matrix is just the logarithm of the entries. (Note that we set $0 \cdot \log 0 \equiv 0$.)

From this definition we can already derive some important properties, namely its lower and upper bound. Firstly, $S(\rho)$ is non-negative and 0 iff. ρ is pure. It is easy to convince oneself of that fact. For pure states we have $\rho = |\phi\rangle\langle\phi|$, which has only one non-zero eigenvalue, $\lambda = 1$. Consequently, Equation (3.3) reduces to $S(\rho) = -1 \cdot \log 1 = 0$. Next, $S(\rho)$ is maximal for the maximally mixed state. In a d-dimensional Hilbert space, the density matrix of the maximally mixed state is $\rho = \mathbb{1}/d$. Hence, Equation (3.3) reduces to

$$-\sum_k \frac{1}{d} \log \frac{1}{d} = -\log \frac{1}{d} = \log d.$$

Quantum relative entropy

Lastly, we want to introduce the quantum relative entropy. The same way we motivated the von Neumann entropy, we want to define a quantum mechanical analogue to the classical relative entropy. This quantity will play a role of major importance in this chapter and is defined as follows.

Definition 3.3 (Quantum relative and cross entropy)**.** Let ρ and σ be density matrices with identical dimension. The relative entropy of ρ to σ is

$$S(\rho \,||\, \sigma) \equiv \mathrm{Tr}[\rho \log \rho] - \mathrm{Tr}[\rho \log \sigma] = -S(\rho) - \mathrm{Tr}[\rho \log \sigma], \tag{3.4}$$

with the von Neumann entropy $S(\rho)$. The last term in Equation (3.4), that is,

$$S(\rho \,||\, \sigma) + S(\rho) \equiv S_C(\rho \,||\, \sigma) = -\mathrm{Tr}[\rho \log \sigma] \tag{3.5}$$

is known as *cross entropy*.

Relative entropy, classical or quantum, quantifies excess surprisal when one assumes σ as a state[3], when the actual state is ρ. In that sense it tells us how different two quantum states are. Interpreting it in this way also provides us with a neat heuristic approach to possible values: the only way we do not lose information is if we assume correctly, and in any other

[3] or probability distribution in the classical case

case we lose some. The most extreme form of this is if $\text{supp}(\rho) \cap \ker(\sigma) = 0$, where the relative entropy diverges. In most general terms the relative entropy fulfills

$$S(\rho \,||\, \sigma) < \infty \iff \text{supp}(\rho) \subseteq \text{supp}(\sigma), \tag{3.6}$$

where $\text{supp}(\bullet) \equiv \ker(\bullet)^{\perp}$ is the support of a linear operator, which is defined as the orthogonal complement to the kernel. Alternatively, for diagonalizable matrices, it is the subspace spanned by eigenvectors with non-zero eigenvalues [73, 74]. This condition can be interpreted in a physical way. Let

$$\rho = \sum_i \lambda_i \, |\lambda_i\rangle\langle\lambda_i| \quad \text{and} \quad \sigma = \sum_i \mu_i \, |\mu_i\rangle\langle\mu_i|$$

be orthonormal eigendecompositions of ρ and σ. Then $S(\rho \,||\, \sigma)$ diverges iff. there are states that ρ features in its mixture that σ does not. That is, ρ has to be more mixed than, or at least as mixed as, σ. If it is as mixed as σ, it has to be a mixture of the same states. This means that the information lost under the assumption of a purer state is infinite. Since we know the von Neumann entropy to be bounded, this divergence occurs solely in the $-\text{Tr}\big[\rho \log \sigma\big]$ term.

One property hinted at earlier is the non-negativity of the relative entropy. This property goes by many different names, depending on context. For instance, in classical statistical mechanics, this result is known as the *Gibbs inequality*. Its quantum mechanical analog is also known as *Klein's inequality*.

3.1.1 Klein's inequality

In this section we will state and prove the relation central to this chapter. Its statement reads as follows.

Theorem 3.4 (Klein's inequality)**.** The quantum relative entropy is non-negative, $S(\rho \,||\, \sigma) \geq 0$, with equality iff. $\rho = \sigma$.

While the statement in itself is written out rather simply, proving Theorem 3.4 requires us to introduce two auxiliary relations, namely *Jensen's inequality* and the *log sum inequality*.

Theorem 3.5 (Jensen's inequality)**.** Let f be a convex function, that is, $f(\lambda x_1 + (1-\lambda)\, x_2) \leq \lambda f(x_1) + (1-\lambda)\, f(x_2)$ with $0 \leq \lambda \leq 1$, and X a discrete random variable. Then

$$\mathbb{E}\big[f(X)\big] \geq f\left(\mathbb{E}\left[X\right]\right). \tag{3.7}$$

Proof of Jensen's inequality. We will prove Jensen's inequality by induction over the mass of X. Associated with each event $x_i \in X$ is its probability p_i, such that our induction hypothesis is

$$\sum_{i=1}^{n} p_i f(x_i) \geq f\left(\sum_{i=1}^{n} p_i x_i\right). \tag{3.8}$$

As base case we have $n = 2$;

$$\begin{aligned}
\mathbb{E}\left[f(X)\right] &= p_1 f(x_1) + p_2 f(x_2) \\
&= p_1 f(x_1) + (1 - p_1) f(x_2) && \text{(Kolmogorov)} \\
&\geq f(p_1 x_1 + (1 - p_1) x_2) && (f \text{ convex}) \\
&= f(\mathbb{E}[X]).
\end{aligned}$$

For the induction step, assume Equation (3.8) holds for some $n \in \mathbb{N}_{>1}$, it must then also hold for $n + 1$;

$$\begin{aligned}
\sum_{i=1}^{n+1} p_i f(x_i) &= p_{n+1} f(x_{n+1}) + \sum_{i=1}^{n} p_i f(x_i) \\
&= p_{n+1} f(x_{n+1}) + (1 - p_{n+1}) \sum_{i=1}^{n} \frac{p_i}{1 - p_{n+1}} f(x_i) \\
&\geq p_{n+1} f(x_{n+1}) + (1 - p_{n+1}) f\left(\sum_{i=1}^{n} \frac{p_i}{1 - p_{n+1}} x_i\right) && \text{(Induction hypothesis)} \\
&\geq f\left(p_{n+1} x_{n+1} + (1 - p_{n+1}) \sum_{i=1}^{n} \frac{p_i}{1 - p_{n+1}} x_i\right) && (f \text{ convex}) \\
&= f\left(\sum_{i=1}^{n+1} p_i x_i\right).
\end{aligned}$$

Since the base case and the induction step hold, we conclude that it holds $\forall n \in \mathbb{N}_{>1}$. □

Corollary 3.6 (Log sum inequality)**.** Let $a_1, \ldots, a_n, b_1, \ldots, b_n \geq 0$ and let $a = \sum_i a_i$ and $b = \sum_i b_i$. Then

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq a \log \frac{a}{b}. \tag{3.9}$$

Proof. Let $f(x) = x \log x$. It is easy to convince oneself that f is convex, and that it lets us

rewrite the left side of Equation (3.9). We then have

$$\begin{aligned}\sum_i a_i \log \frac{a_i}{b_i} &= \sum_i b_i f\left(\frac{a_i}{b_i}\right) = b \sum_i \frac{b_i}{b} f\left(\frac{a_i}{b_i}\right) \\ &\geq b \cdot f\left(\sum_i \frac{b_i}{b} \frac{a_i}{b_i}\right) \qquad \text{(Theorem 3.5)} \\ &= b \cdot f\left(\frac{1}{b} \sum_i a_i\right) = b \cdot f\left(\frac{a}{b}\right) \\ &= a \log \frac{a}{b}.\end{aligned}$$

□

We now have all the tools ready to prove Theorem 3.4. Our proof follows the one laid out in [8], with slight modifications.

Proof of Theorem 3.4. Let $\rho = \sum_i p_i \left|i\right\rangle\!\left\langle i\right|$ and $\sigma = \sum_j q_j \left|j\right\rangle\!\left\langle j\right|$ be orthonormal decompositions for ρ and σ. From Definition 3.3 we can write

$$S(\rho \,||\, \sigma) = \sum_i p_i \log p_i - \sum_i \left\langle i\right| \rho \log \sigma \left|i\right\rangle . \tag{3.10}$$

With the eigendecomposition of ρ and σ we can write $\left\langle i\right| \rho = p_i \left\langle i\right|$ and

$$\left\langle i\right|\log \sigma\left|i\right\rangle = \left\langle i\right|\left(\sum_j \log q_j \left|j\right\rangle\!\left\langle j\right|\right)\left|i\right\rangle = \sum_j P_{ij} \log q_j, \tag{3.11}$$

with $P_{ij} = \left\langle i|j\right\rangle \left\langle j|i\right\rangle \geq 0$. Plugging this into Equation (3.10) yields

$$S(\rho \,||\, \sigma) = \sum_i p_i \left(\log p_i - \sum_j P_{ij} \log q_j\right). \tag{3.12}$$

Note that P_{ij} satisfies $\sum_i P_{ij} = \sum_j P_{ij} = 1$. We can thus interpret the last term in Equation (3.12) as an average of $-\log q_j$. Since $-\log$ is a strictly convex function it follows from Theorem 3.5 that

$$-\sum_j P_{ij} \log q_j \geq -\log r_i, \tag{3.13}$$

where $r_i = \sum_j P_{ij} q_j$, with equality iff. there exists a value of j where $P_{ij} = 1$. This implies

$$S(\rho \,||\, \sigma) \geq \sum_i p_i \log \frac{p_i}{r_i}, \tag{3.14}$$

where the equality occurs iff. there exists a j with $P_{ij} = 1$, i.e. iff. P_{ij} is a permutation matrix.

By Corollary 3.6 and the double stochasticity of P_{ij} we obtain the final result

$$S(\rho \,||\, \sigma) \geq \sum_i p_i \log \frac{p_i}{r_i} \geq 1 \cdot \log \frac{1}{1} = 0. \tag{3.15}$$

□

This result is essentially already our upper bound on the entanglement entropy, since we have

$$S(\rho \,||\, \sigma) \geq 0 \Leftrightarrow -\operatorname{Tr}\left[\rho \log \sigma\right] \geq S\left(\rho\right) \tag{3.16}$$

with the relative and von Neumann entropy $S(\rho \,||\, \sigma)$ and $S(\rho)$ respectively. Since the von Neumann entropy is a more general form of the entanglement entropy, we can employ Klein's inequality to upper bound the entanglement entropy as well and thus try to find a signature of the phase transition.

This idea was used in [3] for a hybrid circuit of Haar-random 2-qubit unitaries and local Z measurements. As discussed in Section 1.2.2, simulating such a system is computationally expensive. Thus, to test their approach they employed matrix product state (MPS) simulations. These allow for a broader spectrum of mixed states than stabilizers do, and it is computationally expensive, but feasible, to diagonalize the density matrix and take its logarithm, to then compute the cross entropy upper bound. In our case, however, we investigate a random circuit consisting of Pauli measurements only and thus employ stabilizer simulations to run simulations on a classical computer. As elaborated in Sections 1.1.4 and 4.1.1, the stabilizer formalism allows for lots of elegant computational shortcuts, allowing efficient quantum simulations on classical computers. Consequently, one might ask if there also is an efficient way to compute the cross or relative entropy. We will explore this question in the next section.

3.1.2 Stabilizers

When researching the quantum relative entropy and cross entropy and possible expressions thereof in the stabilizer formalism, we have not been successful in finding an efficient method, such as the one for the entanglement entropy (see Ref. [27]) [8, 75–83]. In this section we will therefore examine the upper bound on the entanglement entropy given by Theorem 3.4 in the context of stabilizers and the stabilizer formalism. In particular, we will explore the condition of infinite relative entropy, provide and prove a necessary and sufficient condition for $S(\rho \,||\, \sigma) < \infty$ when ρ and σ are stabilizer density matrices (Theorem 3.8), and derive an expression for finite cross and relative entropy.

Recall that we have $S(\rho \,||\, \sigma) < \infty$ for $\operatorname{supp}(\rho) \subseteq \operatorname{supp}(\sigma)$. For the proof of Theorem 3.8 it will prove useful to introduce an auxiliary lemma, which relates the support of a density matrix to the stabilized subspace.

Lemma 3.7. Let ρ be an N-qubit stabilizer density matrix with stabilizer group $\mathcal{S} = \langle g_1, \ldots, g_n \rangle$, and $0 \leq n \leq N$. Then

$$\text{supp}(\rho) = V_{\mathcal{S}}.$$

Proof of Lemma 3.7. By definition (see Definition 1.10), we take $V_{\mathcal{S}}$ to be the vector space stabilized by $\mathcal{S}$. Further, it is the intersection of subspaces fixed by each operator in $\mathcal{S}$, i.e. the eigenvalue one eigenspaces of elements of $\mathcal{S}$ (see Section 1.1.3). More formally we can write

$$V_{\mathcal{S}} = \bigcap_{g \in \mathcal{S}} \{ |\psi\rangle \mid g |\psi\rangle = |\psi\rangle \} = \{ |\psi\rangle \mid g |\psi\rangle = |\psi\rangle \; \forall g \in \mathcal{S} \}.$$

This subspace is projected onto by

$$\mathbb{P}_{\mathcal{S}} \equiv \frac{1}{2^n} \prod_{g \in \mathcal{S}} (\mathbb{1} + g).$$

Recall from Section 1.1 and Definition 1.11 that ρ can be written as a product of projectors

$$\rho = \frac{1}{2^N} \prod_{g \in \mathcal{S}} (\mathbb{1} + g) = 2^{n-N} \mathbb{P}_{\mathcal{S}}$$

We thus have

$$\text{supp}(\rho) = \text{supp}(\mathbb{P}_{\mathcal{S}}) = V_{\mathcal{S}}.$$

□

Theorem 3.8. Let ρ and σ be N-qubit stabilizer density matrices with respective stabilizer groups $\mathcal{S}_\rho$ and $\mathcal{S}_\sigma$. Then

$$\text{supp}(\rho) \subseteq \text{supp}(\sigma) \iff \mathcal{S}_\sigma \leq \mathcal{S}_\rho.$$

That is, with Equation (3.6), $S(\rho \,||\, \sigma)$ takes on finite values iff. $\mathcal{S}_\sigma$ is a subgroup of $\mathcal{S}_\rho$.

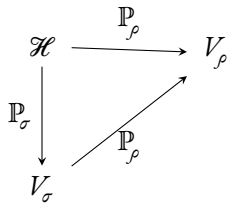
Proof of Theorem 3.8. We will prove implication from both directions to prove equivalence.

"⟸" Let $\mathcal{S}_\sigma \leq \mathcal{S}_\rho$. Then

$$\begin{aligned}
\operatorname{supp}(\rho) = V_{S,\rho} &= \bigcap_{g\in\mathcal{S}_\rho} \{|\psi\rangle \mid g|\psi\rangle = |\psi\rangle\} \\
&= \underbrace{\bigcap_{g\in\mathcal{S}_\sigma} \{|\psi\rangle \mid g|\psi\rangle = |\psi\rangle\}}_{V_{S,\sigma}} \cap \bigcap_{g\in\mathcal{S}_\rho\setminus\mathcal{S}_\sigma} \{|\psi\rangle \mid g|\psi\rangle = |\psi\rangle\} \quad (\mathcal{S}_\sigma \leq \mathcal{S}_\rho) \\
&= V_{S,\sigma} \cap \bigcap_{g\in\mathcal{S}_\rho\setminus\mathcal{S}_\sigma} \{|\psi\rangle \mid g|\psi\rangle = |\psi\rangle\} \\
&\subseteq V_{S,\sigma} = \operatorname{supp}(\sigma)
\end{aligned}$$

which finishes the proof of this direction.

"⟹" Let $V_{S,\rho} = \operatorname{supp}(\rho) \subseteq \operatorname{supp}(\sigma) = V_{S,\sigma}$. For the proof of this direction, consider the relations between the subspaces of the N-qubit Hilbert space $H^{\otimes N} \equiv \mathscr{H}$ outlined in the following diagram.



We can read the diagram as follows: From the N-qubit Hilbert space we can project onto the vector space stabilized by $\mathcal{S}_\sigma$, V_σ, by means of a projection operator. Likewise we can do the same for V_ρ. Since we require $V_{S,\rho} \subseteq V_{S,\sigma}$, the same projection that takes us from $H^{\otimes N}$ to $V_{S,\rho}$ will take us from $V_{S,\sigma}$ to $V_{S,\rho}$. It follows that $\mathbb{P}_\sigma\mathbb{P}_\rho = \mathbb{P}_\rho$.

Let $|\psi\rangle \in V_\rho$. We thus have

$$|\psi\rangle = \mathbb{P}_\rho|\psi\rangle = \mathbb{P}_\sigma\mathbb{P}_\rho|\psi\rangle = \mathbb{P}_\sigma|\psi\rangle .$$

Since $|\psi\rangle$ was an arbitrary element from $V_{S,\rho}$ and $\mathbb{P}_\sigma = \frac{1}{2^n}\prod_{i=1}^{n}(\mathbb{1} + h_i)$ with $h_i \in \mathcal{S}_\sigma$ it follows that all stabilizers of σ also stabilize ρ and thus

$$\mathcal{S}_\sigma \leq \mathcal{S}_\rho.$$

This concludes the proof. □

We have thus shown that the condition for finite values of the relative entropy is equivalent to a simple statement about the group structure of the respective stabilizer groups. Furthermore, we know from [27] that the entropy of entanglement can be expressed in a

simple way through group properties. We'd like for this to also be the case for other entropic quantities, especially the cross and relative entropy, which are our main concern.

It turns out that one can indeed derive expressions for the cross and relative entropy that put these information-theoretic quantities in a relation with abstract group properties. Confer with Theorem 3.10 and Corollary 3.11 for the precise statements. The resulting expressions are remarkable in their simplicity, as well as their similarity to each other and to previous results. As a warm-up for the proofs of Theorem 3.10 and Corollary 3.11 we will derive an expression for the von Neumann entropy in the stabilizer formalism first. This will also aid in the derivation of the expression for relative entropy, since it is a difference of the cross and von Neumann entropy.

Lemma 3.9 (Von Neumann entropy – stabilizers)**.** Let ρ be an N-qubit stabilizer density matrix with stabilizer group $\mathcal{S}_\rho$ and rank $\left|\mathcal{S}_\rho\right| \equiv r$. Then

$$S\left(\rho\right) = N - r. \tag{3.17}$$

Proof. Let $\mathcal{S}_\rho$ be an N-qubit stabilizer group of rank r with corresponding density matrix ρ.

By Definition 3.2, or Equation (3.3) in particular, we have

$$S\left(\rho\right) = -\operatorname{Tr}\left[\rho \log \rho\right] = -\sum_k \lambda_k \log \lambda_k. \tag{3.18}$$

Thus, we need to diagonalize, i.e. find the eigenvalues λ_k of ρ.

Since ρ is a stabilizer density matrix, it can – up to a constant multiple – also be written as a product of projectors onto the +1 eigenspaces of group generators

$$\rho = \frac{1}{2^N} \prod_{i=1}^{r} (\mathbb{1} + g_i) = 2^{r-N} \mathbb{P}_\rho. \tag{3.19}$$

Knowing that projections are diagonalizable with eigenvalues of either 0 or 1, we know that the diagonal form of ρ is the diagonal form of $\mathbb{P}_\rho$ with a constant multiple 2^{r-N}, that is[4]

$$D_\rho = 2^{r-N} \begin{pmatrix} 1 & & \\ & \ddots & \\ & & 0 \end{pmatrix}. \tag{3.20}$$

With $\operatorname{Tr}\left[\rho\right] = 1$ it follows that the eigenvalues of ρ must be $\lambda_k = 2^{r-N}$ for $k = 1, \ldots, 2^{N-r}$ and $\lambda_k = 0$ for all other k. Inserting this back into Equation (3.18) yields

$$\begin{aligned} S\left(\rho\right) &= -\sum_{k=1}^{2^N} \lambda_k \log \lambda_k = -\sum_{k=1}^{2^{N-r}} 2^{r-N} \log 2^{r-N} = -2^{N-r} 2^{r-N} \log 2^{r-N} = -\log 2^{r-N} \\ &= N - r. \end{aligned}$$

[4] Note that the bottom right entry need not be zero. This specific choice was made for illustratory purposes.

□

The expression of the von Neumann entropy is not only interesting going forward, e.g. in the derivation of the relative entropy, but also from a group theoretic perspective. That is, it should be noted that by asking an information-theoretic question, we got a group theoretic answer. Let us convince ourselves that it is a reasonable one, by testing it on our intuition on entropy.

With N qubits we have a generating set of size N, at most. In the case where we have N generators, the state associated with the stabilizer group is a pure state, and the entropy should be 0, which is the case for $r = N$. For the maximally mixed state, the generating set is the empty set and the stabilizer group the trivial group. In accordance to our expectations, this should yield $\log(2^N) = N$ for the von Neumann entropy. Indeed, since the size of the empty set is 0, we have that $r = 0$ and thus $S(\rho) = N$.

Removing one generator thus increases entropy by 1, or put differently, starting from the empty set and adding generators to it decreases entropy by 1 for each generator added. This too should come to no surprise, as we know that for each generator removed (added) from a full set of stabilizers, we double (halve) the stabilized state space, effectively creating (purifying) a perfect mixture of stabilized states.

Theorem 3.10 (Cross entropy – stabilizers)**.** Let ρ and σ be N-qubit stabilizer density matrices with respective stabilizer groups $\mathcal{S}_\rho$ and $\mathcal{S}_\sigma$ that satisfy $\mathcal{S}_\sigma \leq \mathcal{S}_\rho$. Further, let $|\mathcal{S}_\sigma| \equiv s$. Then

$$S_C(\rho \,||\, \sigma) = N - s. \tag{3.21}$$

Proof. Let $\mathcal{S}_\rho$ and $\mathcal{S}_\sigma$ be N-qubit stabilizer groups that satisfy $\mathcal{S}_\sigma \leq \mathcal{S}_\rho$. Their respective density matrices can be written as

$$\rho = \frac{1}{2^N} \prod_{i=1}^{r} (\mathbb{1} + g_i) \quad \text{and} \quad \sigma = \frac{1}{2^N} \prod_{i=1}^{s} (\mathbb{1} + h_i) \tag{3.22}$$

with $r \equiv \left|\mathcal{S}_\rho\right|$ and $s \equiv |\mathcal{S}_\sigma|$. Since we require $\mathcal{S}_\sigma \leq \mathcal{S}_\rho$, we can construct a generating set G_σ of S_σ, where each element of G_σ commutes with the generating set of G_ρ. Note that the generating sets are not necessarily identical. However, since $\mathcal{S}_\sigma$ is by construction a subgroup of $\mathcal{S}_\rho$, which in turn is an abelian subgroup of $\mathcal{P}_N$, all elements of $\mathcal{S}_\sigma$ commute with all elements of $\mathcal{S}_\rho$. It follows that the density matrices themselves also commute, i.e.

$$\left[\rho, \sigma\right] = 0. \tag{3.23}$$

It is a well-known fact from linear algebra and functional analysis that commuting operators share a common eigenbasis and can be diagonalized simultaneously. We therefore write

$\rho = UD_\rho U^{-1}$ and $\sigma = UD_\sigma U^{-1}$ with transformation matrix U and corresponding diagonal matrix $D_{\rho/\sigma}$. We also have that the logarithm of a diagonalizable matrix is [49]

$$\log \sigma = U \log D_\sigma U^{-1}.$$

Consequently,

$$S_C(\rho \,||\, \sigma) = -\operatorname{Tr}\big[\rho \log \sigma\big] = -\operatorname{Tr}\big[UD_\rho U^{-1} U \log D_\sigma U^{-1}\big] = -\operatorname{Tr}\big[D_\rho \log D_\sigma\big]. \tag{3.24}$$

With D_ρ and D_σ diagonal we can write the trace as

$$-\operatorname{Tr}\big[D_\rho \log D_\sigma\big] = -\sum_k \lambda_k \log \mu_k \tag{3.25}$$

with the eigenvalues λ_k and μ_k of ρ and σ, respectively.

As we know from the previous proof, $\lambda_k = 2^{r-N}$ for $k = 1, \dots, 2^{N-r}$ and 0 otherwise, and analogously, $\mu_k = 2^{s-N}$ for $k = 1, \dots, 2^{N-s}$ and 0 otherwise. Note that the subgroup condition implies that $r \geq s$ or more specifically $N - r \leq N - s$. This ensures that to every non-zero entry in D_ρ there is a corresponding non-zero entry in D_σ. In other words, for all k, $\lambda_k \neq 0 \implies \mu_k \neq 0$. More importantly for us, however, is the contrapositive, $\mu_k = 0 \implies \lambda_k = 0$, which tells us that any divergence that might occur in the logarithm gets intercepted by a leading factor of 0.

Inserting this into the sum in Equation (3.25) we get

$$-\sum_i 2^{r-N} \log 2^{s-N} = -2^{N-r} 2^{r-N} (s - N) = N - s, \tag{3.26}$$

which concludes the proof. □

Although Theorem 3.10 and Equation (3.21) are in dire need to be discussed, we do not want to fail to mention that the relative entropy between two stabilizer density matrices follows as a corollary.

