universität wien

MASTERARBEIT | MASTER'S THESIS

Titel | Title

A Tensor Network Description of the X-Cube Model and a Class of Generalizations

verfasst von | submitted by

Johannes Wladika BSc

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2025

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 876
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Physics
Betreut von | Supervisor:	Univ.-Prof. Dr. Norbert Schuch
Mitbetreut von | Co-Supervisor:	Dr. András Molnár master MSc

Contents

Acknowledgments

I am grateful to my supervisor Norbert for introducing me into the exciting field of quantum information and for his invaluable support during my research. I express my deepest gratitude to András for offering me so many helpful discussions and for patiently explaining to me both basic and advanced concepts of tensor networks. I am also thankful to Refik and Adrián for discussions that brought me a great deal of joy during the research process. I gratefully acknowledge the support of the Vienna Doctoral School in Physics, which provided a research fellowship during the research phase of this thesis. Lastly, I want to thank my partner, Annika, as well as my friend, Sebastian, for listening to the often vague descriptions of the research and encouraging me throughout my studies.

Abstract

In this thesis, a tensor network description of the X-Cube model, a prototypical fracton model, is developed and analyzed. The construction provides an explicit representation of the model's ground state using a tensor network description, offering a new perspective on its entanglement structure and symmetries. A detailed analysis of the tensor symmetries reveals how fundamental aspects of the X-Cube model manifest within the tensor network formalism. In particular, three defining characteristics of the X-Cube phase are demonstrated: the sub-extensive ground state degeneracy on the torus, the presence of excitations with restricted subsystem mobility (fractons and lineons), and the layered structure of two-dimensional toric codes.
Additionally, a generalization of the X-Cube tensor network is proposed, which preserves the key features of the original model in the tensor network formalism.
Overall, this work advances the understanding of fracton phases through the lens of tensor networks, bridging insights from quantum information and many-body physics. The methods and results presented here lay the groundwork for further exploration of fractonic tensor networks, including the study of other models, symmetries, and groundstate structures of these exotic phases of matter.

Zusammenfassung

In dieser Arbeit wird eine Tensor Netzwerk Darstellung des X-Cube-Modells, einem beispielhaftem sogenanntem Fracton Modell, entwickelt und untersucht. Diese Beschreibung ermöglicht eine explizite Konstruktion des Grundzustands und eröffnet neue Einblicke in die Struktur der Verschränkung sowie in die zugrunde liegenden Symmetrien des Modells. Durch eine detaillierte Analyse der Symmetrien der Tensoren können drei zentrale Merkmale des X-Cube Modells aus Sichtweise des Tensor Netzwerk Formalismus identifiziert werden: eine Grundzustandsentartung auf dem Torus, die langsamer als das Volumen des Systems wächst, Anregungen mit eingeschränkter Beweglichkeit (Fractons und Lineons) sowie eine Schichtstruktur, die aus zweidimensionalen Toric Codes besteht.
Darüber hinaus wird eine verallgemeinerte Version des X-Cube Modells diskutiert, die die charakteristischen Eigenschaften des ursprünglichen Modells bewahrt.
Diese Arbeit leistet einen Beitrag zum Verständnis von Fracton Phasen aus Sicht der Tensor Netzwerke und verbindet Konzepte der Quanteninformation mit der Theorie der Vielteilchensysteme. Die entwickelten Methoden bieten eine Grundlage für zukünftige Untersuchungen weiterer Modelle und ermöglichen einen systematischen Zugang zur Analyse von Symmetrien und Grundzuständen in diesen neuartigen Phasen der Materie.

1 Introduction

One of the fundamental challenges in quantum many body physics is the so called "curse of dimensionality". In order to describe an arbitrary quantum state of an n-body system 2^n parameters need to be specified. However, actual physical systems interact mostly locally and follow constraints, such as the entanglement area law [3, 10]. The central task of the field is thus to find efficient representations that accurately describe physically relevant states within the exponentially large Hilbert space.

An approach to achieve this is the variational method. The fundamental idea behind the variational method is to first restrict the form of the wavefunction and then to approximate the true ground state by the lowest energy eigenstate of the restricted form. In one dimension, Matrix Product States (MPS) are particularly powerful. The ground state of any one-dimensional gapped local Hamiltonian can be well approximated by an MPS [7]. This idea extends to two dimensions through Projected Entangled Pair States (PEPS) [14, 13], which generalize MPS to higher spatial dimensions. Their structure has been instrumental in advancing our understanding of topological order [11] and symmetry-protected phases in 2D [1].

This thesis explores the application of PEPS to three dimensions, where fracton models such as the X-Cube [15] illustrate even richer phenomenology. The X-Cube model, a Type-I fracton system, hosts excitations with restricted mobility, distinct from the fractal excitations in models like Haah's code [6].

The goal of this thesis is to construct a tensor network representation of the X-Cube model and to investigate how immobile fracton excitations and sub-extensive groundstate properties manifest within the tensor network framework. This work lays the groundwork for broader studies of three-dimensional fracton models via tensor networks. The structure of this thesis is outlined in the following.

First, the basic formalism of Matrix Product States and Projected entangled pair states is introduced. We then review the fundamentals of stabilizer codes, which will be essential for understanding the X-Cube model and its tensor network representation.

In the main chapter of the thesis, the X-Cube tensor is derived and then analyzed with respect to various properties. The symmetry group of the tensor plays a particularly important role.

The third chapter discusses three known properties of the X-Cube model, which are an-

alyzed within the framework of the tensor network. The sub-extensive groundspace degeneracy can be computed within the tensor network formalism. The fractal excitations can be seen purely on the level of the tensor network for which invariants are violated if excitations are present. Finally, the layer structure of the X-Cube model is revealed by a specific action on the tensor network, which couples or decouples layers of toric code. The last chapter expands the results obtained for the Z_2 X-Cube model to a generalized Z_n X-Cube model.

1.1 Notation and Conventions

Throughout this thesis, all three-dimensional illustrations adopt a fixed coordinate system, as shown in Fig. 1.1. For two-dimensional plots, only the x- and z-axes of Fig. 1.1 will be used. This convention is maintained consistently throughout the thesis.

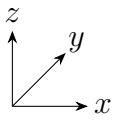


Figure 1.1: The coordinate system used consistently throughout the thesis.

Whenever tensor indices are explicitly labeled in diagrams, their positions correspond to the labeling convention shown in Fig. 1.2.

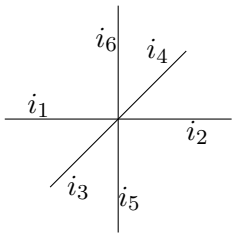


Figure 1.2: Index labeling convention for tensor vertices.

1.2 Introduction to MPS and PEPS

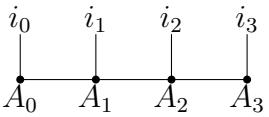
A Matrix Product State is a representation of a quantum state on a one-dimensional array of particles. A given state $|\phi\rangle$ can be expanded in a basis $|i_0 i_1 ... i_{n-1}\rangle$

$$|\phi\rangle = \sum_{i_0 ... i_{n-1}=0}^{d-1} C_{i_0 i_1 ... i_{n-1}} |i_0, i_1, ..., i_{n-1}\rangle \,. \tag{1.1}$$

The coefficients for the basis elements can be rewritten as a product of matrices A_i:

$$|\phi\rangle = \sum_{i_0,i_1...i_{n-1}=0}^{d-1} A_0^{i_0} A_1^{i_1} ... A_n^{i_{n-1}} |i_0, i_1, ..., i_{n-1}\rangle . \tag{1.2}$$

The matrices A_i have dimensions $d \times D_{i-1} \times D_i$, where the first and the last matrix in the product must have dimensions $d \times 1 \times D_0$ and $d \times D_{n-1} \times 1$ respectively. d denotes the local physical dimension, D_i are called the bond dimensions. The minimal required bond dimensions of the matrices $A_1...A_{n-2}$ depend on the state $|\phi\rangle$, more precisely, the entanglement of that state. Less entangled states can be represented with lower-dimensional matrices. An MPS can also be depicted in graphical notation.

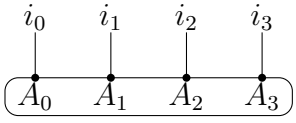
$|\phi\rangle =$ 

(1.3)

Each dot represents a matrix, or more generally a tensor. The edges connected to each dot represent vector spaces. Edges that are connected on both ends are contracted, which corresponds to the matrix multiplication in equation Eq. (1.2). We differentiate between physical and virtual edges. In the depiction above, horizontal edges represent virtual degrees of freedom and vertical edges represent physical ones. In this thesis, we will consider a slight variation of the formulation of MPS specified above, which is better suited to describe translation invariant states. We will refer to this construction as MPS with periodic boundary conditions. To write a state with periodic boundaries as an MPS, the first and last matrix are allowed to have $D_{-1} = D_n > 1$. It is defined as

$$|\phi\rangle = \sum_{i_0,i_1...i_{n-1}}^{d} \mathrm{Tr}\{A_0^{i_0} A_1^{i_1} ... A_n^{i_{n-1}}\} |i_0 i_1 ... i_{n-1}\rangle . \tag{1.4}$$

Graphically, this is represented by the introduction of another virtual edge:

$|\phi\rangle =$ 

(1.5)