Corollary 3.11 (Relative entropy – stabilizers)**.** Let ρ and σ be N-qubit stabilizer density matrices with respective stabilizer groups $\mathcal{S}_\rho$ and $\mathcal{S}_\sigma$, where $\left|\mathcal{S}_\rho\right| \equiv r$ and $|\mathcal{S}_\sigma| \equiv s$. If $\mathcal{S}_\sigma \leq \mathcal{S}_\rho$, then the relative entropy of ρ to σ is

$$S\left(\rho \,||\, \sigma\right) = r - s. \tag{3.27}$$

Proof. Let ρ and σ be density matrices satisfying the stated requirements, $\mathcal{S}_\sigma \leq \mathcal{S}_\rho$ in particular. From Definition 3.3, or more precisely Equation (3.4), we have

$$S\left(\rho \,||\, \sigma\right) = -S\left(\rho\right) - \operatorname{Tr}\big[\rho \log \sigma\big]. \tag{3.28}$$

Inserting our results from Lemma 3.9 and Theorem 3.10 yields

$$-S\left(\rho\right) - \mathrm{Tr}\left[\rho \log \sigma\right] = -(N-r) + N - s = r - s. \tag{3.29}$$

□

Notice that although it is defined as a quantity relating ρ to σ, there is no explicit dependence of ρ in Equation (3.21). What Theorem 3.10 seems to imply is that the cross entropy between two stabilizer density matrices ρ and σ just amounts to the von Neumann entropy of σ. However, this is where one needs to be careful. While the resulting numerical value is entirely dependent on σ and its stabilizer group alone, we have an implicit dependence through the requirement that $\mathcal{S}_\sigma \leq \mathcal{S}_\rho$. Since we were able to show that the cross entropy is infinite in the converse case (see Theorem 3.8), we could add the ρ dependence back in, i.e.

$$-\mathrm{Tr}\left[\rho \log \sigma\right] = \begin{cases} N - |\mathcal{S}_\sigma| & \mathcal{S}_\sigma \leq \mathcal{S}_\rho \\ \infty & \mathcal{S}_\sigma \not\leq \mathcal{S}_\rho \end{cases}. \tag{3.30}$$

This detail will be important later on, as we try to avoid explicit dependences on ρ, which would take us back to the sampling problem.

The independence of ρ in the first case, however, is still quite a remarkable result. Let's examine the expression and convince ourselves that this simplicity is no accident. From the subgroup condition we know that the state described by σ needs to be more mixed or at least as mixed as ρ. That is, the probability distribution on the state space represented by σ contains more states than ρ. However, all states of ρ are featured in σ with uniform probability. Any state in the mixture of σ thus gets assigned the same weight when averaging. Additionally, the probabilities over the larger state space are uniformly distributed as well, turning $\log \mu_k$ into a constant factor. It is therefore an average of a constant over a uniform probability distribution, which is just the constant itself.

We have discussed the mathematical and information-theoretic aspect of the derived statements, but we should also think about the physical implications of these statements. Although we summarized the idea, we want to reemphasize the utility behind these expressions. Our ultimate goal is an upper bound on the half-system entanglement entropy, $S_E\left(\rho\right)$,[5] which can be derived from Equation (3.16). With that, we try to detect a signature of the critical point of the phase transition found by classical simulations. Recall that the half-system entanglement entropy is non-linear in the density matrix, that is, it is an observable of the state itself, requiring exponentially many copies of the same state. Therefore, it is infeasible, and practically impossible, to perform measurements quantifying entanglement with the entanglement entropy. However, by Equation (3.16) we can obtain an upper bound, $S_E \leq -\mathrm{Tr}\left[\rho \log \sigma\right]$, thus circumventing the sampling problem. The choice of σ has some

[5] Here, ρ refers to the reduced density matrix of half the system, where the entire system is in the pure state $|\phi\rangle$

degree of arbitraryness; as long as supp $(\rho) \subseteq$ supp (σ) we get a finite upper bound, and even have equality if $\rho = \sigma$. But this is not the whole picture. It stands to reason that we *should* be picky with our choice of σ and be consistent with it. In particular, we ideally would like to relate σ to the experimental state ρ in some way or another. This can be done by classically reconstructing ρ by means of projecting the measurement outcomes from the record $\mathbf{m}$. Then, by tracing out half of the system, we obtain some σ, which we can use in the upper bound. The classical post-processing of the measurement record requires us to efficiently simulate the system, as well as to be able to project onto specific outcomes. To this end, we prefer to use stabilizer simulations, as they allow the efficient simulation and post-processing of our system. With Theorem 3.10 we now additionally have an efficient method of computing the upper bound to the entanglement entropy, such that the inequality becomes

$$S_E \leq N - s. \tag{3.31}$$

The following sections therefore pertain to the various possible post-processing algorithms one might make use of.

3.2 NAIVE APPROACH

The key idea of the upper bound is to have a quantity that is linear in ρ. By obtaining a measurement record $\mathbf{m}$, we can try and classically reconstruct the experimental density matrix ρ from the record. Since we are doing a classical computation, we have the advantage of being able to project onto measurement results. We therefore can – naively – attempt a reconstruction of our experiment based on the data. Of course, in our case, we also use a classical simulator to generate data.

3.2.1 Methods

After the data has been generated we peform the same protocol as in Chapter 2 and project the measurement outcomes onto σ. However, we start from the same state $\rho = \sigma = |GHZ+\rangle\langle GHZ+|$. After we have completed the circuit, we trace out one half of the qubits and compute the cross entropy, which is given by Equation (3.21).

For our prospective numerical experiments, we would therefore have them support the simulation of mixed states, since we might encouter them by means of partially tracing out the system. To this end, we refer to Chapter 4, in particular Sections 4.3, 4.3.3 and 4.3.4. In these sections, we introduce the algorithms used to realize mixed states, the different entropic quantities, and the partial trace, respectively. Of course, we additionally need to adapt the projection function, since we now *force* the projection upon the state. For the computation of the linear cross entropy, it sufficed to know if projections were successful all the way through. If they were not, we would simply break the loop over the measurement

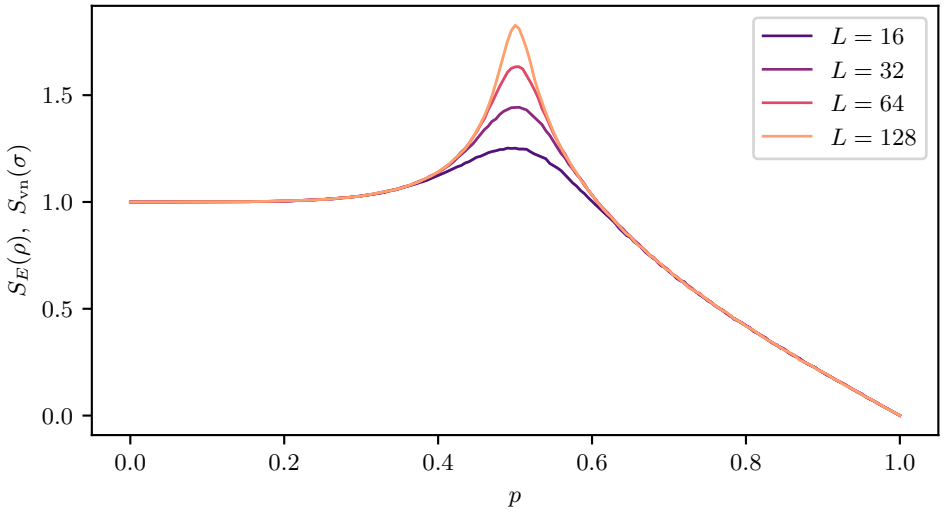


Figure 3.1.: Cross and entanglement entropy in the noiseless system. We project the measurement outcomes from ρ onto σ if possible. Note that we here use periodic boundary conditions. Each datapoint corresponds to $\sim 10^5$ samples.

record and stop the projections. Here, we really want a faithful reconstruction of the density matrix. The algorithm to force projections is outlined in Algorithm 4.10.

As we use the general setup from Chapter 2, we also employ the error model introduced in Section 2.4.1 to emulate noise in the circuit. This is to ensure comparability between the general approaches of the linear cross entropy and the cross entropy as upper bound.

As we have access to the "experimental" density matrix ρ through our numerical simulation, we employ the implementation of the cross entropy as outlined in Theorem 3.10. An important detail to note, however, is that we do not want infinities appearing in our data. As such, in the case of $\mathcal{S}_\sigma \not\leq \mathcal{S}_\rho$, we add $N/2$, where N is the number of qubits in the system. This amounts to replacing the density matrix σ by the maximally mixed state.

3.2.2 Results

In Figure 3.1 the results of the upper bound are shown in the case of no noise in the circuit. Figure 3.1 shows that there is perfect overlap between the original and the replicated system in the noiseless case. This is also what one would expect, since we effectively computed the entanglement entropy of a pure state that happens to have had the same measurement record as ρ. And incidentally, every outcome in the measurement record corresponded to an outcome that could be projected perfectly onto σ. We thus perfectly saturated the upper bound Equation (3.16).

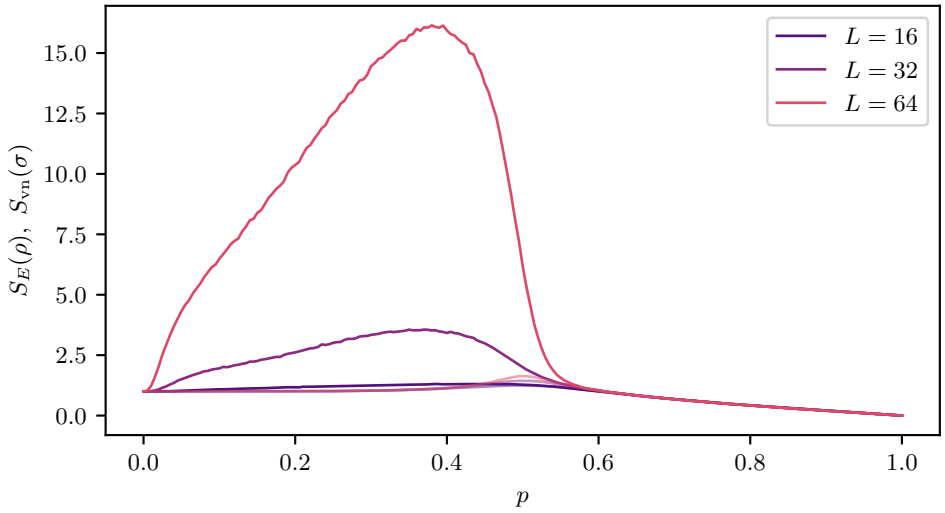


Figure 3.2.: Cross and entanglement entropy in the system with projective X errors. Measurement outcomes from a simulated experiment are projected onto a replica in the classical simulation. If the subgroup condition is not met, we replace the appearing infinity with $L/2$.

We now turn towards a more realistic scenario, namely that of a noisy circuit. The error model we use to emulate noise is the one introduced in Section 2.4.1. Note that we force the projections onto σ, and thus should recover a close estimate of the experimental density matrix. If it turns out that the density matrix obtained from the experiment does not have support in σ, we replace the appearing infinity with $L/2$, where L is the number of qubits.

Figure 3.2 shows that the naive approach fails here already. In the $p < p_c$ regime we often need to correct for infinities in the sample, which get replaced, yes, but since the trivial upper bound is so far off the actual entanglement entropy, we have this diverging behavior even for relatively small systems, which is why only systems up to $L = 64$ are shown. For $p > p_c$ we saturate the bound again, since the X noise commutes with the then more frequently occuring X measurements of the circuit, thereby restoring the adherence to the subgroup condition.

This tells us that we need some way to cope with noise. The subgroup condition requires us to detect the errors, since errors which are not succeeded by a measurement (of any kind) will lead to a change in the structure of $\mathcal{S}_\rho$, which is never detected. This undetectable change, as we can tell from Figure 3.2, happens if an error occurs on a qubit in a state stabilized by an operator, which anticommutes with the type of noise, for instance, if an X error occurs on a qubit stabilized by Z. Since X measurements are more frequent for $p > 1/2$, the X noise is directly mitigated by the frequent measurements.

What about the part where X measurements are not as frequent? We can again estimate a probability of not detecting the altering of the stabilizers. Imagine an error occurs in the very last layer, i.e. after all the measurements in the circuit happened. This is the complementary probability to no errors occuring. We thus have

$$P(\#\text{err} > 0) = 1 - (1 - q)^L. \tag{3.32}$$

As simple example, consider the probability of an X error occuring in the very last layer of a system with $L = 32$ qubits. The probability that an error does not get detected is $P \simeq 0.275$. Thus, about one in every four samples would yield infinity as upper bound, which is useless. We thus make the rather unphysical assumption that no error is allowed to happen after a certain threshold. As we design the circuit in a classical simulator as well, we simply restrict the allowed errors occurring.

Furthermore, recall that the cross entropy is a measure of the quality of an estimate of a probability distribution [84]. As we are essentially estimating the experiment, represented by the density matrix ρ, with a uniform probability distribution over the support of ρ, the quality of our estimate depends only on how many events (or in this case, states) we choose to include in our estimate. In other words, we project onto an eigenstate $|\phi\rangle$ of ρ, where our estimate of the probability of $|\phi\rangle$ is scaled by how many other states we consider with equal probability. Alternatively, in the case where the subgroup condition is not met, we do *not* have $|\phi\rangle$ in supp (σ). In this scenario, we project onto the 0-vector, giving us a divergent term in the cross entropy. This also shows how stabilizer states are particular in that sense. Any stabilizer density matrix we choose for σ will yield this result, independent of ρ, as long as ρ is a valid density matrix.

This means that if we were to continue with the cross entropy and the upper bound in a pure stabilizer setting, we should think about how we could have a finer-grained selection of values the cross entropy can take up. If we only insert pure states, we might get lucky a couple of times, but in all generality, we will fail to estimate the density matrix obtained in an experiment. Thus, for the purely numerical approaches, we should abandon the naive one in favor of reconstruction algorithms that include a broader set of states to support ρ. Of course, they should, as we have been made aware of by the previous results, ensure the fulfillment of the subgroup condition as well.

3.3 OTHER NUMERICAL APPROACHES

We have seen that the naive approach fails once noise comes into play, especially if the modelled errors are not succeeded by a measurement natively included in the circuit. We have also noted that due to the limitations of the stabilizer formalism, we either end up in agreement with the experiment or add infinities to the mix. As we would like to stay in the stabilizer formalism, since it allows polynomial-time simulation of the quantum circuit in question, we need to cope with noise in a different way, that also encompasses the tools we have at our disposal.

We have further found an irreconcilable limit to Equation (3.16) in the form of errors appearing in the last measurement layer. For all the further considerations, we disallow errors after the last measurements. This is, of course, an unphysical assumption to make, but it is, as of right now, not our main concern, as we here want to test the limits of Equation (3.16) on our system. This uncertainty of errors within the very last layers seems to be one of these limits. If we relax the assumption of errors happening everywhere, we could try other numerical approaches and maybe get an idea of how viable the general approach of an upper bound is to find a signature of the entanglement transition. With the primer of the previous protocol failing, we also include two new quantities next to the cross entropy.

The first one is the "infinity ratio", i.e. the ratio of simulations not agreeing with the subgroup condition, therefore having diverging contributions in the sample average, which we regularize "primitively" by replacing it with the trivial upper bound $L/2$. Since this is once more not how a conventional physical observable behaves, we keep track of how often we cheat in the simulation algorithm. This measure therefore quantifies if the algorithm we conceived really did work as intended or if we made conceptual errors. Furthermore, it aids in explaining the behavior of the cross entropy, as we can then gauge if anomalously high values for the upper bound come from the algorithm or the primitive regularization.

To this end, we also have the next quantity, which is the von Neumann entropy of the entire density matrix before the partial trace over half the system. This measure quantifies the degree of mixedness left in the system at the very last timestep. For the naive approach, this last quantity is obviously 0 everywhere, since it produces pure states only.

The following sections introduce the methodology behind two numerical post-processing algorithms that utilize mixed states with the aim to uphold the subgroup condition, as well as show the results of the performed simulations.

3.3.1 Minimal Mixing

The first post-processing algorithm we introduce is what one could call "minimal mixing". In the following, we will introduce the algorithm and the results. Note that the presentation of the results parallels the presentation shown in Figure 2.7, with the methods to obtain this mirroring the ones in Chapter 2. As such, we assume knowledge of how Figure 2.7 is to be

read.

The algorithm

We start our classical reconstruction in the same state as the initial state of the experiment, which is $|GHZ+\rangle\langle GHZ+|$ in particular. Then, we project onto the measurement outcomes where possible. In a noiseless circuit, this recovers the standard entanglement entropy once more. Once we introduce noise this will no longer be given. Recall the scenario shown in Figure 2.4. In this case we had that the projection was successful only with probability $1/2$, i.e. unsuccessful in about half of the performed projections.

For this numerical approach, we choose to ignore the stabilizer generator with the failed projection. That is, we replace the stabilizer by $\mathbb{1}$ where the incompatibility was detected, instead of forcing the projection. This should, in principle uphold the subgroup condition, since we remove particular generators from the generating set. Until it gets measured again, we have that this stabilizer is not a generator. The technical details behind the implementation are laid out in Section 4.3.5 and Algorithm 4.17.

The procedure to compute the cross entropy is still the same as before. We trace out one half of the system, then perform the subgroup check, and add

$$\tilde{S}_C(\rho \,||\, \sigma) = \begin{cases} N - |\mathcal{S}_\sigma| & \mathcal{S}_\sigma \leq \mathcal{S}_\rho \\ \frac{L}{2} & \mathcal{S}_\sigma \not\leq \mathcal{S}_\rho \end{cases} \tag{3.33}$$

to the sample average. In addition to the previous protocol, we also compute the von Neumann entropy of the full state obtained by the algorithm, as well as count how often we added $L/2$ in the case of a mismatching group structure.

One critique that could immediately be raised at this point is the (ab)use of mixed states in this context. Each time we come across a discrepancy, we throw out the generator in question, even though this is the most recent information we have on the state of the system. However, we have already made some unphysical assumptions before, and this is no exception. At this point we want to test the performance of the upper bound in the extreme case of stabilizers and we want to use all the tools available to us, and selectively removing stabilizer generators from the generating set is one of them. *How* we select them is then judged by the performance of the selection in the numerical simulation.

Results

Figure 3.3 shows the upper bound (opaque) and the entanglement entropy (transparent) as a function of the probability parameter p for the minimal mixing algorithm. For the "no error, track everything" case, we—unsurprisingly—recover the identical behavior to the regular entanglement entropy, since every projection is successful.

For all the marginalized runs, we can see that they end up in the infinite case rather often (see also Figure 3.4). This is because we measure instead of project. Thus, we might measure

and get a random result that is orthogonal to the one we would have projected onto, even if there is no error present. In the case where we marginalized out ZZ, i.e. the "Track X" column, we have that for no errors and $p = 0$, the cross entropy agrees with the entanglement entropy. This is because there are no X measurements to track either way, so there is nothing that can go wrong.

However, once we have even the tiniest amount of X measurements, everything goes horribly wrong. Even for relatively small systems of $L = 16$ and $L = 32$ qubits, we almost always end up at the trivial upper bound of $S_E \leq 8$ and $S_E \leq 16$ respectively. An interpretation in terms of quantum error correction would be that we did not correct an error, but chose to let it happen. In a sense, this plot tells us that we have done a poor job correcting errors, or rather that we did not do so at all.

In the case where we only tracked the outcome of ZZ measurements, this phenomenon is not as intense as it is for the outcome of X measurements. This can be explained with the fact that we perform stabilizer measurements, i.e. we know where errors happened, yes, but we do not know the outcome of them. In spirit, this is similar to a decoding or error correcting protocol [71]. Also, the scaling of this particular plot is exponential in the system size up to the theorized critical point, where it flattens and goes to the trivial upper bound.

What about the noisy circuits? Recall from the linear cross entropy that in the case of X-noise occurring, i.e. projective X errors, we had that the marginalized ZZ / Track X plot with X errors was identical to the ideal system. This was due to the fact that the linear cross entropy only contained the probabilities of success or failure, and the fact that the error type commuted with the tracked measurement outcomes. Here, we do not have this luxury. In fact, the same things that apply to the case without errors apply here. We even notice that for $p = 0$, there is no agreement anymore, since we effectively shifted the lines to the left a bit. Before, every ZZ measurement was a genuine stabilizer measurement, since we measured generators of the stabilizer group. This is not the case anymore, since even with $p = 0$, we have sporadic creation and annihilation of clusters everywhere, which perturb the entanglement structure. Sure, it should be corrected instantly, since we perform ZZ measurements everywhere at any time. But this is not enough if errors are sufficiently frequent, which is the case for larger systems. This is also the reason why we chose to show all these situations for smaller system sizes only.

When tracking only the outcomes of ZZ measurements, we find that we do not have the nice exponential growth, but a superexponential growth. the occurrence of X measurements, which we do not even try and "guess" the outcome is now larger. The regime where we only have trivial upper bounds is shifted to the left slightly. The more interesting plot is the one where we track everything. We must have failed in our quest to bring everything under the umbrella of the $\mathcal{S}_\rho$ stabilizer group, since we can see a disturbing discrepancy between S_E and the cross entropy.

This discrepancy can be explained by the fact that two measurement layers are not enough to fully rule out undetectable errors. We can have clusters emerging that fit the measurement

outcomes just enough so that they bypass our detection/error handling scheme. However, they do not bypass the subgroup check. With X-errors occurring, we, of course, correct a lot of errors, which can be seen in the fact that we have a tighter upper bound towards $p = 0$. Closer to the critical point we find a peak, where X errors are frequently contributing to the creation of unforseen clusters, which bypass the error handling scheme. In the regime of $p > p_c$, we find that there is again agreement with the entanglement entropy, since X measurements are more frequent to ZZ measurements X errors, such that noise is.

For ZZ errors, the converse is the case, while we have a combination of both divergences in the case where we include both types of errors.

We can therefore infer already that we have not done enough to sanitize the input of the subgroup check. To confirm this suspicion, consider Figure 3.4. In Figure 3.4 we show how often we add $L/2$ instead of ∞ in the data of Figure 3.3. This gives us an estimation on how much trickery is involved for our approach to not include infinities. That is, what is the ratio of simulations that have to get regularized primitively.

As expected, the different marginalizations yield a high rate of infinities, meaning that the behavior in the corresponding subplots of Figure 3.3 is largely determined by the lack of support of ρ in σ.

Remarkably, however, some of the group structure survives the impact in the case of small systems, even in the presence of errors. This can be attributed to the fact that the initial entanglement survivies in some cases where $p < p_c$. Note that it seems that the large values of the cross entropy in Figure 3.3 in the "Track all" column really do seem to stem from that we replaced with another upper bound. This is yet another indicator that even in the case where we assume that no errors occur in the last segment of the run, we can still feel the pressing weight of errors. Therefore, this algorithm is unfit for post-processing.

As measure for how much contribution the algorithm has in the end, consider Figure 3.5, where the von Neumann entropy of the full density matrix of the reconstruction attempt is shown for all the various previous cases. We can see in the "Track X" column, that there is some contribution from the algorithm itself, since we do not track the results from ZZ measurements, which would interfere with projections in X towards the regime where the initial cluster dies.