In this thesis, only states for which $\forall i A_i = A_{i\oplus 1}$, will be considered. Hence, the bond dimensions reduce to one overall bond dimension D.
The MPS framework is defined for one-dimensional physical arrays. The framework can be extended to two or more dimensions. Writing higher-dimensional MPS representations as equations is not feasible. Instead, the graphical representation is presented as a recipe

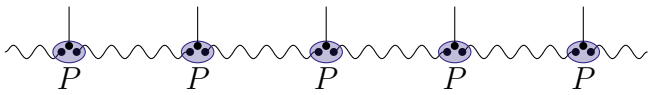


Figure 1.3: An illustration of an MPS as a one-dimensional PEPS. The wavy lines represent maximally entangled pair states (virtual bonds), and the vertical lines represent the physical degrees of freedom. The linear maps P^i are shown as blue shaded nodes acting on each site.

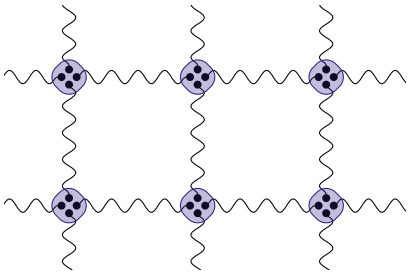


Figure 1.4: As in Fig. 1.3, the wavy lines represent maximally entangled pair states (virtual bonds), and the blue nodes indicate the action of the linear maps P. The physical degrees of freedom are omitted for clarity.

for how the state represented by the tensor network can be calculated. The generalization used in this thesis is called Projected Entangled Pair States. In order to see why this is a natural generalization of MPS it is useful to get another perspective on MPS.
A virtual entangled pair state, $|\psi\rangle = \sum_k^{d-1} |k\rangle |k\rangle$, is placed between every adjacent particles. Then, a linear map acts at the site of every physical particle:

$$P = \sum_{i,k,l} A^i_{kl} |i\rangle \otimes \langle k| \langle l| , \tag{1.6}$$

where A^i are the matrices of the original MPS. At every P one particle of each of the two adjacent entangled pair states $|k\rangle$ and $|l\rangle$ are used as inputs for the tensor A so that it returns the physical state $|i\rangle$. Fig. 1.3 presents this idea graphically.
Overall, this is just a reformulation of MPS, where the contraction along the tensors A^i is reinterpreted as the action on a shared virtual entangled state. This PEPS formulation can be generalized to more than one dimension. Each tensor P then uses more than two entangled states as input. For two dimensions, this is shown in Fig. 1.4. Moreover, each tensor may also output more than one physical degree of freedom. The X-Cube tensor, for example, takes six entangled states as input and maps to six physical degrees of freedom. In summary, the key concept of PEPS is that they provide an efficient description of a quantum state. The key object used for this description is a tensor placed on every physical site that may include multiple degrees of freedom. This tensor takes virtual entangled pair

states as input and produces a physical state.

1.3 Stabilizer codes

Stabilizer codes are a specific type of quantum code. That is, an encoding of information into the subspace of a Hilbert space, called the code space $\mathcal{C}$, such that the information on $\mathcal{C}$ is protected against certain noise. Stabilizer codes are particularly simple examples of qunatum error correction codes, since their encoding of information and correction of noise is mathematically easily accessible. They were introduced in [4] and it was proven that they allow universal quantum computation [5].
The underlying physical system is an array of qubits. For such quantum systems, the sensitivity against unwanted noise from the outside constitutes a problem since information is saved only on a small number of physical qubits. Relatively small disturbances could already lead to loss of information. Stabilizer codes are able to protect themselves against some errors by performing computations in the degenerate groundspace of the stabilizer code.
In the following, a brief overview of the theory behind the stabilizer formalism will be presented. This serves to establish an argument for the groundspace degeneracy of the X-Cube model. As stated above, the physical realization of the code is an array of qubits. Their associated local Hilbert spaces are therefore isomorphic $\mathcal{H} = \mathbb{C}^2$. On each of these local $\mathbb{C}^2$ Hilbert spaces, consider the Pauli operators X, Y, Z.

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{1.7}$$

The local Pauli group G_l acting on a single particle consists of 16 elements. It is generated by these operators:

$$G_l = \langle X, Y, Z \rangle = \{\pm\mathbb{1}, \pm i\mathbb{1}, \pm X, \pm iX, \pm Y, \pm iY, \pm Z, \pm iZ\}. \tag{1.8}$$

Next, consider tensor products of Pauli operators acting on distinct local Hilbert spaces. For example, $X \otimes Y \otimes X \otimes I$ is an operator acting on four sites. For simplicity, these will be written in abbreviated form as $XYXI$. For a fixed system size n, the group G^n includes all possible tensor products of Pauli operators and identities acting on n sites. It is the tensor product of the local G_l groups.

$$G^n = G_l \otimes G_l \otimes ... \otimes G_l \tag{1.9}$$

Elements of this group are acting on all sites of the system, although this action may be

the identity. In the following, n is fixed to the size of the underlying Hilbert space.