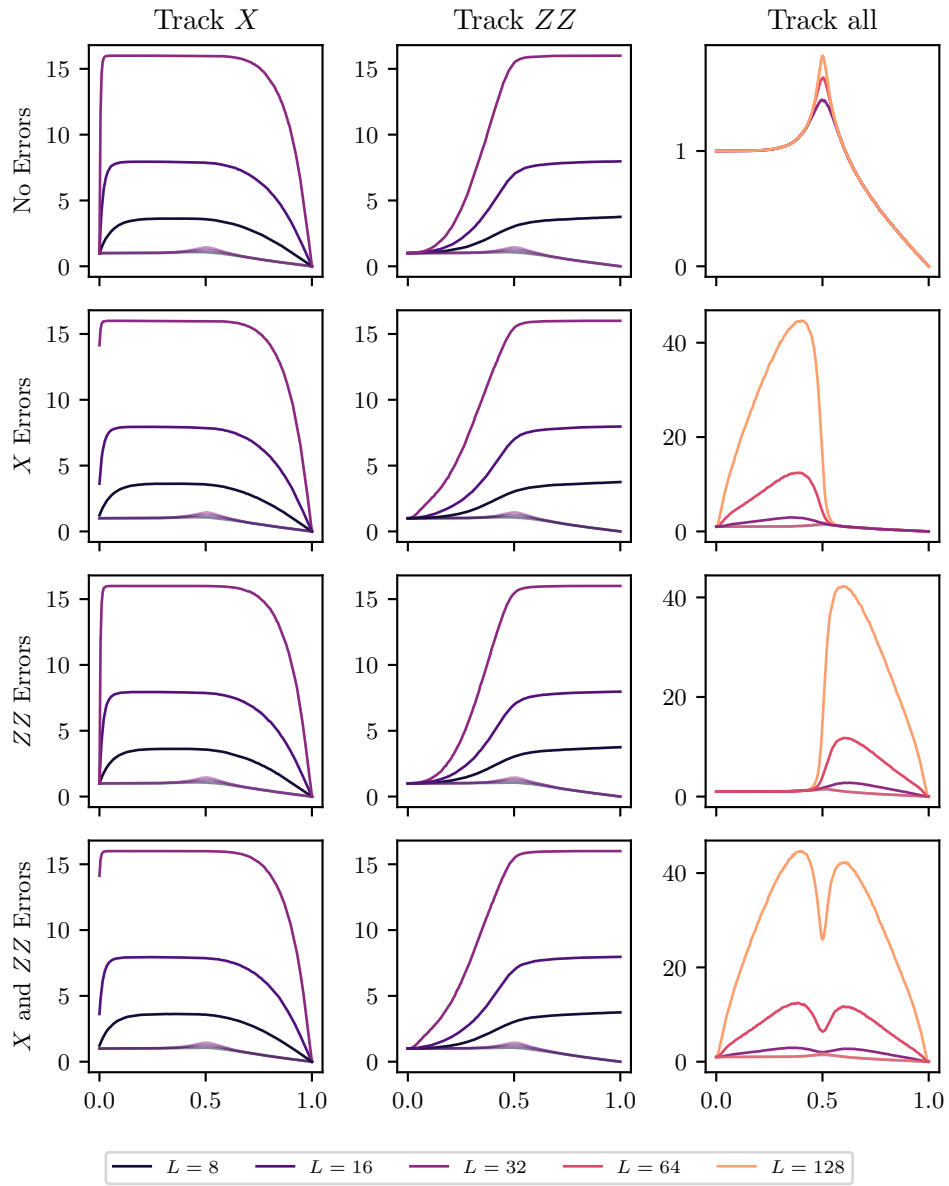


Figure 3.3.: Cross Entropy and entanglement entropy of selected system sizes with periodic boundary conditions for the "minimal mixing" algorithm. To be able to adequately compare it to LXE the different trackings and errors are shown in the same manner as in Figure 2.7.

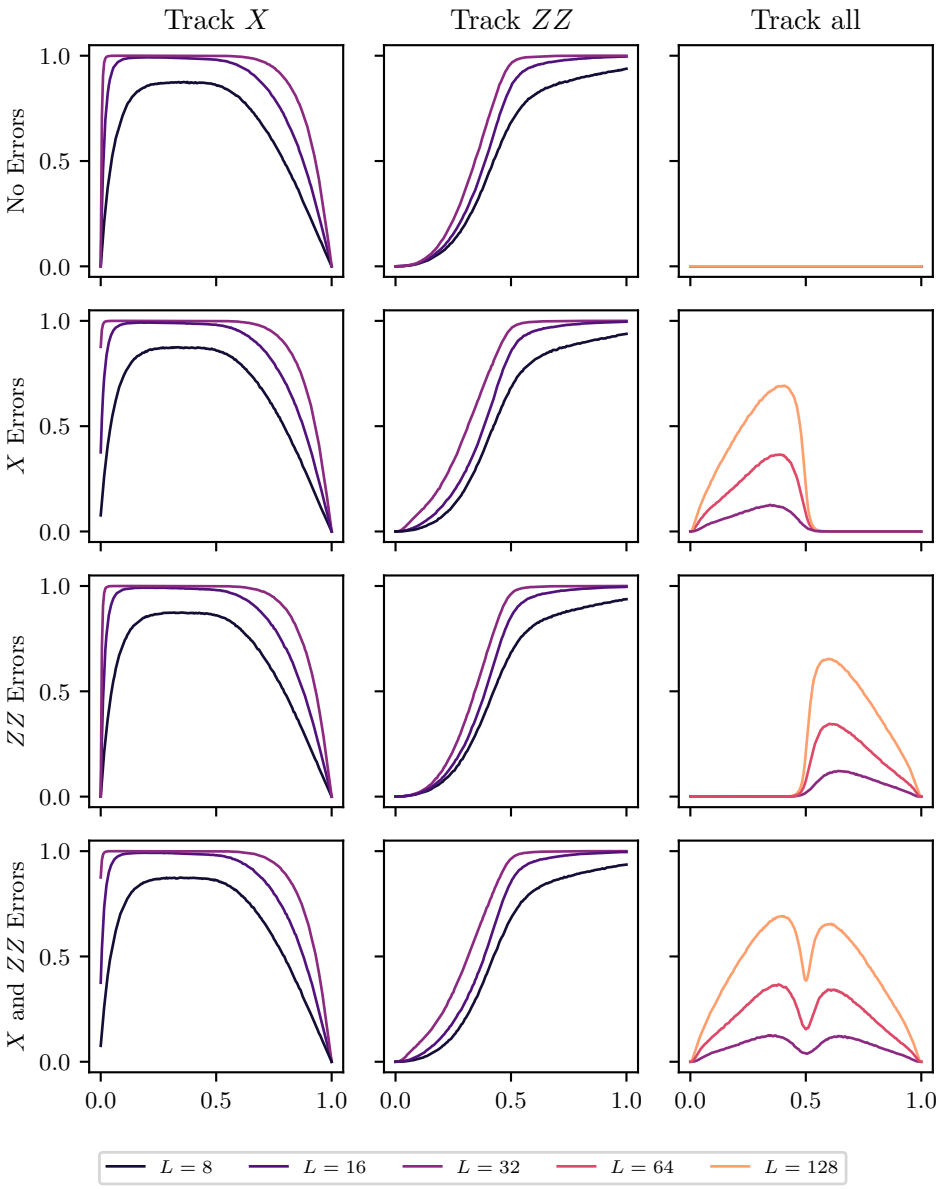


Figure 3.4.: Ratio of divergences in cross entropy to number of samples for the "minimal mixing" algorithm with different system sizes and periodic boundary conditions. To be able to adequately compare it to LXE the different trackings and errors are shown in the same manner as in Figure 2.7.

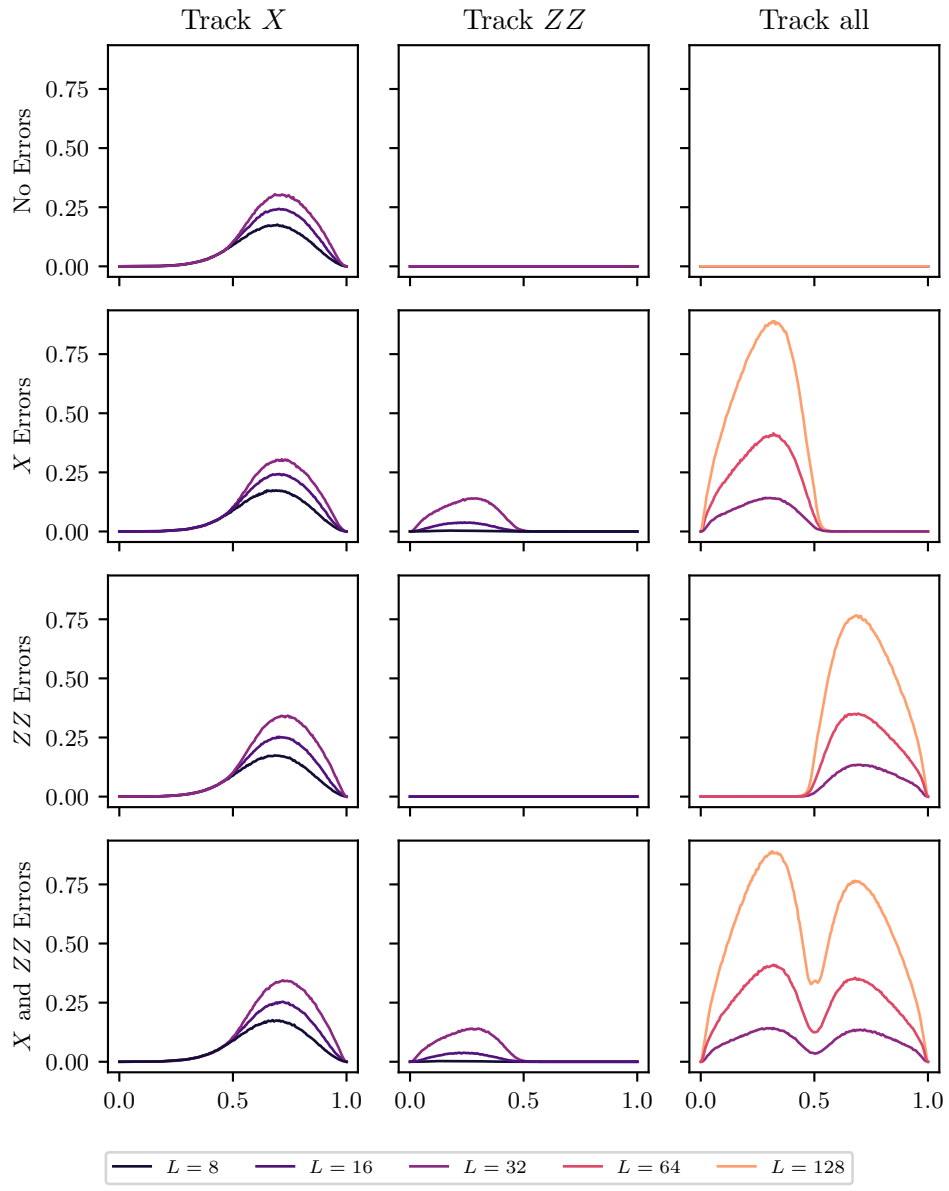


Figure 3.5.: $S(\sigma)$ of full density matrix σ periodic boundary conditions for the "minimal mixing" algorithm. This quantifies how much mixedness was left at the end of a simulation. To be able to adequately compare it to LXE the different trackings and errors are shown in the same manner as in Figure 2.7.

3.3.2 Maximal Mixing

The next algorithm is what we could call "maximal mixing".

The algorithm

It is similar to the previously introduced algorithm with the caveat that we are now not selectively removing stabilizer generators, but removing the whole generating set. This is the most extreme method of how one could deal with the noise, since now once we catch it, we in principle act as if no generator of our generating set is to be trusted. Also, as the trivial group is always a subgroup of any group (see Section 1.1.1), and as the trivial group is generated by the empty set (see Definition 1.2), we should most definitely ensure the fulfillment of the subgroup condition. The technical details behind the implementation of this algorithm is given in Section 4.3.5 and Algorithm 4.18.

Note that this approach should also saturate the upper bound perfectly in the noiseless case, since we only throw out generators when we encounter a mismatch in the projections.

Results

For the maximal mixing approach, we find that the first row of Figure 3.6 is almost identical to the corresponding plots in Figure 3.3 for the other processing algorithm. This is due to the fact that we have not introduced errors and the maximal mixing procedure has no effect on this.

We also see that almost all the "Track X" or "Track ZZ" columns are similar in appearance, except for "Track ZZ" when X errors are present. We can infer an explanation for this by considering Figure 3.8, where the von Neumann entropy of the entire density matrices is shown.

For the "Track all" columns, we have values for the cross entropy that are way smaller than with the previous algorithm. When taking Figure 3.7 into account, we notice that these contributions are (almost) entirely from the post-processing algorithm we chose. For X errors we have that the upper bound attains larger values for $p \approx p_c$ with a slight slant to the left the converse is true for ZZ errors, where the upper bound almost doesn't go back to the original curve.

For the noisiest system with X and ZZ errors, we have a combination of the two effects Interestingly, the scaling at the critical point is linear, since the peaks roughly double in magnitude with doubling the system size L. This is in contrast to the "ideal" case, where we have logarithmic scaling at the critical point The peaks in the top right subplot are spaced evenly, which is as expected with the doubling of system size.

For the infinity ratios, we have made progress: In the "Track all" column, we find that we had no infinities almost everywhere. However, there are noticeable bumps in some of the plots. They are miniscule, but still there. So miniscule that we can rule out the possibility of them substantially contributing to the behavior shown in Figure 3.6. Nonetheless, this

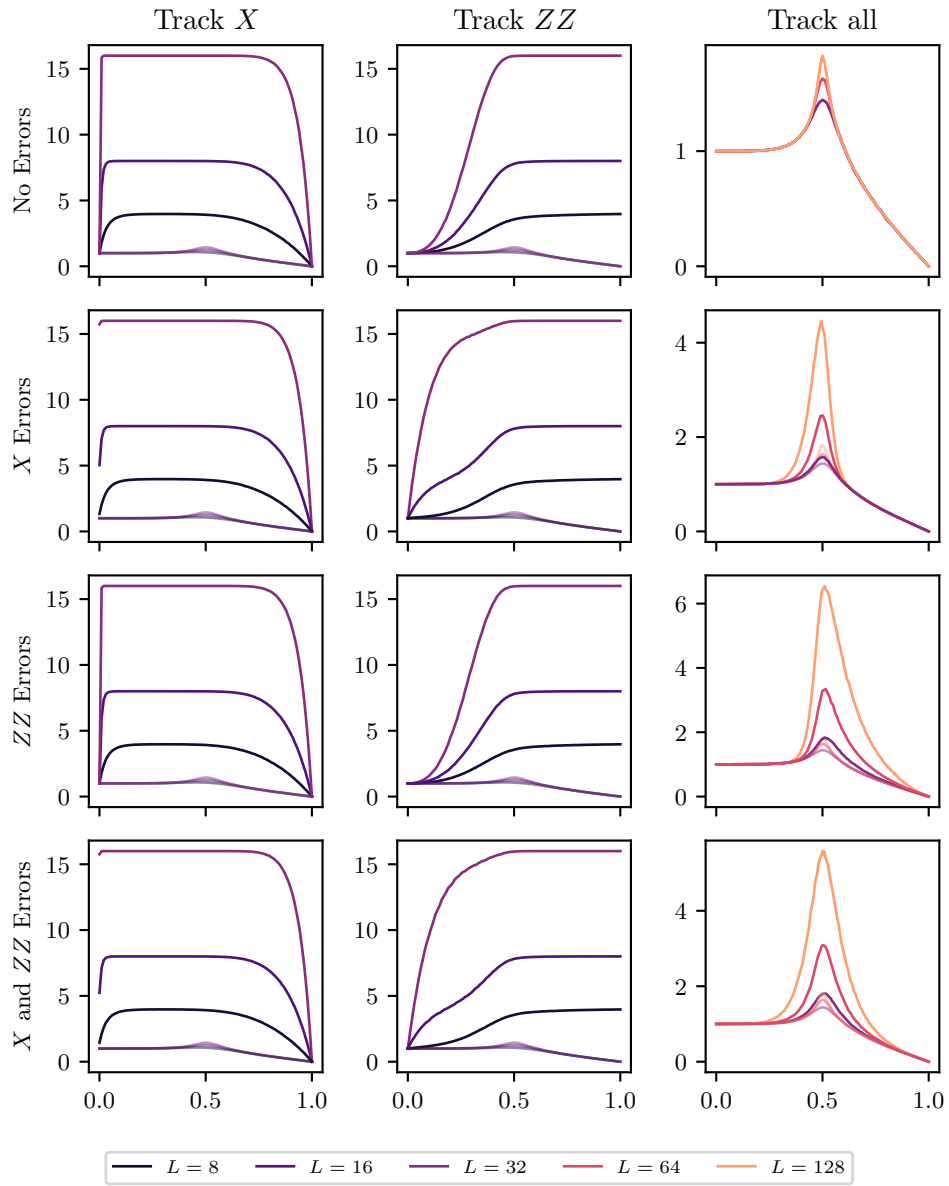


Figure 3.6.: Cross Entropy and entanglement entropy of selected system sizes with periodic boundary conditions. To be able to adequately compare it to LXE the different trackings and errors are shown in the same manner as in Figure 2.7.

shows that even with the most extreme measure of trying to ensure the adherence to the subgroup condition, we still have some samples bypassing it. This undermines the fact that clusters emerge undetected, even within the last measurement layers. Consequently, we find that even when compensating with the most extreme measures, we do not generally comply with the subgroup condition. This fact rules out any purely numerical approaches to detect the phase transition.

When considering the full density matrix, we can again qualitatively measure how good our approach was at the very end. We see that even though the "mixedness" was set to the max, we never have an extremely large value for the von Neumann entropy, as seen in Figure 3.8.

We find that for noisy systems, the algorithm did what it was designed to do, namely, intercept errors. In the presence of X errors we almost immediately recover the whole group, but fail to recover the global X stabilizer. This shows itself in the "Track all" column of Figure 3.8, where it goes to 1 for $p \to 0$. At the critical point, there is a peak for large enough systems, since there we have the highest impact of additional measurements of the competing kind. Remarkably, just as in Figure 3.5 we have "ZZ errors – Track ZZ" be identically 0. This follows from the group structure. We never maximally mix. For the "Track X" column, we just see the peak where the initial cluster dies, which was also present in Figure 3.5.

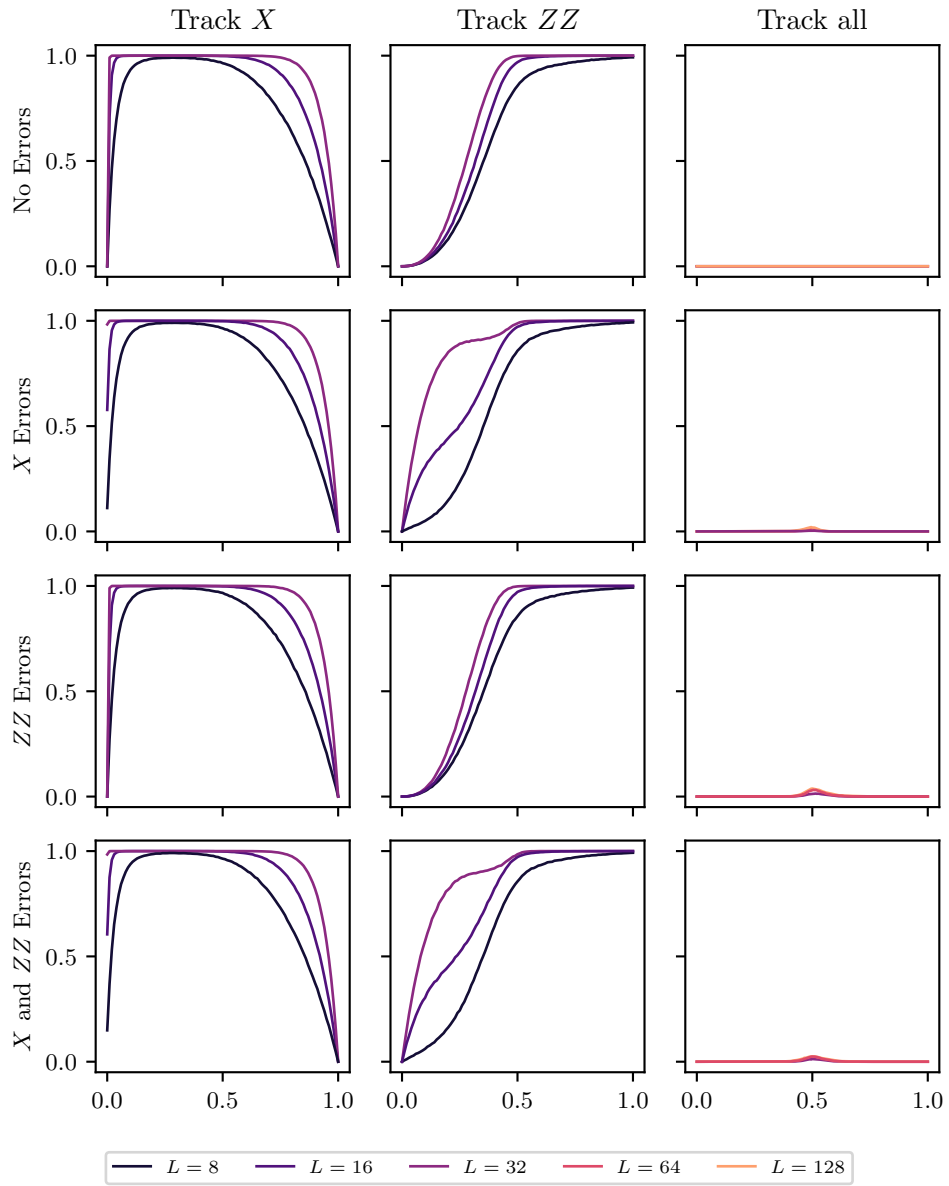


Figure 3.7.: Ratio of divergences in cross entropy to number of samples for different system sizes periodic boundary conditions. To be able to adequately compare it to LXE the different trackings and errors are shown in the same manner as in Figure 2.7.

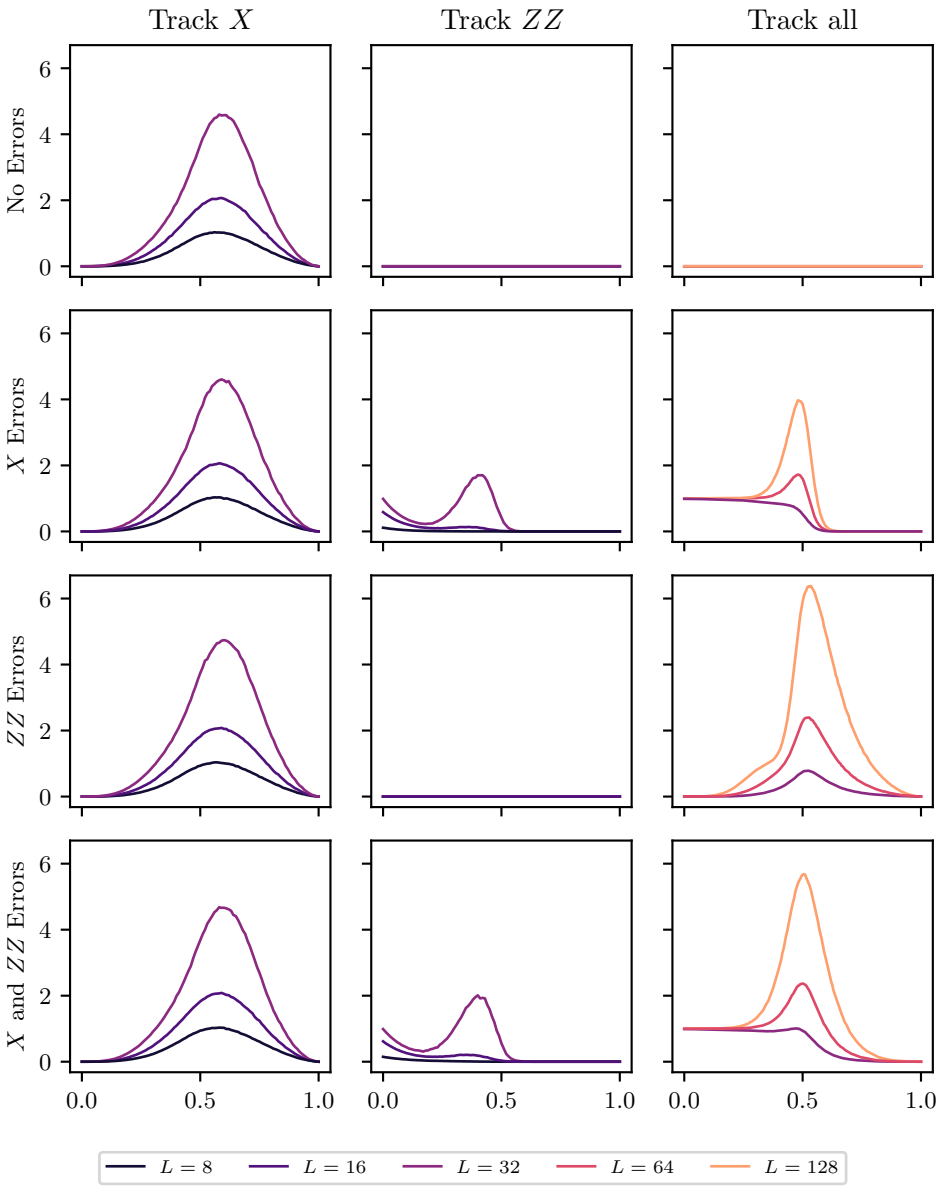


Figure 3.8.: $S(\sigma)$ of full density matrix σ periodic boundary conditions. To be able to adequately compare it to LXE the different trackings and errors are shown in the same manner as in Figure 2.7.

3.4 REGULARIZATION

We can see that, even though we chose our numerical algorithms in a way that should mitigate the infinities, we still have them. Furthermore, the subgroup check is a subtle way to incorporate once more what we try to avoid. Namely, we want to find a density matrix, which is independent of the experimental density matrix ρ. However, with the subgroup check, we implicitly introduce it into our computation again. As we once more require knowledge of the full density matrix, this would bring us right back to the sampling problem. In the following we present two approaches to regularize the infinity appearing in the cross entropy. These approaches require nothing but the numerically computed density matrix σ. The first approach is one, which could be implemented in a stabilizer simulator, which has the drawback of being exponentially worse than the trivial upper bound N, among other things. The other is more flexible, but lacks an efficient implementation in a stabilizer simulation.[6] Note that these approaches are designed to not need the subgroup condition to be fulfilled. As such, the assumption of a noiseless last measurement layer could be restricted again to include errors up to the end of measurements.

Both of these approaches could be the subject of further study, as the computational argument is one of feasibility, but not possibility (i.e., it is possible, but at what computational cost).

3.4.1 Exponential Ansatz

For the first regularization we choose an exponential Ansatz. The basic reasoning behind this Ansatz is twofold. First, we want to ensure that the support of σ is large enough to contain the support of ρ. That is, what we failed to achieve numerically, we attempt to achieve analytically. We want to ensure that the classical processing step is only a function of the measurement outcomes, $\sigma \equiv \sigma(\mathbf{m})$, and not the experimentally obtained density matrix, which we implicitly had by checking the subgroup condition on each run.

The second reason for this Ansatz in particular is the computability. Since we are performing classical computations via the stabilizer formalism, we reduce the memory requirements significantly. Storing a state of an N qubit Hilbert space is done in quadratic space complexity (see Chapter 4). If we were to write out the density matrix of, e.g., $N = 100$ qubits explicitly, we would have to deal with an exponentially large matrix. That is, if we choose σ arbitrarily, we would be faced with the herculean task of diagonalizing a $2^{100} \times 2^{100}$ matrix when computing its logarithm, way beyond the scope of any reasonable computational feasibility. By exponentiation we ensure that the logarithm "gets eaten up" such that we are dealing with the density matrix itself with an additional normalizing term.

[6] It is not definitive if there exists such an implementation or not. For now, it does not exist yet.

As mathematical expression we may write[7]

$$\tilde{\sigma} = \frac{\exp[\ln(2)\sigma]}{\mathrm{Tr}\big[\exp[\ln(2)\sigma]\big]}. \tag{3.34}$$

Substituting $\tilde{\sigma}$ back into the right hand side of Equation (3.16) we get

$$-\mathrm{Tr}\big[\rho \log \rho\big] \leq -\mathrm{Tr}\big[\rho \log \tilde{\sigma}\big] = -\mathrm{Tr}\big[\rho\sigma\big] + \log\big(\mathrm{Tr}\big[\exp[\ln(2)\sigma]\big]\big). \tag{3.35}$$

Note that the original σ is still our classical reconstruction, i.e. a density matrix in the stabilizer formalism. As σ is therefore a product of commuting projectors, and with that a projector itself, we can once again compute $-\mathrm{Tr}\big[\rho\sigma\big]$ efficiently. If $\rho = \sigma$, it recovers the purity of ρ with a negative sign, and if $\rho \perp \sigma$, then the trace is 0. In a sense, we here have recovered an inner product between density matrices, defined as

$$\langle \rho, \sigma \rangle = \mathrm{Tr}\big[\rho\sigma\big]. \tag{3.36}$$

This inner product can be efficiently computed in the stabilizer formalism, as we merely need to count projectors, and if they are orthogonal, the resulting inner product is 0. We remark that for density matrices, this inner product is bounded with $\mathrm{Tr}\big[\rho\sigma\big] \in [0, 1]$.

To efficiently compute the resulting upper bound from choosing the exponential Ansatz, we need to resolve the logarithm in Equation (3.35). To this end we proceed by first calculating the argument of the trace, which turns out to yield the result of the trace as byproduct.

Since the density matrix of interest is, up to a prefactor, a projector, we rewrite $\ln(2)\sigma$ as $c\mathbb{P}$ with some constant factor c. By the matrix exponential we have

$$\begin{aligned}
\exp[c\mathbb{P}] = \sum_{k=0}^{\infty} \frac{(c\mathbb{P})^k}{k!} &= \mathbb{1} + \sum_{k=1}^{\infty} \frac{(c\mathbb{P})^k}{k!} \\
&= \mathbb{1} + \mathbb{P} \sum_{k=1}^{\infty} \frac{c^k}{k!} && (\mathbb{P}^2 = \mathbb{P}) \\
&= \mathbb{1} + \mathbb{P} \left(\sum_{k=0}^{\infty} \frac{c^k}{k!} - 1 \right) && (\text{shift index}) \\
&= \mathbb{1} + \mathbb{P}\,(e^c - 1) \\
&= \mathbb{1} + 2^{N-n}\sigma \left(2^{2^{n-N}} - 1 \right) && (c = 2^{n-N}\ln(2) \text{ and } \sigma = 2^{n-N}\mathbb{P}_s).
\end{aligned}$$

We can then calculate the trace easily, since $\mathrm{Tr}[\mathbb{1}] = 2^N$ and $\mathrm{Tr}[\sigma] = 1$. It follows that

$$\mathrm{Tr}\big[\exp[\ln(2)\sigma]\big] = 2^N + 2^{N-n}\left(2^{2^{n-N}} - 1\right). \tag{3.37}$$

[7] The factor of ln 2 comes from the fact that we have the entropic quantities with base 2. By a leading factor of $1/\ln 2$ this factor can be omitted.

Note that the parentheses in Equation (3.37) never vanish, but only asymptotically reach 0 in the thermodynamic limit. This is also the case if one flips the sign, i.e. $\exp[-c\mathbb{P}]$, where one obtains

$$\mathrm{Tr}\big[\exp[\ln(2)\sigma]\big] = 2^N + 2^{N-n}\left(1 - 2^{-2^{n-N}}\right). \tag{3.38}$$

For both of these cases we have that the resulting upper bound is worse than trivially assuming N, with diminishing fluctuations in n,

$$\log\left(\mathrm{Tr}\big[\exp[\ln(2)\sigma]\big]\right) = \log\Big(2^N + 2^{N-n}\left(2^{2^{n-N}} - 1\right)\Big) \geq N. \tag{3.39}$$

Since this Ansatz is worse than the trivial upper bound with variations in n being small compared to the system size, we did not immediately see utility in this approach. However, all quantities in the above calculations are technically computable in the stabilizer simulator, and could thus be subject of future study.

3.4.2 Rescaling probabilities

The second regularization approach could be called the naive regularization approach. We again want the support of σ to span the entire Hilbert space. With σ being a product of projectors, it is a projector itself, up to multiplicative factors. To complete the Hilbert space, we would need to add the rest of the $2^N - 2^{N-n}$ projectors needed, and rescale them in order to put more (or less) weight on our prediction. Concretely, we want σ to have the form of

$$\tilde{\sigma} = (1-\varepsilon)\sigma + c\sum_{s\in\ker(\sigma)} \mathbb{P}_s \tag{3.40}$$

$$= (1-\varepsilon)\sigma + \frac{\varepsilon}{\mathrm{Tr}\left[\sum_{s\in\ker(\sigma)} \mathbb{P}_s\right]} \sum_{s\in\ker(\sigma)} \mathbb{P}_s, \tag{3.41}$$

such that the diagonal form of σ is

$$D_\sigma = \begin{pmatrix} 1-\varepsilon & & & & \\ & c & & & \\ & & \ddots & & \\ & & & c & \\ & & & & c \end{pmatrix}. \tag{3.42}$$

This way, every basis vector of $H^{\otimes N}$ is included in σ with a weight of either $(1-\varepsilon)$ or c. As we want to ensure that $\mathrm{Tr}[\tilde{\sigma}] = 1$, c is parametrized by ε. We now want to derive an expression for $c(\varepsilon)$, i.e. the trace in Equation (3.41), as well as a way to determine the constituents of the sum in Equation (3.40).

In principle, we need the sum in Equation (3.40) to go over all orthogonal projections in the kernel of σ. Since the support of σ is fixed by the projectors onto the eigenspaces of the tensor products of Pauli matrices contained in the generating set, we need the orthogonal complement of each of them. Suppose we have the respective signs of the generators g_i in a bitstring r_i of length n, i.e. $\mathbf{r} \in \mathbb{F}_2^n$. We then write σ as

$$\sigma = \frac{1}{2^N} \prod_{i=1}^{n} \left(\mathbb{1} + (-1)^{r_i} g_i \right). \tag{3.43}$$

The remaining sum in Equation (3.40) is then one over all the combinations of values of r_i, which are *not* the original bitstring. We might write this as

$$\sum_{s \in \ker(\sigma)} \mathbb{P}_s = \sum_{\mathbf{r}' \neq \mathbf{r}} \frac{1}{2^n} \prod_{i=1}^{n} \left(\mathbb{1} + (-1)^{r_i} g_i \right). \tag{3.44}$$

The subspaces of each of the above projectors are one-dimensional, and thus their trace is 1. Since there are 2^N total dimensions, and 2^{N-n} of them are already in the support of σ, we have a total of $2^{N-n}\,(2^n - 1)$ additional projectors.[8] By rearranging and by using the completeness relation, one can also reduce the number of projectors needed to $|\mathcal{S}_\sigma|$. However, the trace still remains the same, since we want to rescale the missing diagonal entries uniformly. This then yields

$$c = \frac{\varepsilon}{2^{N-n}\,(2^n - 1)} \tag{3.45}$$

and thus

$$\tilde{\sigma} = (1 - \varepsilon)\sigma + \frac{\varepsilon}{2^{N-n}\,(2^n - 1)} \sum_{s \in \ker(\sigma)} \mathbb{P}_s. \tag{3.46}$$

Having determined a form for the auxiliary density matrix, we want to examine it in the context of the cross entropy. It now becomes

$$-\operatorname{Tr}\left[\rho \log \sigma\right] = -\sum_n \langle n|\rho|n\rangle \log \lambda_n \tag{3.47}$$

with $\lambda_n = (1 - \varepsilon)$ or $\lambda_n = \varepsilon \left(2^{N-n}(2^n - 1)\right)^{-1}$. While this appears to be better than the diagonalization of a $2^N \times 2^N$ matrix, we still need to be aware of – or compute – the representation of ρ in the space, where σ is diagonal. Therefore, we do not quite dodge the task of handling an exponentially large matrix.

[8] One can also arrive at this through combinatorical arguments over the sign vector $\mathbf{r}$.

3.5 SUMMARY

In this chapter we investigated the utility of Klein's inequality to detect the entanglement transition in the projective transverse-field Ising model. The idea behind this approach is from Garratt and Altman in [3], where it was numerically studied with MPS simulations for Haar-random hybrid circuits. To adapt to our system, we derived the inequality in the stabilizer formalism. We then employed the derived expressions in a stabilizer simulation of the PTIM, where we investigated different numerical post-processing approaches. We then concluded the chapter by suggesting regularizations to deal with unwanted infinities.

4

CLASSICAL SIMULATION OF STABILIZER CIRCUITS

"Always programming a new type of antidote in your perimeter"

Quasimoto

In this chapter we present the simulation algorithm and the modifications to the existing infrastructure (see Ref. [70]) that were necessary for the numerical experiments of this thesis. Usually, these would fall into the *Methods* sections of the respective chapters. However, the modifications were extensive enough to warrant the dedication of an entire chapter to them. First, we will briefly introduce the simulation algorithm based on the results from Gottesman and Knill [28], and Aaronson and Gottesman [12]. Then, we briefly present the functions necessary for computing the linear cross entropy χ from Chapter 2 as this was only a small addition. We then finish off with the larger discussion of mixed states in the stabilizer formalism and the realizations of the mixed state simulation algorithms, as well as the functions for the various entropic quantities, implementing our results from Section 3.1.2.

A computer science primer

This chapter is something of an outlier compared to the other chapters. While the previous part of this work pertains to *natural* science – namely, theoretical physics – where simulations served us as a tool rather than as the primary object of study, a substantial portion of this thesis has been dedicated to expanding an existing stabilizer simulator to deal with new problems (see Sections 2.2.1, 3.1.2 and 3.3). As a result, our discussion thereof will necessarily feature terminology commonly associated with the field of *computer* science. To ensure clarity, we will briefly introduce key computer science concepts relevant to our discussion.

The first concept we introduce is asymptotic behavior, along with the so-called *Big* $\mathcal{O}$ notation. It is not exclusive to computer science as such, as we often deal with asymptotic behavior in physics as well. In computer science, the Big $\mathcal{O}$ notation is used to describe the space (i.e. data storage) and time requirements of algorithms with increasing input size. The notation is defined precisely in Definition 4.1.

Definition 4.1 (Θ and $\mathcal{O}$ notation [86])**.** Let $g(n)$ be a positive function. Then $\Theta(g(n))$ is the set of functions

$$\Theta(g(n)) = \left\{ f(n) \mid \liminf_{n\to\infty} \frac{f(n)}{g(n)} > 0 \ \wedge \ \limsup_{n\to\infty} \frac{f(n)}{g(n)} < \infty \right\}.$$

Similarly, $\mathcal{O}(g(n))$ is the set of functions

$$\mathcal{O}(g(n)) = \left\{ f(n) \mid \limsup_{n\to\infty} \frac{f(n)}{g(n)} < \infty \right\}.$$

We will later see some examples of this notation in use. For now it suffices to note that different problems and the algorithms we use to solve them (or verify the validity of a solution) are grouped into a variety of *complexity classes*. For instance, an algorithm is said to scale polynomially (in time) if its asymptotic behavior is $\mathcal{O}(n^\alpha)$ with $\alpha \geq 1$, and exponentially if it is $\mathcal{O}(2^n)$. These classes constitute an important subfield in theoretical computer science in the form of complexity theory, where numerous classes exist.[1] The two examples from above fall in the classes P (polynomial time) and EXP (exponential time) respectively. While there are lots of caveats to this highly simplified explanation, a deeper examination of complexity theory lies beyond the scope of this thesis. The key takeaway for our purposes is summarized in Definition 4.2.

[1] A comprehensive and continuously updated list can be found in [59].

Definition 4.2 (Algorithmic efficiency [59])**.** A problem is considered efficiently solvable on a classical computer if it belongs to the complexity class P.

We acknowledge that this definition is highly simplified, and specifically tailored to our purposes.[2] However, within the scope of our work, this definition suffices.

Having laid out these foundational concepts of computer science, we can now delve deeper into the subtleties of stabilizers and the problem of simulating stabilizer circuits, potentially gaining a new appreciation for their intricacies.

4.1 CONSEQUENCES OF THE GOTTESMAN-KNILL THEOREM

In this section we explore the implications of Theorem 1.14 and provide a proper introduction to the simulation algorithm that forms the foundation for the numerical experiments in this thesis. In particular, we present the *tableau algorithm*, as proposed by Aaronson and Gottesman in [12]. Consequently, this section draws largely from outside sources, namely Refs. [9, 12, 28, 83]. As the present section serves to introduce the computational algorithms based on the stabilizer formalism, it can (and should) be read as a sequel to Sections 1.1 and 1.1.3.

We begin by recalling that the stabilizer group does not need to be stored in full to unambiguously describe the state. Since the stabilizer group is finite, its structure can be fully encapsulated by storing only its *generators* in memory. This reduces the amount of data to be stored to memory from 2^n to n, owing to the well-known fact from group theory that a finite group G has a generating set of size $\log|G|$. That is, an n-qubit pure state $|\phi\rangle$ with stabilizer group $\mathcal{S}\left(|\phi\rangle\right)$ has a generating set of size $\log 2^n = n$.

To determine the actual memory requirements we examine the generators themselves. Each generator consists of an array of n Pauli matrices and a sign. Since there are four Pauli matrices (including the identity), we require two bits to encode each of them, along with an additional bit for the sign. Consequently, the memory requirements for encoding a pure state in the stabilizer formalism are $n(2n+1)$. In other words, storing only the stabilizer generators reduces the space complexity of stabilizer simulations from $\mathcal{O}\left(2^n\right)$ to $\mathcal{O}\left(n^2\right)$.

For practical purposes, these bits can be assorted to two $n \times n$ matrices and a vector containing the n signs. This way of writing the generators is called the *Tableau Representation*. As an example, consider the state $|\phi\rangle = |0000\rangle$, which is stabilized by $\mathcal{S}(|\phi\rangle) = \langle Z_1, Z_2, Z_3, Z_4\rangle$.

[2] It even omits the mention of *Turing machines* entirely.

The stabilizer tableau $\mathcal{T}$ of $|\phi\rangle$ is then given by

$$\mathcal{T}_{|\phi\rangle} = \left[\begin{array}{cccc|cccc|c} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right]. \tag{4.1}$$

Each row in Equation (4.1) represents a generator of $\mathcal{S}(|\phi\rangle)$. The first four columns are the X-matrix, the next four are the Z-matrix and the last column represents the sign. The i-th column of the X and Z matrix encode the Pauli matrix at position i in the tensor product of the corresponding generator, where $00 \equiv I$, $01 \equiv Z$, $10 \equiv X$, and $11 \equiv Y$.

At this point, we have merely stored the state in memory. However, Theorem 1.14 implies that simulations of stabilizer circuits can also be done efficiently on a classical computer. As stated in Definition 4.2, for an algorithm to be considered "efficient", it must have polynomial time complexity. Thus, the Gottesman-Knill theorem can be rephrased to state that the bits encoding $|\phi\rangle$ can be updated in polynomial time after a Clifford gate is applied to $|\phi\rangle$. Before we show this, let us first consider how the individual gates act on our generator tableau.

Algorithm 4.3 (Application of Clifford gates to stabilizer tableau)**.** The following modifications have to be applied to the stabilizer tableau after the application of

- H to qubit a:

  Swap column a of the X matrix with the a-th column of the Z-matrix.

- S to qubit a:

  Modify column a of the Z-matrix such that the new $z_{ia} = z_{ia}\texttt{XOR}x_{ia}$, i.e. apply bitwise XOR from column a of X into column a of Z.

- CNOT from control a to target b:

  Apply bitwise XOR from column a to b of X, and from column b to column a of Z.

- Random or deterministic outcome of computational basis measurement on qubit a:

  The outcome is deterministic iff. column a of the X-matrix is all 0s.

- Measurement of qubit a with a random outcome:

  Let $x_{ia} = 1$. We first apply bitwise XOR from row i into any subsequent row j, where $x_{ja} = 1$. We then set row i to 0 everywhere, except $z_{ia} = 1$ and set the sign to 0 or 1 randomly.

These rudimentary algorithms follow the rules of the stabilizer formalism, as laid out in Section 1.1. Let us now consider the performance of the algorithms in Algorithm 4.3.

Theorem 4.4 (Simulating stabilizer gates)**.** Simulating a Clifford gate on an n-qubit stabilizer state requires $\Theta(n)$ time, while a measurement gate is simulated in $O(n^2)$ or $O(n^3)$ time for random and deterministic outcomes respectively.

Proof. We already know that an n-qubit stabilizer state $|\phi\rangle$ can be represented as an $n \times (2n+1)$ tableau $\mathcal{T}$, where the rows are the n generators in the aforementioned $(2n+1)$-bit representation. Any computational basis state can thus be represented by

$$\mathcal{T} = \left[\begin{array}{cccc|cccc|c} 0 & 0 & \dots & 0 & 1 & 0 & \dots & 0 & \pm \\ 0 & 0 & \dots & 0 & 0 & 1 & \dots & 0 & \pm \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 & 0 & \dots & 1 & \pm \end{array}\right], \tag{4.2}$$

where Z and I are encoded as 01 and 00 respectively, and $\pm$ is a placeholder for either 0 or 1 depending on the sign. Simulating a Clifford gate U on $|\phi\rangle$ maps g_i, that is, the i-th row of $\mathcal{T}$, to Ug_iU. As discussed in Algorithm 4.3, this operation updates at most two columns. Therefore, simulating U takes $\Theta(n)$ time.

To show the scaling behavior of measurements we consider computational basis measurements on qubit a, Z_a, without loss of generality.[3] Each individual qubit in $|\phi\rangle$ is either in a computational basis state $|0\rangle$ or $|1\rangle$, where we have a deterministic outcome, or in a superposition of both states with equal amplitude, where either outcome is random with probability $p = 1/2$. Recall from Section 1.1 that the process of determining the type of outcome (random or deterministic) is done by checking the commutation relations between the measurement operator and the generators of $|\phi\rangle$. If some generator g_k anticommutes with Z_a, i.e.

$$\{g_k, Z_a\} = 0 \quad \text{or} \quad [g_k, Z_a] \neq 0,$$

the outcome is random, otherwise it is deterministic (this implies a scaling of $\mathcal{O}(n)$ already, since we go through the entire matrix in the worst case). We now consider these cases separately.

Case 1, random outcome: Let g_k be a stabilizer generator with $\{g_k, Z_a\} = 0$, implying a random outcome. As the outcome, we randomly choose $x \in \{0, 1\}$. We then multiply any subsequent rows that anticommute with Z_a by g_k in order for them to commute with Z_a. We then replace g_k by $\pm Z_a$ depending on the random outcome. Since this algorithm takes up to n row multiplications, the runtime scales with $\mathcal{O}(n^2)$.

[3] We can consider those without loss of generality, since we can always apply $\mathcal{O}(n)$ unitaries to change basis in $\Theta(n)$ time, which does not affect the scaling behavior of $\mathcal{O}(n^2)$.

Case 2, deterministic outcome: If we did not find an anticommuting generator, we know that our stabilizer group contains Z_a. However, this does not necessitate Z_a being in the generating set. We thus need to modify $\mathcal{T}$ such that its row space contains $(00\cdots0_a\cdots0 \mid 00\cdots1_a\cdots0 \mid \pm) = \pm Z_a$ and then read out the signs vector, which is the result. This is done by Gaussian elimination, which requires $\mathcal{O}(n^3)$ time. □

This proof, in principle, also proves Theorem 1.14, since the simulation of any of a circuit's constituents is in P. Consequently, a finite number m of them will be in $\mathcal{O}(mn^\alpha)$, and thus also in P.

4.1.1 The Aaronson-Gottesman Algorithm

In the previous section we have shown that measurements take $\mathcal{O}(n^3)$ time in practice. However, Aaronson and Gottesman showed in [12] that with the cost of a factor 2 increase in memory requirements we can improve measurements to have quadratic time complexity $\mathcal{O}\left(n^2\right)$, independent of "outcome type". In particular, for each of the n stabilizer generators we store a destabilizer (also equivalently referred to as antistabilizer) generator, which are also tensor products of Pauli operators. These $2n$ operators together generate the full n-qubit Pauli group $\mathcal{P}_n$.

The tableau idea applied in Equation (4.1) can be expanded into

$$\mathcal{T}_{|\phi\rangle} = \left[\begin{array}{ccc|ccc|c} x_{11} & \cdots & x_{1n} & z_{11} & \cdots & z_{1n} & r_1 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{n1} & \cdots & x_{nn} & z_{n1} & \cdots & z_{nn} & r_n \\ \hline x_{(n+1)1} & \cdots & x_{(n+1)n} & z_{(n+1)1} & \cdots & z_{(n+1)n} & r_{n+1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{(2n)1} & \cdots & x_{(2n)n} & z_{(2n)1} & \cdots & z_{(2n)n} & r_{2n} \end{array}\right] \tag{4.3}$$

for an arbitrary stabilizer state $|\phi\rangle$. The first n rows represent the newly introduced destabilizer states, while the last n rows constitute the tableau of stabilizers we have already discussed. For later convenience, an additional $(2n+1)$st row is added to the tableau. A minimal example of the generalized tableau is the state $|00\rangle$, which has

$$\mathcal{T}_{|00\rangle} = \left[\begin{array}{cc|cc|c} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \end{array}\right]. \tag{4.4}$$

A quick remark on notation: we will refer to the i-th row of $\mathcal{T}$ by R_i, where it is clear from the respective context if the row refers to a stabilizer or destabilizer generator. If we

explicitly refer to one of them, we write g_i for the i-th stabilizer generator and h_i for the i-th antistabilizer generator. Matrix elements of the X and Z matrices are denoted by x_{ab} and z_{ab} respectively. Entries of the sign vector are denoted by r_i (see Equation (4.3)).

Although it is simply stated, it is not immediately obvious how (a) we choose destabilizer generators, and (b) how this improves the scaling of measurements. For now we say that the simulation algorithm starts in the state $|0\rangle^{\otimes n}$, where the initial tableau is taken as the n-qubit generalization of Equation (4.4). Any other stabilizer state can then be arrived at via Clifford or Pauli measurement gates. The former can be implemented as follows.

Algorithm 4.5 (Improved simulation of Clifford gates)**.** Let $\oplus$ denote bitwise `XOR`. The implementations are

- CNOT from control a to target b:

  $\forall i \in \{1, \ldots, 2n\}$ $r_i := r_i \oplus x_{ia}z_{ib}\,(x_{ib} \oplus z_{ia} \oplus 1)$, $x_{ib} := x_{ib} \oplus x_{ia}$, and $z_{ia} := z_{ia} \oplus z_{ib}$.

- H on qubit a:

  $\forall i \in \{1, \ldots, 2n\}$ $r_i := r_i \oplus x_{ia}z_{ia}$, then swap x_{ia} and z_{ia}.

- S on qubit a:

  $\forall i \in \{1, \ldots, 2n\}$ $r_i := r_i \oplus x_{ia}z_{ia}$, then set z_{ia} to $z_{ia} \oplus x_{ia}$.

We know from earlier considerations that simulating measurement gates, requires the multiplication of generators. To this end, we introduce a subroutine called `rowsum(h,i)`. It updates the h-th generator to be $h + i$ and keeps track of the phase r_h. Its implementation is given in Algorithm 4.6.

Algorithm 4.6 (rowsum)**.** First we define a function $g(x_1, z_1, x_2, z_2)$ taking 4 bits as input and outputting the exponent of the imaginary unit i after $x_1 z_1$ and $x_2 z_2$ are multiplied. We focus on four cases explicitly, namely

x_1	z_1	$g(x_1, z_1, x_2, z_2)$
0	0	0
0	1	$x_2\,(1 - 2z_2)$
1	0	$z_2\,(2x_2 - 1)$
1	1	$z_2 - x_2$

For the rowsum routine we set r_h to 0 if

$$2r_h + 2r_i + \sum_{j=1}^{n} g\left(x_{ij}, z_{ij}, x_{hj}, z_{hj}\right) \equiv 0 \,(\mathrm{mod}\, 4). \tag{4.5}$$

or set r_h to 1 if the sum in Equation (4.5) is congruent to 2 mod 4. Next, we set x_{hj} to $x_{ij} \oplus x_{hj}$ and z_{hj} to $z_{ij} \oplus z_{hj}$ for all j, where $\oplus$ denotes bitwise XOR.

Next it will prove useful to define the *symplectic inner product* between two Pauli operators A and B in tableau representation as

$$A \cdot B = x_{a1} z_{b1} \oplus \cdots \oplus x_{an} z_{bn} \oplus x_{b1} z_{a1} \oplus \cdots \oplus x_{bn} z_{an}. \tag{4.6}$$

This inner product is 0 if A and B commute and 1 if they anticommute.

Equipped with rowsum and the symplectic inner product, we can now examine the simulation of measurement gates.

Algorithm 4.7 (Improved simulation of measurement gates)**.** Suppose we measure $\hat{O}$. As a 0-th step, check if there exists a $p \in \{n+1, \ldots, 2n\}$ (the stabilizer generators) such that $[\hat{O}, R_p] \neq 0$. This can be done by multiplication with respect to Equation (4.6). Then there are two cases:
Case 1, such a p exists. First, call rowsum(i, p) for all $i \in \{1, \ldots, 2n\}$ such that $i \neq p$ and $[\hat{O}, R_i] \neq 0$. Next, set the $(p-n)$-th row equal to the p-th row. Then, set row p equal to $\hat{O}$ and set r_p to 0 or 1 with equal probability. Finally, return r_p as measurement outcome.
Case 2, such a p does not exist. First, set the entire $(2n+1)$st row (the one added for convenience earlier) to 0. Next, call rowsum$(2n+1, i+n)$ for all $i \in \{1, \ldots, n\}$ (the destabilizer generators) such that $[\hat{O}, R_i] \neq 0$. Finally, return r_{2n+1} as measurement outcome.

With Algorithms 4.5 and 4.6, all possible allowed modifications to the stabilizer tableau $\mathcal{T}$ are defined. Proposition 4.8 collects some symmetries of this simulator. That is, some commutation relations are invariant under operations of the Aaronson-Gottesman tableau algorithm. It is these relations we want to keep intact, when setting out to expand the existing simulator with more functionalities.

Proposition 4.8 (Invariants of the tableau algorithm)**.** The following are invariant under operations of the tableau algorithm:

1. $R_{n+1}, \dots, R_{2n}$ generate $\mathcal{S}(|\phi\rangle)$, and $R_1, \dots, R_{2n}$ generate $\mathcal{P}_n$.
2. $R_1, \dots, R_n$ commute.
3. $\forall h \in \{1, \dots, n\},\ \{R_h, R_{h+n}\} = 0$
4. $\forall i, h \in \{1, \dots, n\},$ with $i \neq h,\ [R_i, R_{h+n}] = 0$

We will conclude this section with a note of caution. The simulator we set out to expand has, in contrast to how it was defined in this section, the stabilizers in even-numbered rows of the tableau, and each associated antistabilizers in the row below it. Although it is merely a change of indices, it is a major one, and for didactic reasons we chose to forego this change. Since it only matters which antistabilizer is associated with each stabilizer, we try to limit the mention of any explicit ordering of generators in the tableau.

4.2 NEW FUNCTIONS ON PURE STATES

The first function implemented is the *project* function. It is used in the algorithm for computing the linear cross entropy, as successful projections in the circuit yield $\chi = 1$ and $\chi = 0$ otherwise. Luckily for us, projections are already a part of the existing simulator in the form of projective measurements. The important difference between the measurement algorithm and projections is that we already know the measurement outcome as an argument of our function. That is, we take the usual measurement algorithm, but add a new argument for the measurement result, which is projected onto. As such, the projection always works for a random result, since one of the steps in the measurement was to flip a coin for the sign, which is now a fixed value. However, for the deterministic case we are not as agnostic. Since we try to project onto a state, which is already in the stabilizer, the signs of what we have and what we pass as argument should match. If they do not, the function returns `false`. This is then used as break condition when computing the LXE. The project algorithm is outlined in Algorithm 4.9.

Algorithm 4.9 (Projection onto Pauli eigenstates)**.** Suppose we want to apply $P = \frac{I+\hat{O}}{2}$. As step 0, we again need to check if projection changes the state, by checking if there exists a stabilizer generator R_p such that $[\hat{O}, R_p] \neq 0$. Then there are two cases:
Case 1, projection modifies the state. We repeat the steps from case 1 of Algorithm 4.7, but instead of randomly choosing the sign, we use the one we want to project onto. Finally, we return `true` since the projection was successful.
Case 2, projection does not modify the state. We similarly repeat the steps from case 2 of Algorithm 4.7, but return `true` or `false` depending on if the sign we pass as function argument matches r_{2n+1} or not.

An important point to note is that this function does not resemble anything we could do in an experiment. There is no experimental apparatus conceivable to perform the operation we are simulating. It is possible only because we perform a classical simulation and know mathematically what a projection operation does. This kind of degree of freedom is noteworthy, and we should keep it in mind going forward.

Forced projection

As we just discussed, the projection can fail and return `false`. This is rather unhelpful, since this would terminate the simulation. In Section 3.2, this behavior is undesirable. Consequently, we need to add a function, which forces a specific projection by altering the structure of the tableau.

The only way we know if a projection fails is by computing the outcome of the deterministic measurement and checking if the value is equal. This process involves reconstructing the measurement operator with the help of anticommuting antistabilizers (see Algorithm 4.7). If the result is not equal, we do not have the generator with the correct sign in the stabilizer group. As a consequence, we need to alter the tableau in a certain way. To this end, we employ the method of case 1 in Algorithm 4.7 and place the correct sign. We thus pretend to be able to project successfully. This algorithm is summarized in Algorithm 4.10.

Algorithm 4.10 (Forced projection)**.** The algorithm works the same way as Algorithm 4.9, but instead of returning `false` upon failed projection, we manipulate the tableau analogously to case 1 of Algorithm 4.7 and insert the correct sign.

Note that both Algorithms 4.9 and 4.10 utilize the pre-existing subroutines for measurement gates from the improved simulation algorithm. Consequently, they are also $\mathcal{O}\left(n^2\right)$, with the latter being slightly worse than the former, but equal asymptotic behavior.

4.3 IMPLEMENTING MIXED STATES

The tableau algorithm in the form it is outlined in Section 4.1.1 currently supports stabilizer circuit simulations exclusively for pure states. This, however, neglects a more general class of quantum states, namely, mixed states. It has become apparent in Chapter 3 why this adaption is necessary for our purposes. Ideally, we want to build on the existing infrastructure we have been using for pure states and extend it such that pure states arise as a special case within the simulator. Let us approach this problem heuristically by thinking about an intuitive approach to incorporate mixed states to the simulator.

Consider the density matrix of a general N-qubit stabilizer state with generating set $\mathcal{S}(\rho) = \langle g_1, \dots, g_n \rangle$,

$$\rho = \frac{1}{2^N} \prod_{i=1}^{n} I + g_i. \tag{4.7}$$

Here, ρ describes a pure state iff. $n = N$ and a mixed state otherwise. That is, by reducing the number of generators, we increase the state's mixedness, with the maximally mixed state represented by the trivial group, generated by the empty set.

What about the action of unitary transformations, i.e. Clifford gates, on our state? Since applying unitaries to density matrices works by conjugation with U and $U^\dagger$, their application remains unchanged from the pure-state case. Measurements, however, introduce a new challenge: One can show that when measuring any Pauli operator $\hat{O}$ on a qubit in a maximally mixed state, the outcomes will be random and their probabilities uniformly distributed. However, this contrasts the previous instances where measurement outcomes were random, since there are no anticommuting generators of ρ with $\hat{O}$. This is certainly something to take note of in modifying the existing measurement algorithm.

With mixed states come new possibilities for quantities we could query on our system. For instance, the Von Neumann entropy (see Definition 3.2) of a pure state is always 0. It is only non-zero for a mixed state. Furthermore, the cross and relative entropy only really get particularily interesting when considering mixed states, as we have discussed in detail in Section 3.1.2. We should therefore include the ability to access these entropic quantities, and also compare two states to another, as is done for the cross and relative entropy.

One central question remains left unanswered: how *do* we obtain mixed states within this framework? That is, if the previous program pertained to pure states only, what would be a natural way to introduce mixedness? One approach could certainly be to start with a pure state of a larger system and then trace out entangled qubits. So, a natural inclusion to our algorithm would be a partial trace function. Alternatively, we can recognize once more that we are not bound by the limits of nature; similarly to the project function, which has no natural or experimental pendant, we can artificially introduce mixedness by selectively removing stabilizers. For instance, removing a generator that acts only on one qubit (which

is not coupled to other qubits) should yield a mixed state for that qubit. This allows outside control of mixedness within the simulation.

Let us briefly summarize this train of thought:

Minimal reworking: The current algorithm (see Section 4.1.1) should be extended with minimal changes. Any previously written simulation based on it may not break.

Generator count: An N-qubit pure state has N stabilizer generators, while a mixed state has $0 \leq n < N$ generators.

Unitaries: The application of unitaries is agnostic to mixedness. They should work *exactly* the same way they have before.

Measurements: Measurements introduce a new contingency for random results. The existing measurement function should be made to be able to handle it.

Entropy: Mixed states allow for a broader spectrum of entropic quantities to be computed.

Partial trace: There should be a function, which implements the partial trace over (at least) one qubit.

Classical advantages: Classical simulation allows us to construct artificial ways of introducing mixedness.

In the following sections we will go over the algorithms and their implementations of each of the above points in detail.

On "minimal reworking"

The point about "minimal reworking" is somewhat unrelated to physics, but rather computer science (once again). Going forward, our working philosophy should, of course, be the faithful implementation of physical phenomena to classical computers. However, we also try to follow common practices and principles of software development. In particular, we aim for as little repetition as necessary, by the DRY (don't repeat yourself) principle, and introduce new subroutines only if deemed unavoidable. If there exists a function that does what we want, we will use it, as we have already done for Algorithms 4.9 and 4.10.

4.3.1 The mix attribute

For the first point[4] – the generator count –, we introduce the *mix* attribute to the simulator. In the existing implementation, each multi-qubit stabilizer system is an instance of

[4] Considering "Minimal reworking" is our working philosophy, the thing to keep in mind, we count it as 0-th point.

the `Qubits` class. Its member functions are the possible actions on a stabilizer state. By introducing the `mix` attribute, which is an integer $0 \leq mix \leq N$, we keep track of how many stabilizers contribute to the description of the physical state. The technical use of this attribute is to control which rows of the tableau correspond to those generators which describe the physical state, and which rows are generators completing the tableau. To clarify, if we were to take the stabilizer tableau of an arbitrary pure state, its `mix` value will be 0. Incrementing this value by 1 increases the mixedness and constraints the rows of the tableau we interpret as the stabilizer generators. Consider the example of the two-qubit bell state $|\phi\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$ with generating set $\langle ZZ, XX\rangle$. The corresponding stabilizer tableau (omitting the scratch row) is

$$\mathcal{T}_{|\phi\rangle} = \left[\begin{array}{cc|cc|c} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{array}\right]. \tag{4.8}$$

If we now increment `mix` by 1, the last (anti)stabilizer row is still carried around in the simulation, but does not correspond to any stabilizer of the state. For the above state, incrementing `mix` reduces the generating set to $\langle ZZ\rangle$, which corresponds to a mixed state with density matrix

$$\rho = \frac{1}{4}(I + ZZ) = \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.9}$$

The corresponding bit matrix in the tableau representation then reads

$$\mathcal{T}_{\rho} = \left[\begin{array}{cc|cc|c} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{array}\right] \tag{4.10}$$

where the dashed lines correspond to the increment of `mix`. Note that we, in principle, have all the freedoms laid out in Proposition 4.8 to choose the generators below the dashed line as it is only needed for mathematical and technical convenience. We will see the advantages of keeping tabs on the full-rank tableau in the following sections.

4.3.2 Unitaries and measurements

Now that we have introduced a method to describe a mixed state to the simulator, let us further investigate how its member functions need to be adapted in order to faithfully

simulate the behavior of mixed states. Luckily for us, adapting unitary transformations is a rather simple task, since it works the same as for mixed states. It thus requires no further inquiry.

The same cannot be said about measurements. As introductory example, consider the mixed state described by the density matrix in Equation (4.9). Suppose we now perform a Z measurement on the first qubit. The fact that the measurement operator commutes with every generator[5] might lead us to believe that the measurement outcome is deterministic. However, we also have

$$\begin{aligned} p(m_Z = +1) &= \mathrm{Tr}\left[\frac{I + Z_1}{2}\frac{1}{4}(I + ZZ)\right] \\ &= \mathrm{Tr}\left[\frac{1}{8}(I + Z_1Z_2 + Z_1 + Z_2)\right] \\ &= \frac{1}{8}\mathrm{Tr}[I] = \frac{1}{2}. \end{aligned} \tag{4.11}$$

This seems to contradict our previous method of determining the *type* of measurement outcome. As a consequence, we need to include this new contingency into the existing measurement function. At this point, recall that we still keep the whole tableau of generators, as if we had a pure state, and just track how many are not descriptors of the physical state. Therein also lies the solution to the problem; we can intercept the case for random outcomes in a mixed state by checking the tableau below the mix line as well. Note that we also include the antistabilizers in this case. The commutation relations laid out in Proposition 4.8 ensure the success of this algorithm. The augmentations of Algorithm 4.7 are summarized in Algorithm 4.11.

Algorithm 4.11 (Simulation of measurement gates on mixed states)**.** Suppose we measure $\hat{O}$. As a 0th step, check if there exists a $p \in \{n+1, \dots, 2(n-mix)\}$ (generators describing the state) such that $[\hat{O}, R_p] \neq 0$. Then there are three cases:
Case 1, such a p exists. This case is the same as case 1 of Algorithm 4.7.
Case 2, such a p does not exist. First, check analogously to step 0 if there exists a $p \in \{n - mix, \dots, n, 2(n - mix), \dots, 2n\}$ such that $[\hat{O}, R_p] \neq 0$. If such a p does not exist, this case reduces to case 2 of Algorithm 4.7. If it exists, we have the new third case.
Case 3, there exists a $p \in \{n - mix, \dots, n, 2(n - mix), \dots, 2n\}$ such that $[\hat{O}, R_p] \neq 0$. We perform the steps from case 1 of Algorithm 4.7 with this p. Then we swap rows such that R_p is included in the topmost row of the `mix` stabilizers. (Depending on p, this might include a swap of an antistabilizer to the stabilizers.) Then, we decrement `mix` by 1.

[5] ZZ in this simple example

Note that this additional case does not worsen the time-complexity of the simulation of measurement gates, in theory as well as in practice. The only addition to the existing algorithm is another loop over more rows. The total number of rows, however, is limited to $2N$ for N qubits, which does not worsen the performance or change the asymptotic behavior.

4.3.3 Entropies

By taking a broader class of quantum states into our consideration, we unlock a broader set of entropic quantities to be investigated and computed. We will here introduce the functions that compute the von Neumann, cross and relative entropy.

Von Neumann entropy

The first new function the `mix` attribute allows us to realize is the von Neumann entropy. We know from Section 3.1.2 that the von Neumann entropy has a simple expression for stabilizer states, namely

$$S(\rho) = -\operatorname{Tr}[\rho \log \rho] = N - |\mathcal{S}_\rho| \tag{4.12}$$

with the number of qubits N and the rank of the stabilizer group $|\mathcal{S}_\rho|$. Since the rank of a group is the size of its smallest generating set, we already have the von Neumann entropy baked into our extension. By definition, `mix` is the number of generators the state has fewer than a pure state, which has N generators. Thus, the von Neumann entropy is just `mix`. Remarkably, we therefore have $\mathcal{O}(1)$ scaling for the von Neumann entropy, since it is a lookup of a constant. Due to its simplicity we will forego a detailed summary of the function here. Nonetheless, the implementation of the von Neumann entropy in the simulator (written in `C++`) can be found in Appendix A.

Cross and relative entropy

We want to include a way to compute the cross and relative entropy between two stabilizer states in our simulator. However, this requires an auxiliary subroutine, which will be introduced beforehand.

`is_subgroup_of`

We know from Theorem 3.8 that the relative or cross entropy $\operatorname{Tr}[\rho \log \sigma]$ only takes on finite values if and only if the stabilizer group of σ, $\mathcal{S}(\sigma)$, is a subgroup of the stabilizer group of ρ, $\mathcal{S}(\rho)$. As such, before continuing with the implementation of other entropic quantities, it seems a worthwhile endeavor to introduce a subroutine that verifies if one stabilizer group is a subgroup of the other. To simplify notation, we fix the `Qubit` objects such that we ask if σ is a subgroup of ρ.

The function `is_subgroup_of` should, as its name implies, return a boolean; `true` if σ is a subgroup of ρ, and `false` if it is not. Its implementation could, for instance, be as a member function of the Qubit class, which takes as input a reference to an instance of another Qubit object. This way, the function is still within the scope of the class, and already has access to the tableau and consequently the generators.

Answering the question if one stabilizer group is the subgroup of another is equivalent to answering the question if we can generate the smaller group from group elements of the larger one. That is, if there is no generating set of $\mathcal{S}(\rho)$ that *does not* also generate $\mathcal{S}(\sigma)$, we know that $\mathcal{S}(\rho) \not\leq \mathcal{S}(\sigma)$. Put differently, if there is an element in the stabilizer of σ that cannot be constructed with ρ stabilizers by means of the group operation, the condition fails. It follows that we need a way to determine if the generating set of σ is contained in ρ.

One advantage of the tableau representation is that, while it is not a proper "representation" in the strictest mathematical sense[6], it does feature a way to represent the group operation, namely *rowsum*. The question to answer can then be abstracted to the following. Given two stabilizer tableaus, corresponding to σ and ρ, respectively, can the tableau of σ be transformed to contain only 0s using only rowsum(s, r), where r and s are rows of $\mathcal{T}_\rho$ and $\mathcal{T}_\sigma$, respectively?

Note that this question pertains to matrices and row manipulation thereof. What we arrived at through this chain of arguments is Gaussian elimination of a combined stabilizer tableau. If we write the stabilizer tableaus of ρ and σ above one another and then perform Gaussian elimination on the resulting tableau, we can deduce the subgroup property by checking if the rows corresponding to σ have non-zero entries. If they do, there is a generator of σ we could not construct from generators of ρ. Consequently, the function should return `false` in these cases and `true` for all zeros. The algorithm of this function is summarized in Algorithm 4.12.

Algorithm 4.12 (Is subgroup of)**.** Let $\mathcal{T}_\sigma$ and $\mathcal{T}_\rho$ be tableaus of (possibly mixed) N-qubit stabilizer states σ and ρ, respectively. W.l.o.g we say that we want to determine if $\mathcal{S}(\sigma) \leq \mathcal{S}(\rho)$.
First, construct a new auxiliary matrix M as follows. Rows $1, \ldots, N -$ `rho.mix` of M correspond to the "non-mix" stabilizer generators in $\mathcal{T}_\rho$. Rows $(N -$ `rho.mix` $+ 1), \ldots, (2N -$ `rho.mix` $-$ `sigma.mix`$)$ correspond to the analogous rows of $\mathcal{T}_\sigma$.
Next, perform a standard Gaussian elimination algorithm on M. If, at any point during the algorithm, the pivot is found in a row corresponding to $\mathcal{T}_\sigma$, return `false`. Finally, if the elimination algorithm finished without returning `false`, the pivot was never found in a row corresponding to the σ tableau. Thus, these rows are all 0s, and the function returns `true`.

[6] In representation theory, the group elements are mapped onto GL(V) and the group operation becomes matrix multiplication.

As this is a Gaussian elimination algorithm, which scales as $\mathcal{O}(n^3)$ in practice, Algorithm 4.12 scales as such. Recall that this is also what Fattal et al. had for their algorithm of the *entanglement entropy*.

Cross and relative entropy, ctd.

With the subgroup check in place, we can tend to the computation of the other entropic quantities. If the subgroup condition does *not* hold, the functions for the cross and relative entropy should return ∞. With finite computational resources, representing an infinite value is rather impossible. Thus, in the case where the subgroup check fails, these functions return `quiet_nan`s.

In the other case, we employ the results from Section 3.1.2, namely Theorem 3.10 and Corollary 3.11 for the cross and relative entropy, respectively. The cross entropy between ρ and σ then simply becomes the von Neumann entropy of σ, while the relative entropy becomes the difference between `sigma.mix` and `rho.mix`.

In conjunction with Algorithm 4.12 we find agreement with the $\mathcal{O}\left(n^3\right)$ scaling of the entanglement entropy, found by Fattal et al. in Ref. [27].

4.3.4 Partial trace

The next item on the list is a function that implements the partial tracing over a subsystem. The function itself is rather short, but relies on two auxiliary functions, which hide the work required. We first introduce the subroutines to then combine it into the full partial trace function.

`get_state_type`

The first auxiliary subroutine needed for the partial trace algorithm we call `get_state_type`. It takes a qubit position a as input and outputs the number of unique stabilizer generators minus one on that qubit. The name of the function stems from the fact that we can have three different state types: entangled, product and mixed. A qubit with two unique stabilizer generators, i.e. $g_{ia} = Z$ and $g_{ja} = X$ with $i \neq j$, will be in an entangled state with another qubit $b \neq a$. If there is only one unique stabilizer generator, we have qubit a in a product state, where the state is the state stabilized by the generator. Finally, no stabilizers correspond to a mixed state, since the empty set generates the trivial group, which corresponds to a mixed state in the stabilizer formalism.

The algorithm works by checking the q-th column for each stabilizer in the tableau. This is then decoded the same way we encoded the Pauli matrices in the tableau algorithm ($00 \equiv I$, $01 \equiv Z$, $10 \equiv X$, $11 \equiv Y$) and (in case of a non-zero value) stored into a variable `dummy`. If we have two differing non-zero values for our Pauli encoding, we know our qubit to be in an entangled state with at least one other qubit, and we return 1. If there are no other generators, we return 0 and if `dummy` is 0, qubit a is in a mixed state. This algorithm is formally written

out in Algorithm 4.13. Since this is a novel subroutine, which has no counterpart in the previous simulator, we have included a flowchart representation of the algorithm, shown in Figure 4.1.

Algorithm 4.13 (Determine state type)**.** Let a be the qubit we want to determine the "state type" of. First, set a dummy variable to 0. Then, loop over non-mix stabilizers and compute $2x_{ia} + z_{ia}$. If this quantity is non-zero, set the dummy variable equal to it, then continue looping if necessary. If $2x_{ia} + z_{ia}$ is non-zero again, and not equal to `dummy`, return 1. If the loop ends and `dummy` is non-zero, return 0. If the loop breaks with `dummy` $= 0$, return -1.

`rowreduce`

The next subroutine we expand the simulator with is `rowreduce`, which is also vital to the `ptrace` algorithm. Remember that the tableau algorithm is based on the stabilizer generators and we already know that adding two rows together, i.e. multiplying two generators, leaves the commutation relations invariant. This means that we can perform row reduction to row echelon form on our tableau without effect on the described state.

However, some subtleties need to be taken into account. In principle it is possible to row reduce the entire tableau. But it turns out that we need only to reduce the columns associated with one particular qubit, when employed as subroutine to `ptrace`. Next, we need to pay attention to the fact that our stabilizer tableau has dimensions $n \times 2n$. Ideally, `rowreduce` should modify our stabilizers in a way that there are at most one of X or Z stabilizers for our qubit. A natural first step would then be to treat the respective X and Z column separately. This is where one needs to be careful. Although the X and Z stabilizers are in separate columns, they share the rows, e.g. in the case where $X_{ia} = Z_{ia} = 1 \equiv Y_{ia}$, meaning that a reduction of X will influence the Z column and vice versa. One way to reconcile this is to reduce the X stabilizers first, swapping the row containing X to the bottom if necessary, then doing Z stabilizers. That way we ensure that we do not introduce X stabilizers when adding rows together to get rid of Z stabilizers. This already hints to the next subtlety we need to take into account.

A priori, one would probably perform Gaussian elimination to obtain an upper triangular form. The algorithm thereof is widely studied and the plight of many computer science first year students. It would thus be natural to assume that we want to have the reduced rows as first rows of the matrix. Nevertheless, our case is different; since we exclusively call `rowreduce` to trace out a qubit, it will later prove convenient to have the reduced rows on the *bottom* of the tableau. This way, we can later simply set `N=N-1;`. We refer to the later discussion of the partial trace function for a more in-depth explanation why this is done.

The last subtlety we want to highlight is the fact that with each modification of the

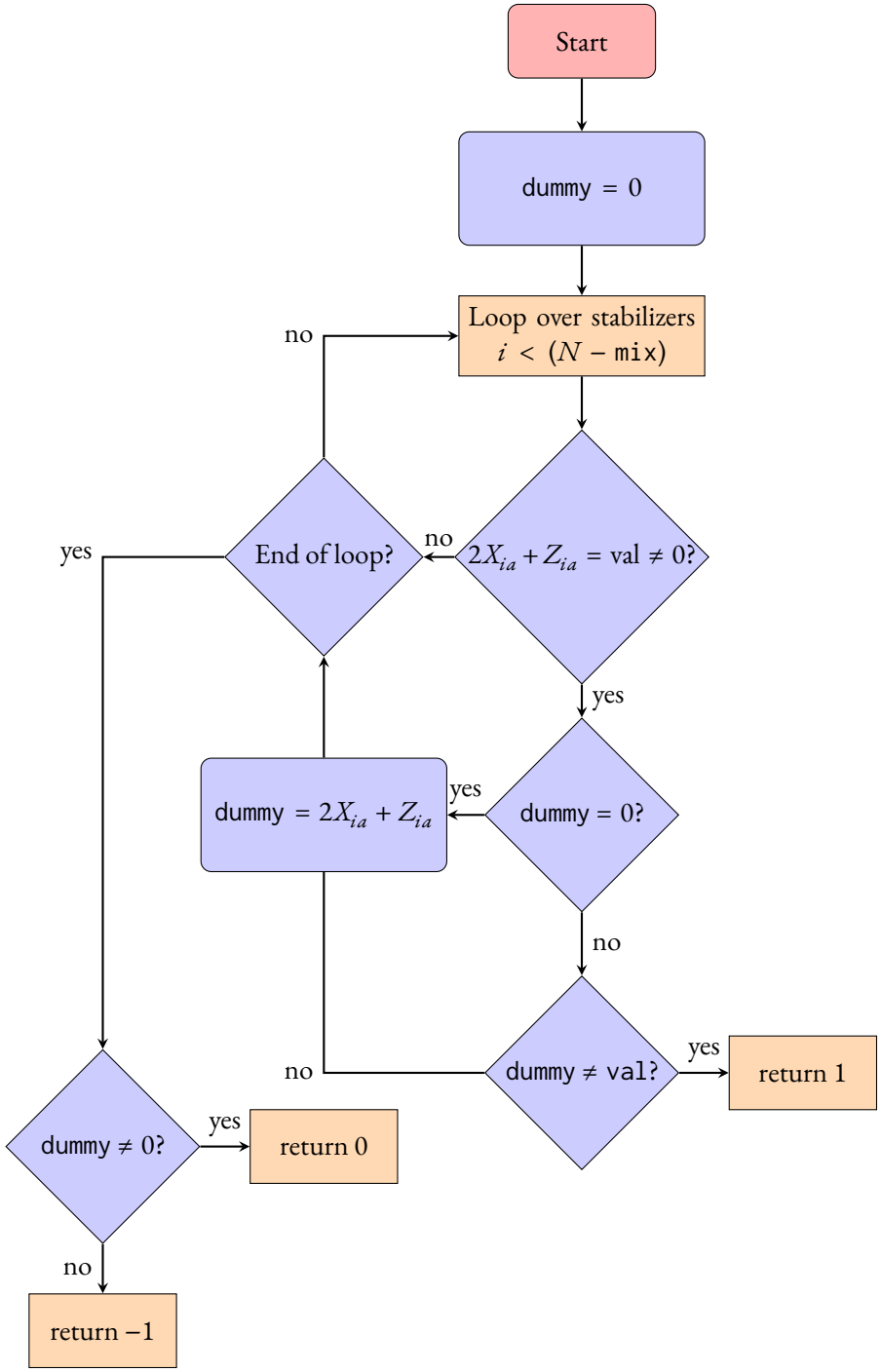


Figure 4.1.: Flowchart representation of the `get_state_type` subroutine for qubit a. Note that the function terminates upon returning one of the possible values.

stabilizer generators, we need an appropriate modification of the *anti*stabilizers to keep the commutation relations of Proposition 4.8 intact. Although we do not technially need to modify the antistabilizers for our purposes, it is still necessary if we want to continue applying the tableau algorithm on the rowreduced tableau. To this end, see Proposition 4.14, where this statement is formalized and proven.

Proposition 4.14. Let $\mathcal{T}$ be a tableau with stabilizer and antistabilizer generators $S = \langle g_1, \dots, g_n \rangle$ and $A = \langle h_1, \dots, h_n \rangle$ respectively, where the generators fulfil the commutation relations of Proposition 4.8. Replacing g_j by $g_i g_j$ in the stabilizer generators leaves Proposition 4.8 invariant if h_i is replaced by $h_i h_j$ in the antistabilizer generators.

Proof of Proposition 4.14. We will first recall the invariants as given in Proposition 4.8. They read

1. $R_{n+1}, \dots, R_{2n}$ generate $\mathcal{S}(|\phi\rangle)$, and $R_1, \dots, R_{2n}$ generate $\mathcal{P}_n$.
2. $R_1, \dots, R_n$ commute.
3. $\forall b \in \{1, \dots, n\},\ \{R_b, R_{b+n}\} = 0$
4. $\forall i, b \in \{1, \dots, n\},$ with $i \neq b,\ [R_i, R_{b+n}] = 0$

We prove the statement by showing that each of the above points still hold for the proposed modifications.

1. Since we merely multiplied generators, this holds by group theoretic arguments
2. All of the antistabilizers did commute previously, therefore, their product commutes as well
3. There are only two relations of relevance for this point

   $$\{h_i h_j, g_i\} \stackrel{!}{=} 0 \quad \text{and} \quad \{h_j, g_i g_j\} \stackrel{!}{=} 0,$$

   since all the other generators are left as they were. To show this anticommutation relation we employ the well-known identity

   $$\{AB, C\} = A[B, C] + \{A, C\}B$$

   to obtain

   $$\{h_i h_j, g_i\} = h_i \underbrace{[h_j, g_i]}_{=0} + \underbrace{\{h_i, g_i\}}_{=0} h_j = 0 \qquad \text{and}$$
   $$\{h_j, g_i g_j\} = \{g_i g_j, h_j\} = g_i \underbrace{[g_j, h_i]}_{=0} + \underbrace{\{g_i, h_i\}}_{=0} g_j = 0.$$

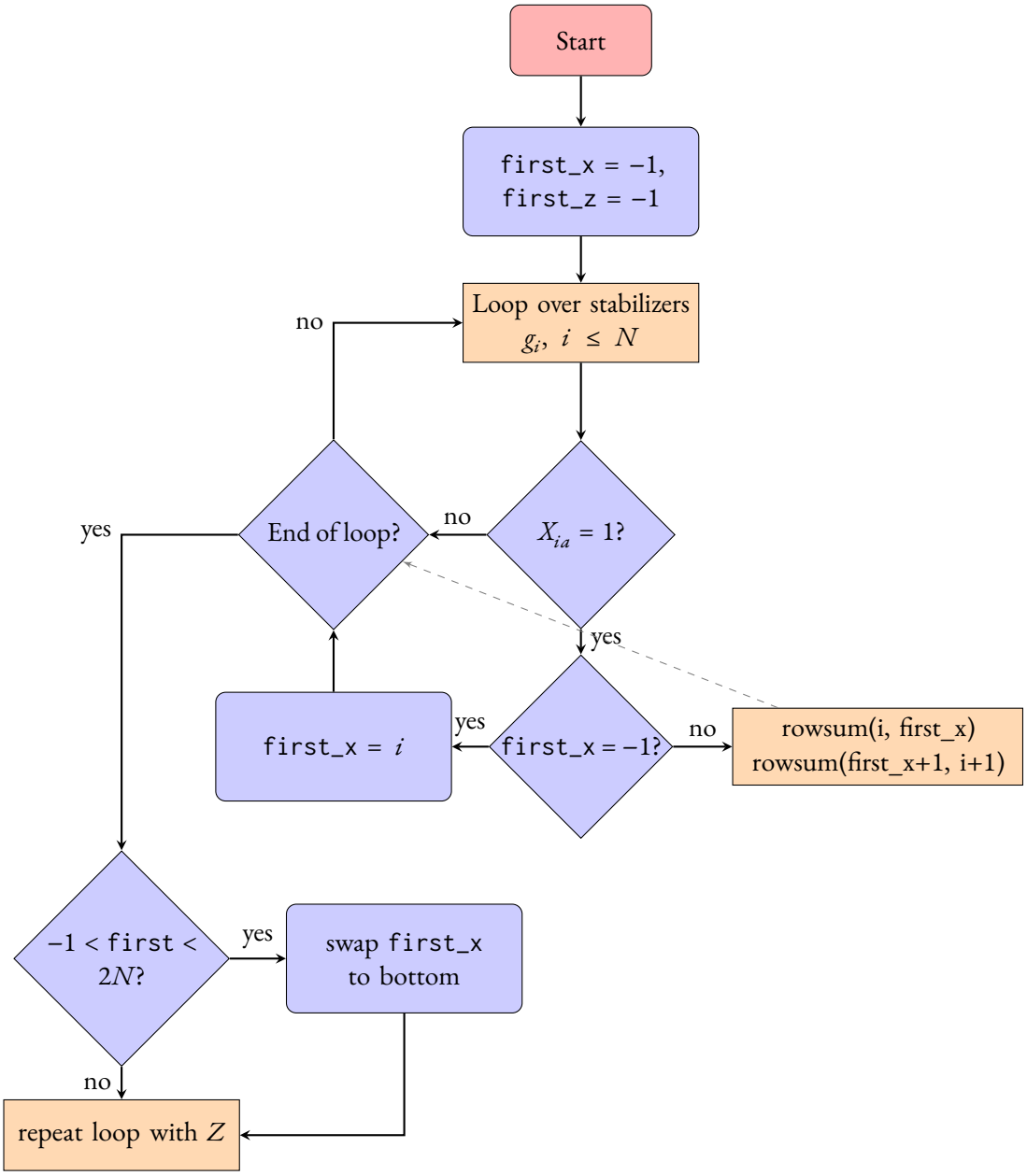


Figure 4.2.: Flowchart representation of the rowreduce subroutine. Since the function returns nothing, or `void`, it terminates after the loop over Z stabilizers. The "repeat loop with Z" box is therefore to be read as the loop over stabilizers with every x or X replaced by Z.

4. As we started from a valid tableau, fulfilling the commutation relations, we need to verify this point only for one of the combinations, namely

$$[\tilde{g}_j, \tilde{h}_i] = [g_i g_j, h_i h_j].$$

This is done with another commutator identiy,

$$[AB, CD] = A[B, C]D + [A, C]BD + CA[B, D] + C[A, D]B.$$

We thus have

$$\begin{aligned}
[g_i g_j, h_i h_j] &= g_i \underbrace{[g_j, h_i]}_{=0} h_j + [g_i, h_i] g_j h_j + h_i g_i [g_j, h_j] + h_i \underbrace{[g_i, h_j]}_{=0} g_j \\
&= (g_i h_i - h_i g_i) g_j h_j + h_i g_i (g_j h_j - h_j g_j) \\
&= g_i h_i g_j h_j - h_i g_i g_j h_j + h_i g_i g_j h_j - h_i g_i h_j g_j \\
&= \underbrace{\{g_i, h_i\}}_{=0} g_j h_j - h_i g_i \underbrace{\{g_j, h_j\}}_{=0} \\
&= 0.
\end{aligned}$$

□

With all the subtleties accounted for, we can begin to construct `rowreduce`. It should take a qubit position as input, and return `void`, since it merely modifies the matrix. We start by looping through the X stabilizers. The first time where $X_{ia} = 1$, the row number i is stored in a variable `first_x = i`. For each subsequent row with $X_{ka} = 1$ we call `rowsum(k,first_x)` and `rowsum(first_x+1,k+1)`, where `h+1` is the associated antistabilizer to stabilizer `h`. After looping through the X stabilizers, we move row `first_x` to the bottom if necessary. We then repeat the previous procedure with the Z stabilizers, also moving `first_z` to the bottom if necessary. The rowreduce algorithm is summarized in Algorithm 4.15 and represented as a flowchart in Figure 4.2.

Algorithm 4.15 (Rowreduce)**.** Let a denote the qubit we want to reduce. First, set two helper variables `first_x` and `first_z` to -1. Then loop over all rows in the X matrix of the stabilizers. If $X_{ia} = 1$ for the first time, set `first_x` to i. For any subsequent rows j with $X_{ja} = 1$, call rowsum(j, `first_x`) and the dual in the antistabilizers according to Proposition 4.14. Once the loop hits the end, swap row i to the bottom of the tableau, depending on if `first_x` $\neq -1$. Then repeat the previous steps for the Z matrix.

A last thing to remark is the scaling behavior. Since we essentially perform Gaussian elimination on one (or two) columns only, we reduce the time complexity from $\mathcal{O}(n^3)$ of the usual Gaussian elimination algorithm to $\mathcal{O}(n^2)$.

Partial trace

We can now combine the two previous subroutines to an algorithm realizing the partial trace, also refered to as `ptrace` for short. The algorithm traces out one qubit and modifies the remaining stabilizers accordingly. For instance, tracing out a qubit in a product state will just remove this qubit, since it does not correlate with any other qubit. If we do have correlations in the form of entanglement, the algorithm will modify the remaining stabilizers in a way that increases the mixedness.

Consider the paradigmatic example of the two-qubit Bell state

$$|\phi\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \quad \text{with density matrix} \quad \rho = |\phi\rangle\langle\phi| .$$

After tracing out any of the two qubits we are left with a mixed state of the form

$$\rho_i = \frac{1}{2}\left[|0\rangle\langle 0|_i + |1\rangle\langle 1|_i\right],$$

which is the maximally mixed state for qubit i. The analogous operation in the stabilizer picture is starting with the state stabilized by $\mathrm{Stab}(|\phi\rangle) = \langle ZZ, XX\rangle$. Then, by means of partially tracing out one qubit arriving at $\mathrm{Stab}(\rho_i) = \langle\rangle = \{I\}$.

Conversely, if we start out in a product state, such as $|\phi\rangle = |00\rangle$, and then trace out one qubit, the state of the qubit is still pure. Thus, the decrease in purity after the partial trace operation depends on the correlations of the qubit to be traced out. This is what we need the `get_state_type` function for. One of the steps in the partial trace function should be to increase or decrease the `mix` attribute by an amount depending on the state type. The value of this alteration is exactly the return value of `get_state_type` for the qubit to be traced out. With that we have a sensible first step in the `ptrace` algorithm, namely calling `get_state_type` and updating `mix` accordingly.

The next step is to move the columns corresponding to qubit a to position N. The reason for this is the same as why we moved the rows to the bottom in the `rowreduce` subroutine (see Algorithm 4.15). Since we decrease the system size by 1, we set N to $N - 1$, and since the tableau dimensions depend on this N explicitly, moving a column to N and then decrementing N by 1 amounts to deleting the column.

Then, we call rowreduce on the last column. After rowreduce has been called, the stabilizer generators of qubit a are in column N and row N, or possibly $N - 1$ too.

Finally, we decrement N, thereby removing the qubit from the system. The `ptrace` function is summarized in Algorithm 4.16.

Algorithm 4.16 (Partial trace)**.** Let a be the qubit to be traced out. First, call `get_state_type` (see Algorithm 4.13) and add its return value to `mix`. Then, move column a to N by means of transposition. Next, call rowreduce on column N. Finally, decrement N by 1.

4.3.5 Classical control of mixedness

The last point we want to discuss is the control we wield over the quantum simulation and the possible algorithms we can conceive to artificially introduce mixedness to our system. The algorithms used in Sections 3.3.1 and 3.3.2 were realized by selectively choosing the appropriate qubits to be cast to a mixed state. We therefore conclude the presentation of the algorithms added to the simulator by explaining how they function.

Minimal mixing

The first algorithm introduced is the more involved of the two. Here, we specifically choose a generator to be discarded such that the subgroup condition *should* be met in principle. The physics of the algorithm (and why it is a rather unphysical one) is discussed in detail in Section 3.3.1. Here, we only concern ourselves with the algorithm as it is implemented in the simulator.

The basic idea of the algorithm is similar to the projection function, but instead of returning `true` or `false` if a projection is successful or not, we do some more work on the tableau after the default projection algorithm would return `false`. In a sense, it is a softer version of a forceful projection, as it is done in the naive approach (see Section 3.2). Once we detect a faulty projection, we ensure that the measurement operator is included with the correct sign. This is done by imitating the case of a measurement with random outcome. In a way, we pretend that the projection has to alter the tableau (which is has to anyway) and insert the correct generator. We want to emphasize that one needs to take Proposition 4.14 into account when altering the tableau in this manner. Not being careful in this step could result in a tableau containing $-I$ as generator, which is not allowed.

After the correct generator is in place, we move it to the "mix" generators and increment `mix` by 1. The whole algorithm is summarized in Algorithm 4.17.

Algorithm 4.17 (Minimal mixing)**.** The algorithm works the same way as Algorithm 4.9 with one key exception. When a projection fails, the pertinent generator gets assigned the correct sign by doing the steps in case 1 of Algorithm 4.7. Then, swap rows such that the generator is with other mix generators, then increment mix by 1.

Maximal mixing

The second algorithm is rather simple. As we discard the entire generating set once a projection fails, there is no complex manipulation of the tableau in the subroutine. Instead, we set `mix=N` once the measurement outcome in the record does not match the expectation. For completeness' sake, Algorithm 4.18 summarizes the steps.

Algorithm 4.18 (Maximal mixing)**.** The algorithm works the same way as Algorithm 4.9 with a contingency added in case of failed projection. If a projection fails, set `mix=N`.

4.4 SUMMARY

In this chapter we introduced a simulation algorithm used to simulate a wide class of quantum circuits, known as Clifford circuits. We gave an overview of the existing infrastructure present to perform numerical experiments of quantum computations on a classical platform using the stabilizer formalism. We then derived algorithms for new functionalities employed in the simulations in the remainder of the thesis. In particular, we introduced the simulability of mixed states by extending the previous functions and subroutines to handle this class of quantum states appropriately. We furthermore introduced new functions that only become well-defined on mixed states.

CONCLUSION AND OUTLOOK

In this thesis we investigated different approaches to detect the measurement-induced entanglement phase transition in the projective transverse-field Ising model. In particular, the approaches employed measures related to cross entropy: a linearized cross entropy, and an upper bound on the entanglement entropy via Klein's inequality, which features the cross entropy.

In Chapter 1 we provided a broad overview of the core concepts pertaining to the thesis. We introduced the stabilizer formalism and its connection to classical simulations of quantum circuits. We then gave an introduction to the theory of entanglement transitions, with the projective transverse-field Ising model (PTIM) as the primary paradigm used throughout the rest of the thesis. We then explained the sampling problem in greater detail.

In Chapter 2 we discussed the first approach, the linear cross entropy. We gave an intuition behind the quantity and derived a method to efficiently compute it in Clifford circuits. We then utilized this method to investigate the behavior of the linear cross entropy in classical simulations of the projective transverse-field Ising model. Furthermore, we showed that the linear cross entropy factorizes elegantly into a product of two separate quantities dependent only on one type of measurement. Finally, we highlighted its behavior in noisy realizations of the PTIM by employing an error model with different error types.

In Chapter 3 we examine the approach based on bounding the entanglement entropy from above. In particular, we employed Klein's inequality to provide an upper bound with the cross entropy. We first derived the inequality in general, before giving an expression thereof in the stabilizer formalism. We then employed the derived expression in stabilizer simulations of the PTIM, exploring the utility of different post-processing algorithms with the previously presented noise protocol. We finalize the chapter by proposing two different regularization Ansätze to deal with infinities and discussing their utility as well.

It is clear that stabilizer simulations played an important part in the scope of this thesis. We therefore dedicated Chapter 4 to outlining the simulation algorithm, as well as the introduction of new functions and subroutines added to the simulator, specifically relevant to the thesis. These most notably include the addition of mixed states to the simulator, as well as the adaptation of the previously existing functionalities.

OUTLOOK

While we exhaustively covered the linear cross entropy and different numerical approaches, one could more closely investigate the subgroup condition of the cross entropy. While the

proof of the condition is rigorous and makes sense in the context of information theory, single qubits in orthogonal product states equally do not contribute to the entanglement entropy. One could thus try an approach, where one is agnostic to the signs when comparing product states on the same site with the cross entropy.

Furthermore, one could try other approaches, which are not explicitly discussed in this thesis. For instance, one could employ other measures, such as the entanglement asymmetry, or imagine entirely new algorithms, possibly by combining the approaches of the upper bound and the linear cross entropy.

A

CODE LISTINGS

C++ code for the computation of χ_C.

Listing A.1: Computation of the linear cross entropy for one measurement history of a circuit

```
bool success = true;
for (int t=0; t<t_max; t++) {
    for (int s=0; s<L; s++) {
        if (random_circuit_x[t][s] && success) {
           success = P.project(X,s,measurements_x[t][s]);
        }
    }
    for (int s=0; s<L-1; s++){
        if (random_circuit_zz[t][s] && success) {
           success = P.project(Z,s,Z,periodic(s+1,L),measurements_zz[
    t][s]);
        }
    }
}
if (success) {
results.lxs.av++;
results.lxs.se++;
}
```

C++ code for the `entropy_vn` function, computing the von Neumann entropy.

Listing A.2: `entropy_vn` function in the simulator

```
int Qubits::entropy_vn() {
    return mix;
}
```

C++ code for the `is_subgroup_of` function.

Listing A.3: `is_subgroup_of` function in the simulator

```
      bool Qubits::is_subgroup_of(Qubits& other) {
          if (other.N != N || other.mix > mix) return false;


          // Signs
          BYTE* comb_r = new BYTE[(2*N+1+7)/8];
```

```
7
8           for (int i=0; i<(2*N+1+7)/8; i++)
9               comb_r[i] = 0;
10
11          for (int g = 0; g<N-other.mix; g++) {
12              set_bit(comb_r,g,get_bit(other.r,2*g));
13          }
14          for (int g = N; g<2*N-mix; g++) {
15              set_bit(comb_r,g,get_bit(this->r,2*(g-N)));
16          }
17
18
19          // Table
20          int Nb = (2*N+1+7)/8; // N X stabilizer, N Z stabilizer
21
22          BYTE* loc_buf = new BYTE[2*N*Nb];
23          for (int i=0; i<2*N*Nb; i++) loc_buf[i]=0;
24          // 2N rows, since we're comparing two stabilizer matrices
25          BYTE** local = new BYTE*[2*N];
26          for (int r=0; r<2*N; r++) {
27              local[r] = loc_buf+Nb*r;
28          }
29          for (int i=0; i<2*N; i++) {
30              for (int j=0; j<((2*N+1)+7)/8; j++)
31                  local[i][j] = 0;
32          }
33
34          for (int c=0; c<N; c++) { // fill columns
35              for (int r=0; r<N-other.mix; r++) { // fill rows 1-N
    with other
36                  set_bit(local[r],2*c,other.has(X,c,2*r));
37                  set_bit(local[r],2*c+1,other.has(Z,c,2*r));
38              }
39              for (int r=N; r<2*N-mix; r++) { // fill rows N+1-2N
    with this
40                  set_bit(local[r],2*c,this->has(X,c,2*(r-N)));
41                  set_bit(local[r],2*c+1,this->has(Z,c,2*(r-N)));
42              }
43          }
44
45
46            int R=0;
47            for (int c=0; c<2*N; c++) {
48
49  #if DEBUG_ES
50                /* print_bit_matrix(local,N,2*n); */
51  #endif
52
53                // Find pivot
```

```
                bool pivot = true;
                for (int r=R; r<2*N; r++) {
                    if (get_bit(local[r],c)) {
                        // Swap pivot to top
                        if (pivot && r<N) {
                            swap_rows(local,R,r);
                            swap_bits(comb_r,R,r);
                            pivot = false;
                        }
                        // pivot should never be on the bottom,
                        // since it is then no longer a subgroup
                        else if (pivot && r>=N) {
                            return false;
                        }
                        // Zero column below pivot
                        else {
                            add_row_to(local,Nb,R,r);
                            // XOR the bit at position r in comb_r[r]
     with the bit at position R in comb_r[R]
                            comb_r[r / 8] ^= ((comb_r[R / 8] >> (R %
    8)) & 1) << (r % 8);
                        }
                    }
                }
                if (!pivot) R++;
            }
            // check that there is no hanging sign anywhere
            // This would be the case if the groups matched perfectly
     up to
            // sign differences
            for (int i=N; i<2*N-mix; i++) {
                if (get_bit(comb_r, i)) return false;
            }

            return true;


            delete[] comb_r;
            delete[] local;
            delete[] buf;

        }
```

C++ code for the `cross_entropy` function.

Listing A.4: `cross_entropy` function in the simulator

```
double Qubits::cross_entropy(Qubits& other) {
    if (!(this->is_subgroup_of(other))) return std::numeric_limits<
    double>::quiet_NaN();
```

```
    else return (double) mix;
}
```

C++ code for the `relative_entropy` function.

Listing A.5: `relative_entropy` function in the simulator

```
double Qubits::relative_entropy(Qubits& other) {
    if (!(this->is_subgroup_of(other))) return std::numeric_limits<
    double>::quiet_NaN();
    else return (double) mix - other.mix;
}
```

C++ code for the `get_state_type` function.

Listing A.6: `get_state_type` function in the simulator

```
int Qubits::get_state_type(const int q) {
    int dummy{}, val{};
    for ( int i = 0; i < 2*(N-mix); i+=2 ) {
        val = 2*get_bit(tab[q],2*i)+get_bit(tab[q],2*i+1);
        if ( val != 0 && dummy == 0 ) dummy = val;
        else if ( val != 0 && val != dummy ) return 1;
    }
    if (dummy) return 0;
    else return -1;
}
```

C++ code for the `rowreduce` function.

Listing A.7: `rowreduce` function in the simulator

```
void Qubits::rowreduce(const int q) {
   // helper variables
   int first_x = -1, first_z = -1;
   int num_x{}, num_z{};
   int last_stab = 2*(N-1), last_anti = 2*N-1;

   // loop through x stabilizer
   for (int j = 0; j < 2*N; j += 2) {
       if ( get_bit( tab[q], 2*j ) && num_x == 0 ) {
           first_x = j;
           num_x++;
       }
       else if ( get_bit( tab[q], 2*j ) && num_x > 0 ) {
           rowsum(j,first_x);              // update stabilizer
           rowsum(first_x+1, j+1);     // update antistabilizer in
    parallel
           num_x++;
       }
   }
```

```
19
20      if ( num_x > 0 && first_x != last_stab ) {
21          for (int k = first_x; k<last_stab; k++){
22              rowswap(k,k+2);
23          }
24      }
25
26
27      // loop through z stabilizer
28      for (int j = 0; j < 2*N; j += 2) {
29          if ( get_bit( tab[q], 2*j+1 ) && num_z == 0 ) {
30              first_z = j;
31              num_z++;
32          }
33          else if ( get_bit( tab[q], 2*j+1 ) && num_z > 0 ) {
34              rowsum(j,first_z);              // update stabilizer
35              rowsum(first_z+1,j+1);      // update antistabilizer in
        parallel
36              num_z++;
37          }
38      }
39
40
41      if ( num_z > 0 && ((num_x > 0 && first_z != last_stab) || num_x
        ==0)) {
42          for (int k = first_z; k<last_stab; k++){
43              rowswap(k,k+2);
44          }
45      }
46
47  }
```

C++ code for the ptrace function.

Listing A.8: ptrace function in the simulator

```
1   void Qubits::ptrace(const int q) {
2       assert(q>=0);
3       assert(get_num_qubits()>1);
4       assert(q<N);
5
6       // helper variables
7       BYTE* helper_tab = tab[q];
8       int last_stab = 2*(N-1);
9       int last_anti = 2*N-1;
10
11      // move the traced-out qubit to the end of the table so that the N
        -=1
12      // line at the end of this function becomes the tracing-out step
```

```
    // we need to do this before anything else, otherwise we break
     stuff,
    // since we do rowsum()
    for (int j = q; j < N - 1; j++) {
        tab[j] = tab[j + 1];
    }
    tab[N-1] = helper_tab;

    mix = mix + get_state_type(N-1);
    rowreduce(N-1);
    --N;
}
```

C++ code for the various `project` functions. An arument decides what to do upon failed projection. This function was exclusively used in Chapter 3, and not for the linear cross entropy.

Listing A.9: `project_or_mix` function in the simulator for a single qubit

```
void Qubits::project_or_mix(const int8 o, const int q, const bool m,
    const int alg) {

      // Random or not random?
      int p=-1;
          for (int h=0; h<2*(N-mix); h+=2) {
              if (!commutes(o,q,h)) {
                  p=h;
                  break;
              }
          }

      if (p==-1) {
          for (int h = 2*(N-mix); h<2*N; ++h) {
             if (!commutes(o,q,h)) {
                // Antistabilizer with 'mix' label anticommutes ->
    random
                p=h;
                break;
             }
          }
      }


      // Random result!
      if (p!=-1) {

          // Make projector o commute with everything except
    stabilizer p
          for (int h=0; h<2*N; h++) {
```

```
28              if ( !commutes(o,q,h) && h!=p ) rowsum(h,p);
29          }
30
31          // Make p new antistabilizer
32          if (p%2) copyrow(p-1,p);
33          else copyrow(p+1,p);
34
35
36          // Replace p by new stabilizer
37          zerorow(p);
38          set_qubit(o,q,p);
39          set_sign(p,m);
40
41          // add mix generators to (anti-)stabilizers
42          if (p >= 2*(N-mix)) {
43              if (p%2) {
44                  if (p==2*(N-mix)+1) {
45                      rowswap(p-1,2*(N-mix)+1);
46                  }
47                  else {
48                      rowswap(p-1,2*(N-mix)+1);
49                      rowswap(p,2*(N-mix));
50                  }
51                 --mix;
52              }
53              else {
54                 rowswap(p,2*(N-mix));
55                 rowswap(p+1,2*(N-mix)+1);
56                 --mix;
57              }
58          }
59
60      }
61      // Deterministic result!
62      else {
63          zerorow(2*N);
64          std::vector<int> dummy = {};
65          for (int h=0; h<2*N; h+=2) {
66              if ( !commutes(o,q,h+1) ) {
67                  rowsum(2*N,h);
68                  dummy.push_back(h);
69              }
70          }
71          if (!(get_bit(r,2*N) == (bool) m)) {
72              switch (alg) {
73                  case 0:
74                      {
75                         /* MINIMAL MIXING
76                          * ==============
```

```
77                          *
78                          * The algorithm works as follows:
79                          * 1. pretend that we just got a random
    result
80                          * on antistabilizers (i.e. measurement
    operator
81                          * anticommutes with some antistabilizer)
82                          * 2. perform subroutine of random result on
83                          * said antistabilizer
84                          * 3. update the corresponding _stabilizer_!
85                          * 4. swap the stab/destab rows to the mixed
     states
86                          * 5. increment mix
87                          */
88                         p = dummy[0];
89                         dummy.erase(dummy.begin());
90                         toggle_bit(r,p);
91                         for (auto const& h : dummy) {
92                             rowsum(p,h);
93                             rowsum(h+1,p+1);
94                         }
95                         if (p<2*(N-mix-1)){
96                             for (int i=p; i<2*(N-mix-1); ++i){
97                                 rowswap(i,i+2);
98                             }
99                         }
100                        mix++;
101                    } break;
102                case 1:
103                    {
104                        /* MAXIMAL MIXING
105                         * ==============
106                         *
107                         * This algorithm is pretty straightforward
    :
108                         * 1. add all generators to mixed
109                         * 2. done
110                         */
111                        mix = N;
112                    } break;
113                case 2:
114                    {
115                        p = dummy[0];
116                        dummy.erase(dummy.begin());
117                        for (int h=0; h<2*N; h++) {
118                            if ( !commutes(o,q,h) && h!=p )
119                                rowsum(h,p+1);
120                        }
121                        zerorow(p);
```

```
122                             set_qubit(o,q,p);
123                             set_sign(p,m);
124                         } break;
125                 }
126             }
127 
128         }
129 }
```

B

FRAGMENTS

ON TANGENTS — OR: WHAT IS THIS CHAPTER?

While writing this thesis my thoughts have wandered off once or twice. Almost always while writing out something completely different. Incidentally, this is how the paragraph you're reading right now came to be.

Sometimes I'd write down something that amuses me, sometimes I'd get frustrated and investigate my frustrations in text. Maybe I had even had a profound thought once in a while. Regardless of how these tangents got there, or what they are about, off-topic tangents are not something you'll want to put in the main body of your thesis, no matter how ingenious or witty they are. Come to think of it, they probably do not belong in the appendix either.

Either way, I would have been quite distraught to delete these paragraphs without any of them ever seeing the inside of a printer. So, in order for them not to go to waste, I decided to collect them here for mine and my friends' amusement. These texts really are silly sometimes.

ON REFERENCES

I like how references are provided in physics: all those little numbers in little brackets or as superscript above a claim to support it. It allows me to follow the bibliography in parallel to the paper and check some pertinent publications easily. This is, of course, very different in many other disciplines, e.g. psychology or philosophy, where it is common-practice to give your sources in alphabetical order sorted by the author's surname.

I claim that physicists are not appreciative enough of this. It will never fail to frustrate me if people cite some sh-t that has nothing at all to do with the claim in question. This is especially so if the claim is better supported by the results of other papers; papers that a lot of times make it into the `.bib` file of at least one of the authors, or why else would they be prominently cited in a section, where it doesnt make sense to do so. Ultimately, the highest degree of frustration for me is if the references are off by one, or cite a previous project by the same authors. This kind of situation lies in the uncanny valley of "is it them or me?". And more often than not, it should be *me*, right? But then you go on and on, read the cited papers thoroughly, try to understand the context in which it is referenced in the work you originally went through, and now, multiple layers deep in the reference rabbithole are you ultimately forced to realize that it's not you, it's them.

ON THE UTILITY OF SIMULATIONS

One of my favourite video games is *Cities: Skylines*. As the name suggests, it is a city-building game (which graciously runs natively on Linux). In it, you act as an almost omnipotent mayor and are tasked with developing and maintaining a settlement raised from scratch. Initially, one has a single plot of buildable area, located in a vast landscape, acting as a canvas for you to design your city as you see fit. The initial plot of land usually conveniently features a highway exit, where the canonical first step is the design of a road network with appropriate residential, commercial, and industrial zoning for citizens to move in, and for industry and businesses to ship cargo in and out. As the game progresses your city grows, unlocking you more buildings and services such as schools, hospitals, or public transit. (If the populace is sufficiently educated, one can even unlock a "Large Hadron Collider" in the later stages of the game.) However, one also faces the challenges of a growing city, such as the increasing demands for basic utilities, and most importantly, traffic congestion and its consequences. All this is to say: it simulates the progression from a village to a metropolis.

At the end of the day, however, it is only a simulation of what is happening in the real world. While the developers made sure to accurately model behavior of the masses, daily routines of thousands—if not millions—of citizens, and their commutes constituted by about as much vehicles, they at the same time made sure that I can play this game on my used ThinkPad, more than 10 years of age. It is therefore only natural that some mechanisms of human behavior had to be "coarse grained" away for the game to be (a) playable and (b) enjoyable. (Think about walking around in Stuttgart: construction sites as far as the eye can see. Confer *Cities: Skylines*, where buildings just spawn out of nowhere as soon as you command them to be placed.) It would be highly absurd to suggest that every nuance of life in a city can be mapped one to one to a program.

Yet as it turns out, the game does a surprisingly good job of accurately depicting traffic flow and the dynamics of city life, now being used to teach urban planning and landscaping students[1] [87, 88]. Picture this: Cleverly assembled instructions run on cleverly assembled silicon circuits teaching urban planning students that the road network they conceived of was not-so cleverly assembled. An almost kafkaesque sight to behold.

Now, what can we as physicists learn from this excursion into my video game preferences? The daily routine of many physicists today, even at the LHC, consists of running simulations on (almost) the same x86 instruction set that allows me to build the Large Hadron Collider in my virtual city with the press of a button. That is, the computer hardware is completely agnostic to the simulation it is running. Matter of fact, within the confines of computability, we can play God with the processors, making them produce any result we want.

This begs the question: what kind of results *should* we want from a simulation? Remember, the truly remarkable thing about the game Cities: Skylines is the way it accurately

[1] Educational software, now in your local steam library.

portrays *the real world*, while still being a video game. Should our demands on simulations of natural phenomena therefore be held to the same standard? If that were strictly so, a radical conclusion one might draw is that the epistemic value of a simulation that models something not translateable to an experiment is nill. It implies that the simulation tells me about as much about the phenomenon it is supposed to model as the following `C` code:

```
#include <stdio.h>
int main(int argc, char *argv[]) {
  printf("Hello World!");
  return 0;
}
```

Or maybe it tells me that whatever phenomenon whose existence I advocate for, is not "real"—whatever that means for you.

I claim that this, albeit a straw man argument, is short-sighted. The extrapolation from well-established models to potentially untestable theories that only exist on paper or computer simulations *can* have epistemic value.[2] One might still be able to answer *how* they are untestable, potentially gaining insight in the workings of nature by understanding the restrictions it shackles us with.

ON CARGO CULT SCIENCE

In my time at the institute I was fortunate enough to be able to attend the group retreat. For this I prepared a talk on Feynman's notion of Cargo Cult Science and how it connects to Max Horkheimer and Theodor W. Adorno's Dialectic of Englighenment. This might be the biggest tangent i went on, since I wrote an entire transcript of what I was going to say in the talk. This, of course, isn't an exact transcription of what was presented, but a rough outline and still a major tangent when writing this thesis.

The Talk

Hello everyone, it humbles me to start this day of talks by fellow members of this institute with some topic merely tangentially related to the field of theoretical physics. Nonetheless, I hope you'll enjoy the talk and the topic interesting and thought-provoking. And alas, we will not leave the realm of theory: As you can see, the title of my talk is *Cargo Cult Science and the Dialectic of Enlightenment*, and in it I wanna give a brief, and hopefully entertaining, overview of critical theory's foundations.

To set the stage, I'd like to begin with a quote from a figure we all hold in high esteem: Richard Feynman. In Caltech's 1974 commencement address, he coined the term I used for

[2] Maybe this is just me coping with the fact that there is no pompous publishable result in my thesis, but I digress.

the first half of this talk's title. It is a reflection that resonates suprisingly well with the theme of today's talk. He said [89]:

> During the Middle Ages there were all kinds of crazy ideas, such as that a piece of rhinceros horn would increase potency. Then a method was discovered for separating the ideas— which was to try one to see if it worked, and if it didn't work, to eliminate it. This method became organized, of course, into science. And it developed very well, so that we are now in the scientific age. [...]
>
> But even today I meet lots of people who sooner or later get me into a conversation about [...] some form of mysticism.

Feynman, whose brilliance in both physics and pedagogy I think we all deeply admire, captures a central tension that has persisted throughout the history of human thought. His observation, highlighting the transition from superstition to science, echoes the broader narrative of the Enlightenment—a period that heralded the triumph of reason and empirical inquiry over myth and mysticism.

Because surely, this method, the scientific method as it were, which we hold as bedrock of our endeavors, is a product of the Enlightenment era, of enlightenment thought. And surely, this thesis of pure reason and rationality, stands in contradiction, in antithesis, to UFOs or astrology.

But here's where it gets interesting: the story of the Enlightenment is not as straightforward as a simple victory of light over darkness. As it turns out, two german philosophers, Max Horkheimer (who was born in Zuffenhausen) and Theodor W. Adorno have decades prior also grappled with the complexities and contradictions inherent in Enlightenment thought. So much so that, when they were forced into exile by the emergence of fascism in the form of national socialism, they wrote a book about it, entitled *Dialectic of Enlightenment*.

In this work, Horkheimer and Adorno argue that the very rationality and scientific progress that emerged from the englightenment era, contain within them the seeds of their own undoing. Or to say it in their words [90, 91]: " Seit jeher hat die Aufklärung im umfassendsten Sinn fortschreitenden Denkens das Ziel verfolgt, von den Menschen die Furcht zu nehmen und sie als Herren einzusetzen. Aber die vollends aufgeklärte Erde erstrahlt im Zeichen triumphalen Unheils. " " Enlightenment, understood in the widest sense as the advance of thought, has always aimed at liberating human beings from fear and installing them as masters. Yet the wholly enlightened earth is radiant with triumphant calamity. "

Enlightenment

So what *is* a *Dialectic of Enlightenment*? Well, let us maybe begin with the Age of Enlightenment. If I would do this talk in german, I could refer to the english translation of the term *Aufklärung*. It is sometimes referred to as the *Age of Reason* [92], and caused quite a ruckus in 17th and 18th century europe. Think of the american and french revolution. The rise

of the first democracies, secularism and also: industrialization. The invention of the steam engine and so on. The most prominent englightenment thinkers include characters such as Immanuel Kant, Francis Bacon, David Hume and this one, I am sure you are all well familiar with the coordinate system named after him.

Enlightenment at its core represented a radical shift away from the teachings of priests and witch doctors, explanations from religion and myth. Enlightenment, in Kant's words, is "the human being's emergence from self-incurred minority. Minority is inability to make use of one's own understanding without direction from another."; "Understanding without direction from another" is understanding guided by reason [93, 94]. So instead of relying on horror stories, the likes of Galileo, Newton or Kepler made use of their own understanding without direction from another.

Francis Bacon, one of the intellectual architects of this movement, famously argued that mythical thinking stood in the way of what he called "the happy match between the mind of man and the nature of things" [95]. For Bacon, and indeed for many Enlightenment thinkers, myths and superstitions were obstacles that clouded human understanding. The goal was to clear these obstacles, to purify thought by aligning it more closely with the natural world, which could be understood through careful observation, experimentation, and the application of reason.

But as you could probably guess, this profound shift in thought was not without its complexities and contradictions. After all, why would someone as deeply embedded in the scientific tradition as Richard Feynman identify a phenomenon like Cargo Cult Science if we are truly living in a scientific age? Did the Enlightenment not champion ideals of liberty, progress, and rationality that should have eradicated such pseudo-scientific practices?

Instrumental reason

Horkheimer and Adorno would argue that yes, the Enlightenment indeed promoted these ideals. It encouraged a pursuit of knowledge that liberated humanity from the shackles of superstition and myth, setting the stage for unprecedented progress in science, technology, and human understanding. However, they also caution that this progress came with unintended consequences. Alongside the liberation that Enlightenment thought provided, it also introduced new forms of control and domination, encapsulated in what they describe as "instrumental reason".

To quote the Dialectic once more, Horkheimer and Adorno write:

> Myth becomes enlightenment and nature mere objectivity. Human beings purchase the increase in their power with estrangement from that over which it is exerted. Enlightenment stands in the same relationship to things as the dictator to human beings. He knows them to the extent that he can manipulate them. The man of science knows things to the extent that he can make them. Their "in-itself" becomes "for him".

Instrumental reason refers to a way of thinking that prioritizes efficiency, utility, and control above all else. It's a kind of reasoning that asks not "What is true?" but rather "What works?" or "What can we do with this?" While this approach can lead to remarkable technological advancements and practical applications, it can also reduce human thought to a mere tool for manipulating the world, stripping it of deeper meaning or ethical consideration [96].

A seemingly harmless manifestation of this instrumental reason might be familiar to many of us—think of the gut reaction from your relatives when they ask you what you're currently working on, perhaps during a holiday dinner. After hearing your explanation, they might immediately ask, "What can you do with it? How can it be applied?" This reaction, though well-meaning, reflects a broader societal tendency to value knowledge primarily for its practical utility rather than for its intrinsic worth or the deeper understanding it provides.

And even Feynman can provide us with an example for instrumental reason: In the commencement address he says this while talking about scientific integrity:

> For example, I was a little surprised when I was talking to a friend who was going to go on the radio. He does work on cosmology and astronomy, and he wondered how he would explain what the applications of this work were. "Well," I said, "there aren't any." He said, "Yes, but then we won't get support for more research of this kind." I think that's kind of dishonest.

This instrumental mindset, while useful in many contexts—and I cannot stress this enough: Yes of course, the human condition has been immensely improved by modern medicine and the likes—, also has the potential to narrow our vision. It can lead to a world where the worth of ideas is measured solely by their immediate applicability, where knowledge is pursued not for the sake of enlightenment but for the sake of control—whether that be control over nature, over society, or even over ourselves.

In this way, the Enlightenment's legacy is double-edged. While it has given us the tools to understand and shape the world in ways that were previously unimaginable, it also carries the risk of reducing all human thought and endeavor to mere instruments of utility, potentially leading us down a path where the original humanistic goals of the Enlightenment—liberty, progress, and the pursuit of truth—are overshadowed by the drive for efficiency and control.

Think of the developments of the information age: personal computers, cellphones, the internet. What has started in the pursuit of science has ended up within the total computability of human behavior. The incommensurable rigidity of the bureaucratic apparatus is also one striking example we can name. It is the result of a long tradition of rationalization. One of my favourite movies, Brazil by Terry Gilliam from 1985, brings this fetishization of bureaucracy to its radical, fascistoid conclusion.

This is but one of the Dialectics of Enlightenment, the thesis of liberation and the antithesis of control. To finish off, let us now return back to Feynman and his Cargo Cult Science. His idea is not too far off from the ones of Horkheimer and Adorno, in the sense that an element of the mythical is still remanent in the age of science. We have pseudoscience co-opting the language, the vocabulary of the "proper" sciences, and they even experiment and observe to test for their hypotheses.

This persistence of the mythical within the scientific age is a crucial point. Feynman's concept of Cargo Cult Science is not merely about quackery or obvious pseudosciences; it is a warning about the dangers that arise when the form of scientific inquiry is imitated without the substance.

In Cargo Cult Science, the methods of science—experimentation, observation, and even the use of scientific terminology—are employed, but they are disconnected from the rigorous skepticism, critical thinking, and openness to disconfirmation that define genuine scientific inquiry. The result is something that looks like science on the surface, but lacks the integrity and depth of true scientific understanding.

This is where the connection to Horkheimer and Adorno's critique becomes particularly relevant. Just as the Enlightenment's promise of liberation can be perverted into new forms of control, so too can the tools and language of science be co-opted in ways that undermine its true purpose. The rise of pseudoscience is a testament to this dialectical tension. Pseudoscience often flourishes not in opposition to science, but by mimicking it, by adopting its outward forms while stripping away its inner rigor.

Take, for example, the spread of misinformation in the digital age. The internet, a product of Enlightenment ideals about the free exchange of information, has become a breeding ground for both genuine scientific knowledge and dangerous pseudoscientific ideas. The same technology that allows us to access vast repositories of knowledge can also lead us down rabbit holes of conspiracy theories and falsehoods, all cloaked in the guise of scientific legitimacy.

Moreover, the bureaucratic rationalization we discussed earlier plays a role here as well. In the rush to measure, quantify, and control every aspect of human life, we may inadvertently create environments where pseudoscientific ideas can thrive. The pressure to produce results, the emphasis on measurable outcomes, and the bureaucratic obsession with procedures over substance can all contribute to a culture where Cargo Cult Science becomes increasingly prevalent.

Neural networks, machine learning algorithms, and large language models are the most recent culmination of Enlightenment thinking. However, every time we open the 'For You page' on our silicon-controlled block of glass and plastic, we find ourselves incapacitated anew.

wrapping up

So, what does this mean for us as scientists? It means we must remain vigilant. The Enlightenment gave us powerful tools for understanding the world, but it also gave us the responsibility to use those tools wisely. We must constantly remind ourselves of the difference between the appearance of science and its reality, between the mere performance of scientific procedures and the genuine pursuit of knowledge.

In conclusion, Feynman's cautionary tale about Cargo Cult Science is more relevant than ever. As we navigate the complexities of the modern world, filled with both the promise of technological advancement and the perils of pseudoscience, we must strive to uphold the true spirit of the Enlightenment. This involves not only applying rigorous scientific methods but also engaging in critical self-reflection, ensuring that our pursuit of knowledge remains aligned with the ideals of reason, progress, and, ultimately, human freedom.

BIBLIOGRAPHY

[1] Nicolai Lang and Hans Peter Büchler. "Entanglement Transition in the Projective Transverse Field Ising Model". In: *Physical Review B* 102.9 (Sept. 23, 2020), p. 094204. DOI: 10.1103/PhysRevB.102.094204 (cit. on pp. v, vii, 25, 27).

[2] Yaodong Li et al. "Cross Entropy Benchmark for Measurement-Induced Phase Transitions". In: *Physical Review Letters* 130.22 (June 1, 2023), p. 220404. DOI: 10.1103/PhysRevLett.130.220404 (cit. on pp. v–viii, 32, 35, 36, 60).

[3] Samuel J. Garratt and Ehud Altman. *Probing Post-Measurement Entanglement without Post-Selection*. Comment: 13 pages. Oct. 2, 2023. DOI: 10.48550/arXiv.2305.20092. arXiv: 2305.20092 [cond-mat, physics:quant-ph]. URL: http://arxiv.org/abs/2305.20092. Pre-published (cit. on pp. v, vi, viii, 32, 61, 67, 94).

[4] Maria Tikhanovskaya et al. *Universality of the Cross Entropy in $\mathbb{Z}_2$ Symmetric Monitored Quantum Circuits*. Comment: 12+6 pages, 16 figures. V2: References added. Aug. 14, 2023. DOI: 10.48550/arXiv.2306.00058. arXiv: 2306.00058 [cond-mat, physics:quant-ph]. URL: http://arxiv.org/abs/2306.00058. Pre-published (cit. on pp. vi, viii, 35, 36, 45, 47, 49, 50, 54, 57, 60).

[5] G W F Hegel. *Phenomenology of Spirit*. Galaxy Books. New York, NY: Oxford University Press, June 1979 (cit. on p. xiii).

[6] Critical Race Studies, director. *Keynote: Cheryl I. Harris*. scriptwriter Cheryl Harris. Jan. 12, 2016 (cit. on p. xiii).

[7] Simon Baron-Cohen et al. "The Autism-Spectrum Quotient (AQ): Evidence from Asperger Syndrome/High-Functioning Autism, Males and Females, Scientists and Mathematicians". In: *Journal of Autism and Developmental Disorders* 31.1 (Feb. 1, 2001), pp. 5–17. DOI: 10.1023/A:1005653411471 (cit. on p. 1).

[8] Michael Nielsen and Isaac Chuang. *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press, 2010 (cit. on pp. 2, 4, 66, 67).

[9] Daniel Gottesman. *Stabilizer Codes and Quantum Error Correction*. Comment: 114 pages, LaTeX. Caltech Ph.D. Thesis. May 28, 1997. DOI: 10.48550/arXiv.quant-ph/9705052. arXiv: quant-ph/9705052. URL: http://arxiv.org/abs/quant-ph/9705052. Pre-published (cit. on pp. 2, 97).

[10] Daniel Gottesman. "Class of Quantum Error-Correcting Codes Saturating the Quantum Hamming Bound". In: *Physical Review A* 54.3 (1996), pp. 1862–1868. DOI: 10.1103/PhysRevA.54.1862 (cit. on p. 2).

[11] Daniel Gottesman. *Surviving as a Quantum Computer in a Classical World*. University of Maryland, May 7, 2024 (cit. on pp. 2, 5–7).

[12] Scott Aaronson and Daniel Gottesman. "Improved Simulation of Stabilizer Circuits". In: *Physical Review A* 70.5 (Nov. 30, 2004), p. 052328. DOI: 10.1103/PhysRevA.70.052328 (cit. on pp. 2, 95, 97, 100).

[13] Scott Aaronson. "Introduction to Quantum Information Science Lecture Notes". In: () (cit. on p. 2).

[14] Scott Aaronson. "Introduction to Quantum Information Science II Lecture Notes". In: () (cit. on p. 2).

[15] Matthew P. A. Fisher et al. "Random Quantum Circuits". In: *Annual Review of Condensed Matter Physics* 14 (Volume 14, 2023 Mar. 10, 2023), pp. 335–379. DOI: 10.1146/annurev-conmatphys-031720-030658 (cit. on pp. 2, 16).

[16] Neil W Ashcroft and N David Mermin. *Solid State Physics*. Vol. 46. 1978 (cit. on p. 2).

[17] C. N. Yang and R. L. Mills. "Conservation of Isotopic Spin and Isotopic Gauge Invariance". In: *Physical Review* 96.1 (Oct. 1, 1954), pp. 191–195. DOI: 10.1103/PhysRev.96.191 (cit. on p. 2).

[18] L. Landau. "The Theory of Phase Transitions". In: *Nature* 138.3498 (Nov. 1936), pp. 840–841. DOI: 10.1038/138840a0 (cit. on pp. 2, 27).

[19] Nicolai Lang. *Lecture Notes on Topological Phases of Matter*. University of Stuttgart, 2021 (cit. on pp. 2, 25, 26).

[20] Anthony Zee. *Group Theory in a Nutshell for Physicists*. 1st ed. In a Nutshell Series v.17. Princeton: Princeton University Press, 2016. 1 p. (cit. on pp. 2, 3, 5).

[21] Markus Stroppel. *Höhere mathematik 1: Lineare algebra und konvergenz für ingenieure, mathematiker und physiker*. 2023 (cit. on p. 2).

[22] A. R. Calderbank et al. *Quantum Error Correction via Codes over GF(4)*. Sept. 10, 1997. DOI: 10.48550/arXiv.quant-ph/9608006. arXiv: quant-ph/9608006. URL: http://arxiv.org/abs/quant-ph/9608006. Pre-published (cit. on p. 7).

[23] A. R. Calderbank and Peter W. Shor. "Good Quantum Error-Correcting Codes Exist". In: *Physical Review A* 54.2 (Aug. 1, 1996), pp. 1098–1105. DOI: 10.1103/PhysRevA.54.1098 (cit. on p. 7).

[24] Andrew Steane. “Multiple-Particle Interference and Quantum Error Correction”. In: *Proceedings of the Royal Society of London Series A* 452 (Nov. 1, 1996), pp. 2551–2577. DOI: 10.1098/rspa.1996.0136 (cit. on p. 7).

[25] Charles H. Bennett and Gilles Brassard. *Quantum Cryptography: Public Key Distribution and Coin Tossing*. Mar. 14, 2020. DOI: 10.48550/arXiv.2003.06557. arXiv: 2003.06557. URL: http://arxiv.org/abs/2003.06557. Pre-published (cit. on p. 7).

[26] Charles H. Bennett and Stephen J. Wiesner. “Communication via One- and Two-Particle Operators on Einstein-Podolsky-Rosen States”. In: *Physical Review Letters* 69.20 (Nov. 16, 1992), pp. 2881–2884. DOI: 10.1103/PhysRevLett.69.2881 (cit. on p. 7).

[27] David Fattal et al. *Entanglement in the Stabilizer Formalism*. Comment: 4 pages, 3 figures. June 22, 2004. DOI: 10.48550/arXiv.quant-ph/0406168. arXiv: quant-ph/0406168. URL: http://arxiv.org/abs/quant-ph/0406168. Pre-published (cit. on pp. 11, 62, 67, 69, 111).

[28] Daniel Gottesman. *The Heisenberg Representation of Quantum Computers*. July 1, 1998. DOI: 10.48550/arXiv.quant-ph/9807006. arXiv: quant-ph/9807006. URL: http://arxiv.org/abs/quant-ph/9807006. Pre-published (cit. on pp. 15, 95, 97).

[29] E. Schrödinger. “Die gegenwärtige Situation in der Quantenmechanik”. In: *Naturwissenschaften* 23.48 (Nov. 1, 1935), pp. 807–812. DOI: 10.1007/BF01491891 (cit. on p. 16).

[30] A. Einstein, B. Podolsky, and N. Rosen. “Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?” In: *Physical Review* 47.10 (May 15, 1935), pp. 777–780. DOI: 10.1103/PhysRev.47.777 (cit. on p. 16).

[31] Edward Witten. *Notes on Some Entanglement Properties of Quantum Field Theory*. Aug. 6, 2018. DOI: 10.48550/arXiv.1803.04993. arXiv: 1803.04993. URL: http://arxiv.org/abs/1803.04993. Pre-published (cit. on p. 16).

[32] Adam Nahum et al. “Quantum Entanglement Growth under Random Unitary Dynamics”. In: *Physical Review X* 7.3 (July 24, 2017), p. 031016. DOI: 10.1103/PhysRevX.7.031016 (cit. on p. 16).

[33] Yaodong Li, Xiao Chen, and Matthew P. A. Fisher. “Quantum Zeno Effect and the Many-Body Entanglement Transition”. In: *Physical Review B* 98.20 (Nov. 19, 2018), p. 205136. DOI: 10.1103/PhysRevB.98.205136 (cit. on p. 16).

[34] Amos Chan et al. “Unitary-Projective Entanglement Dynamics”. In: *Physical Review B* 99.22 (June 24, 2019), p. 224307. DOI: 10.1103/PhysRevB.99.224307 (cit. on p. 16).

[35] Brian Skinner, Jonathan Ruhman, and Adam Nahum. "Measurement-Induced Phase Transitions in the Dynamics of Entanglement". In: *Physical Review X* 9.3 (July 22, 2019), p. 031009. DOI: 10.1103/PhysRevX.9.031009 (cit. on p. 16).

[36] Yaodong Li, Xiao Chen, and Matthew P. A. Fisher. "Measurement-Driven Entanglement Transition in Hybrid Quantum Circuits". In: *Physical Review B* 100.13 (Oct. 15, 2019), p. 134306. DOI: 10.1103/PhysRevB.100.134306 (cit. on pp. 16, 22).

[37] Yimu Bao, Soonwon Choi, and Ehud Altman. "Theory of the Phase Transition in Random Unitary Circuits with Measurements". In: *Physical Review B* 101.10 (Mar. 3, 2020), p. 104301. DOI: 10.1103/PhysRevB.101.104301 (cit. on pp. 16, 36).

[38] Yimu Bao, Soonwon Choi, and Ehud Altman. "Symmetry Enriched Phases of Quantum Circuits". In: *Annals of Physics*. Special Issue on Philip W. Anderson 435 (Dec. 1, 2021), p. 168618. DOI: 10.1016/j.aop.2021.168618 (cit. on p. 16).

[39] Finn Schmolke and Eric Lutz. *Measurement-Induced Quantum Synchronization and Multiplexing*. June 22, 2023. DOI: 10.48550/arXiv.2306.12986. arXiv: 2306.12986 [cond-mat, physics:quant-ph]. URL: http://arxiv.org/abs/2306.12986. Pre-published (cit. on p. 16).

[40] Jesse C. Hoke et al. "Measurement-Induced Entanglement and Teleportation on a Noisy Quantum Processor". In: *Nature* 622.7983 (Oct. 19, 2023), pp. 481–486. DOI: 10.1038/s41586-023-06505-7. arXiv: 2303.04792 [cond-mat, physics:hep-th, physics:quant-ph] (cit. on p. 16).

[41] Adam Nahum et al. "Measurement and Entanglement Phase Transitions in All-To-All Quantum Circuits, on Quantum Trees, and in Landau-Ginsburg Theory". In: *PRX Quantum* 2.1 (Mar. 30, 2021), p. 010352. DOI: 10.1103/PRXQuantum.2.010352 (cit. on p. 16).

[42] Adam Nahum and Brian Skinner. "Entanglement and Dynamics of Diffusion-Annihilation Processes with Majorana Defects". In: *Physical Review Research* 2.2 (June 4, 2020), p. 023288. DOI: 10.1103/PhysRevResearch.2.023288 (cit. on pp. 16, 24).

[43] Beni Yoshida. *Decoding the Entanglement Structure of Monitored Quantum Circuits*. Comment: 68 pages, many figures. Sept. 17, 2021. DOI: 10.48550/arXiv.2109.08691. arXiv: 2109.08691 [cond-mat, physics:hep-th, physics:quant-ph]. URL: http://arxiv.org/abs/2109.08691. Pre-published (cit. on p. 16).

[44] Brian Skinner. *Lecture Notes: Introduction to Random Unitary Circuits and the Measurement-Induced Entanglement Phase Transition*. Comment: 23 pages, 14 figures. July 6, 2023. DOI: 10.48550/arXiv.2307.02986. arXiv: 2307.02986 [cond-mat, physics:quant-ph]. URL: http://arxiv.org/abs/2307.02986. Pre-published (cit. on pp. 16, 20).

[45] Andrew C. Potter and Romain Vasseur. *Entanglement Dynamics in Hybrid Quantum Circuits*. Nov. 23, 2021. DOI: 10.48550/arXiv.2111.08018. arXiv: 2111.08018. URL: http://arxiv.org/abs/2111.08018. Pre-published (cit. on p. 16).

[46] Borivoje Dakić, Vlatko Vedral, and Časlav Brukner. "Necessary and Sufficient Condition for Nonzero Quantum Discord". In: *Phys. Rev. Lett.* 105.19 (Nov. 2010), p. 190502. DOI: 10.1103/PhysRevLett.105.190502 (cit. on p. 17).

[47] Johann von Neumann. *Mathematische Grundlagen Der Quantenmechanik*. Berlin: Springer, 1968 (cit. on pp. 18, 62).

[48] C. E. Shannon. "A Mathematical Theory of Communication". In: *The Bell System Technical Journal* 27.3 (1948), pp. 379–423. DOI: 10.1002/j.1538-7305.1948.tb01338.x (cit. on pp. 18, 62).

[49] Brian C. Hall. *An Elementary Introduction to Groups and Representations*. May 31, 2000. DOI: 10.48550/arXiv.math-ph/0005032. arXiv: math-ph/0005032. URL: http://arxiv.org/abs/math-ph/0005032. Pre-published (cit. on pp. 20, 72).

[50] John Cardy. "The Number of Incipient Spanning Clusters in Two-Dimensional Percolation". In: *Journal of Physics A: Mathematical and General* 31.5 (Feb. 1998), p. L105. DOI: 10.1088/0305-4470/31/5/003 (cit. on p. 24).

[51] J. L. Cardy. "Critical Percolation in Finite Geometries". In: *Journal of Physics A: Mathematical and General* 25.4 (Feb. 1992), p. L201. DOI: 10.1088/0305-4470/25/4/009 (cit. on p. 24).

[52] Joseph Rudnick and George Gaspari. "Bond Percolation on a Finite Lattice: The One-State Potts Model Reconsidered". In: *Journal of Statistical Physics* 42.5 (Mar. 1, 1986), pp. 833–860. DOI: 10.1007/BF01010448 (cit. on p. 24).

[53] Dietrich Stauffer Aharony Ammon. *Introduction To Percolation Theory: Second Edition*. 2nd ed. London: Taylor & Francis, Jan. 31, 2017. 192 pp. DOI: 10.1201/9781315274386 (cit. on pp. 24, 30).

[54] Ernst Ising. "Beitrag zur Theorie des Ferromagnetismus". In: *Zeitschrift für Physik* 31.1 (Feb. 1, 1925), pp. 253–258. DOI: 10.1007/BF02980577 (cit. on p. 25).

[55] Stephen Blundell. *Magnetism in Condensed Matter*. 1. publ., repr. Oxford Master Series in Physics Oxford Master Series in Condensed Matter Physics 4. Oxford: Oxford Univ. Press, 2011. 238 pp. (cit. on p. 25).

[56] Philippe Di Francesco, Pierre Mathieu, and David Sénéchal. *Conformal Field Theory*. Graduate Texts in Contemporary Physics. New York, NY: Springer, 1997. DOI: 10.1007/978-1-4612-2256-9 (cit. on p. 30).

[57] NobelPrize.org. *The Nobel Prize in Physics 2013 – Advanced Information*. Nobel Prize Outreach AB, 2013 (cit. on p. 31).

[58] Scott Aaronson. "Shadow Tomography of Quantum States". In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2018. New York, NY, USA: Association for Computing Machinery, June 20, 2018, pp. 325–338. DOI: 10.1145/3188745.3188802 (cit. on p. 31).

[59] *Complexity Zoo*. URL: https://complexityzoo.net/Complexity_Zoo (cit. on pp. 31, 96, 97).

[60] Crystal Noel et al. "Measurement-Induced Quantum Phases Realized in a Trapped-Ion Quantum Computer". In: *Nature Physics* 18.7 (July 2022), pp. 760–764. DOI: 10.1038/s41567-022-01619-7 (cit. on p. 31).

[61] Jin Ming Koh et al. *Experimental Realization of a Measurement-Induced Entanglement Phase Transition on a Superconducting Quantum Processor*. May 20, 2022. DOI: 10.48550/arXiv.2203.04338. arXiv: 2203.04338. URL: http://arxiv.org/abs/2203.04338. Pre-published (cit. on p. 31).

[62] Hirsh Kamakari et al. *Experimental Demonstration of Scalable Cross-Entropy Benchmarking to Detect Measurement-Induced Phase Transitions on a Superconducting Quantum Processor*. Mar. 1, 2024. DOI: 10.48550/arXiv.2403.00938. arXiv: 2403.00938. URL: http://arxiv.org/abs/2403.00938. Pre-published (cit. on pp. 31, 35).

[63] Lata Kh. Joshi et al. "Observing the Quantum Mpemba Effect in Quantum Simulations". In: *Physical Review Letters* 133.1 (July 1, 2024), p. 010402. DOI: 10.1103/PhysRevLett.133.010402 (cit. on p. 32).

[64] Filiberto Ares, Sara Murciano, and Pasquale Calabrese. "Entanglement Asymmetry as a Probe of Symmetry Breaking". In: *Nature Communications* 14.1 (Apr. 11, 2023), p. 2036. DOI: 10.1038/s41467-023-37747-8 (cit. on p. 32).

[65] Robert Butler, director. *The Cage*. scriptwriter Gene Roddenberry. Nov. 27, 1988 (cit. on p. 35).

[66] Maria Tikhanovskaya et al. "Universality of the Cross-Entropy in ${\mathbb{Z}}_{2}$ Symmetric Monitored Quantum Circuits". In: *Physical Review B* 109.22 (June 25, 2024), p. 224313. DOI: 10.1103/PhysRevB.109.224313 (cit. on p. 35).

[67] Chao-Ming Jian et al. "Measurement-Induced Criticality in Random Quantum Circuits". In: *Physical Review B* 101.10 (Mar. 3, 2020), p. 104302. DOI: 10.1103/PhysRevB.101.104302 (cit. on p. 36).

[68] Adam Nahum, Sagar Vijay, and Jeongwan Haah. "Operator Spreading in Random Unitary Circuits". In: *Physical Review X* 8.2 (Apr. 11, 2018), p. 021014. DOI: 10.1103/PhysRevX.8.021014 (cit. on p. 36).

[69] Tianci Zhou and Adam Nahum. "Emergent Statistical Mechanics of Entanglement in Random Unitary Circuits". In: *Physical Review B* 99.17 (May 20, 2019), p. 174205. DOI: 10.1103/PhysRevB.99.174205 (cit. on p. 36).

[70] Nicolai Lang. *Clifford Circuit Simulator*. Version 1.0. 2022 (cit. on pp. 42, 44, 95).

[71] Felix Roser, Hans Peter Büchler, and Nicolai Lang. "Decoding the Projective Transverse Field Ising Model". In: *Physical Review B* 107.21 (June 5, 2023), p. 214201. DOI: 10.1103/PhysRevB.107.214201 (cit. on pp. 43, 80).

[72] Samuel J. Garratt and Ehud Altman. "Probing Postmeasurement Entanglement without Postselection". In: *PRX Quantum* 5.3 (July 18, 2024), p. 030311. DOI: 10.1103/PRXQuantum.5.030311 (cit. on p. 61).

[73] Felix Leditzky. *Relative Entropies and Their Use in Quantum Information Theory*. Comment: 200 pages, 11 figures. PhD thesis, University of Cambridge, November 2016. Includes results from arXiv:1308.5961, arXiv:1403.2543, arXiv:1407.6616, arXiv:1506.02635, and arXiv:1604.02119 (see Section 1.4 for detailed information). Nov. 27, 2016. DOI: 10.48550/arXiv.1611.08802. arXiv: 1611.08802 [math-ph, physics:quant-ph]. URL: http://arxiv.org/abs/1611.08802. Pre-published (cit. on p. 64).

[74] Benjamin Schumacher and Michael D. Westmoreland. *Relative Entropy in Quantum Information Theory*. Comment: 36 pages; submitted for proceedings of the AMS special session on Quantum Information and Computation (January, 2000). Apr. 10, 2000. DOI: 10.48550/arXiv.quant-ph/0004045. arXiv: quant-ph/0004045. URL: http://arxiv.org/abs/quant-ph/0004045. Pre-published (cit. on p. 64).

[75] Victor Veitch et al. "The Resource Theory of Stabilizer Quantum Computation". In: *New Journal of Physics* 16.1 (Jan. 2014), p. 013009. DOI: 10.1088/1367-2630/16/1/013009 (cit. on p. 67).

[76] Sönke Niekamp, Matthias Kleinmann, and Otfried Gühne. "Entropic Uncertainty Relations and the Stabilizer Formalism". In: *Journal of Mathematical Physics* 53.1 (Jan. 1, 2012). Comment: 9 pages, 2 figures; published version, p. 012202. DOI: 10.1063/1.3678200. arXiv: 1103.2316 [quant-ph] (cit. on p. 67).

[77] Lorenzo Leone and Lennart Bittel. *Stabilizer Entropies Are Monotones for Magic-State Resource Theory*. Apr. 26, 2024. DOI: 10.48550/arXiv.2404.11652. arXiv: 2404.11652 [quant-ph]. URL: http://arxiv.org/abs/2404.11652. Pre-published (cit. on p. 67).

[78] Lorenzo Leone et al. "Phase Transition in Stabilizer Entropy and Efficient Purity Estimation". In: *Physical Review A* 109.3 (Mar. 1, 2024), p. 032403. DOI: 10.1103/PhysRevA.109.032403 (cit. on p. 67).

[79] Yu-qian Wu and Xiao-yu Chen. "Entanglement Upper Bound of Stabilizer Codewords". In: *2011 International Conference on Intelligence Science and Information Engineering*. 2011 International Conference on Intelligence Science and Information Engineering. Aug. 2011, pp. 73–76. DOI: 10.1109/ISIE.2011.9 (cit. on p. 67).

[80] V. Vedral and M. B. Plenio. "Entanglement Measures and Purification Procedures". In: *Physical Review A* 57.3 (Mar. 1, 1998), pp. 1619–1633. DOI: 10.1103/PhysRevA.57.1619 (cit. on p. 67).

[81] Noah Linden et al. "The Quantum Entropy Cone of Stabiliser States". Comment: 15 pages, uses lipics.cls (http://www.dagstuhl.de/en/publications/lipics). V2 is the final TQC 2013 proceedings version, it has numerous corrections, updated references, and a new co-author. 2013. DOI: 10.4230/LIPIcs.TQC.2013.270. arXiv: 1302.5453 [math-ph, physics:quant-ph] (cit. on p. 67).

[82] Kaifeng Bu. *Extremality of Stabilizer States*. Comment: 6+3 pages. Mar. 20, 2024. DOI: 10.48550/arXiv.2403.13632. arXiv: 2403.13632 [math-ph, physics:quant-ph]. URL: http://arxiv.org/abs/2403.13632. Pre-published (cit. on p. 67).

[83] Amir R. Arab. "Lecture Notes on Quantum Entanglement: From Stabilizer States to Stabilizer Channels". In: *Frontiers of Physics* 19.5 (Apr. 16, 2024), p. 51203. DOI: 10.1007/s11467-024-1397-4 (cit. on pp. 67, 97).

[84] Thomas M. Cover and Joy A Thomas. *Elements of Information Theory*. 2nd ed. Nashville, TN: John Wiley & Sons, June 2006 (cit. on p. 77).

[85] Quasimoto. *Discipline 99 Pt. 1*. 2658 Griffith Park #504 Los Angeles CA 90065 USA, 2000 (cit. on p. 95).

[86] Thomas H. Cormen, ed. *Introduction to Algorithms*. 3rd ed. Cambridge, Mass: MIT Press, 2009. 1292 pp. (cit. on p. 96).

[87] Tayyeb Ahmed Khan and Xin Zhao. "Perceptions of Students for a Gamification Approach: Cities Skylines as a Pedagogical Tool in Urban Planning Education". In: *Responsible AI and Analytics for an Ethical and Inclusive Digitized Society*. Ed. by Denis Dennehy et al. Cham: Springer International Publishing, 2021, pp. 763–773. DOI: 10.1007/978-3-030-85447-8_64 (cit. on p. 134).

[88] Pyry Haahtela. "Gamification of Education: Cities Skylines as an Educational Tool for Real Estate and Land Use Planning Studies". In: (2015) (cit. on p. 134).

[89] Richard P. Feynman. "Cargo Cult Science". In: *Surely You're Joking, Mr. Feynman!* (Cit. on p. 136).

[90] Max Horkheimer and Theodor W. Adorno. *Gesammelte Schriften Band 5: 'Dialektik der Aufklärung' und Schriften 1940-1950*. 4th ed. Max Horkheimer, Gesammelte Schriften in 19 Bänden (Taschenbuchausgabe) 5. Frankfurt am Main, Germany: FISCHER Tachenbuch, June 1987 (cit. on p. 136).

[91] Max Horkheimer and Theodor W. Adorno. *Dialectic of Enlightenment: Philosophical Fragments*. Ed. by Gunzelin Schmid Noerr. Stanford, Calif.: Stanford University Press, 2002 (cit. on p. 136).

[92] William Bristow. "Enlightenment". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta and Uri Nodelman. Fall 2023. Metaphysics Research Lab, Stanford University, 2023 (cit. on p. 136).

[93] Immanuel Kant. "Beantwortung der Frage: Was ist Aufklärung?" In: *Berlinische Monatsschrift* 4.1 (Dec. 1784), pp. 481–494 (cit. on p. 137).

[94] Immanuel Kant. *Kritik Der Reinen Vernunft*. 2nd ed. Vol. 3. Kants Werke. Akademie-Ausgabe. 1781 (cit. on p. 137).

[95] Francis Bacon. *The Works of Francis Bacon: Volume 1: Philosophical Works 1*. Ed. by James Spedding, Robert Leslie Ellis, and Douglas Denon Heath. Vol. 1. Cambridge Library Collection - Philosophy. Cambridge: Cambridge University Press, 2011. DOI: 10.1017/CBO9781139149549 (cit. on p. 137).

[96] Max Horkheimer. *Zur kritik der instrumentellen vernunft*. 1st ed. Figuren des wissens/bibliothek. Frankfurt am Main, Germany: FISCHER Taschenbuch, Dec. 2007 (cit. on p. 138).

CORRIGENDA

Since submitting this thesis, a number of orthographic and lexical errors have caught the author's eyes. Some of them are merely aesthetically displeasing, some of them actively hinder reading flow. (None of the errors corrected perturb the presented physics in any way, shape or form.) Thus, in pursuit of *das Schöne und Erhabene* in major passages of this text, we have performed some minor edits to approach it. This chapter is dedicated to listing all subsequent changes and post-hoc edits with reference to the main text.

In the very likely event of you, the reader, finding more mistakes, please do not hesitate to write me at etiennespringer@protonmail.com with the location and nature of the error.