Definition 1 (Stabilizer)**.** *A subgroup S of G^n is called a stabilizer group if it is abelian and does not contain $-\mathbb{1}^{\otimes n}$.*

The elements $s \in S$ are referred to as stabilizers. The idea of the stabilizer formalism is to use the common $+1$ eigenspace of all s as the code space:

Definition 2 (Code space)**.** *Let S be a stabilizer group. Then, the code space $\mathcal{C}$ is defined to be*

$$\mathcal{C} = \mathit{span}\{|\phi\rangle \,|\forall s \in S : s\,|\phi\rangle = |\phi\rangle\}. \tag{1.10}$$

To every Stabilizer S we can assign a Hamiltonian, such that the groundspace of this Hamiltonian is precisely the code space $\mathcal{C}$. For that, let $B \subset S$ be a generating set of S. Then,

$$H = -\sum_{s\in B} s. \tag{1.11}$$

If all $s \in B$ are local, then the Hamiltonian is called a local Hamiltonian.

Theorem 1. *Let S be a stabilizer group and $\mathcal{C}$ the code space as defined above. Then,*

$$\dim\{\mathcal{C}\} = 2^{n-k}, \tag{1.12}$$

where k is the minimum size of a generating set B of S.

Proof. Let $B_k \subset S$ be a minimal generating set of S, such that

$$\langle B_k\rangle = S, |B_k| = k. \tag{1.13}$$

For each of the generators $s_i \in B_k$ define the projector

$$\Pi_i = \frac{1}{2}(\mathbb{1} + s_i) = \frac{1}{2}\sum_{j=0}^{1} s_i^j. \tag{1.14}$$

Then $\mathcal{C}$ can be obtained by applying all these projections onto the full 2^n dimensional Hilbertspace. The dimension of that subspace is then given by the trace over the projection:

$$\dim\mathcal{C} = \dim \mathrm{Im}\{\Pi_1\Pi_2...\Pi_k\} = \frac{1}{2^k}\,\mathrm{Tr}\{\Pi_1\Pi_2...\Pi_k\} = \frac{1}{2^k}\sum_{j_1...j_k=0}^{1} \mathrm{Tr}\{s_1^{j_1}s_2^{j_2}...s_k^{j_k}\}. \tag{1.15}$$

Since

$$\forall g \neq \mathbb{1}^{\otimes n} \in G^n\,\mathrm{Tr}\{g\} = 0, \tag{1.16}$$

only the trace of the identity contributes:

$$\dim \mathcal{C} = \frac{1}{2^k} \operatorname{Tr}\{\mathbb{1}^{\otimes n}\} = \frac{2^n}{2^k} = 2^{n-k}. \tag{1.17}$$

□

As $\mathcal{C}$ is the groundspace of H, this Theorem implies that

$$\log_2 \mathrm{GSD} = n - k. \tag{1.18}$$

We distinguish between three different kinds of operators of G^n. First, operators in S are stabilizers, which have been discussed above. The second kind are logical operators

$$g \in G^n, g \notin S, \forall s [g, s] = 0. \tag{1.19}$$

They are called logical operators, because they may act non-trivially on the groundspace:

$$\forall s : s\, g\, |\phi\rangle = g\, s\, |\phi\rangle = g\, |\phi\rangle\,. \tag{1.20}$$

The third kind of operators are errors

$$e \in G^n, \exists s : \{e, s\} = 0. \tag{1.21}$$

An error acting on the system will make it leave the groundspace:

$$\exists s : s\, e\, |\phi\rangle = -e\, s\, |\phi\rangle = -e\, |\phi\rangle\,. \tag{1.22}$$

The idea behind error correction on $\mathcal{C}$ is the following. Should an error e occur, a measurement of all stabilizers s will reveal the error. This measurement is referred to as a syndrome measurement. Its outcome is a string of the measurements of all stabilizer terms. Since all stabilizers are in G^n, that is a string of $+1$ and -1 values. The -1 reveal where the error occurred, allowing it to be corrected.
As the final step in this section, one of the simplest and smallest examples of a stabilizer code is presented, to demonstrate how calculations work within the stabilizers framework are performed. As the name suggests, the three-qubit stabilizer code has three physical qubits as the underlying system. Its Hilbert space is therefore $(\mathbb{C}^2)^{\otimes 3}$. Its stabilizer group S is

$$S = \{III, ZZI, IZZ, ZIZ\}. \tag{1.23}$$

A possible choice for the Hamiltonian is:

$$H_3 = -ZZI - IZZ \tag{1.24}$$

The groundspace can be explicitly computed by starting in the $|+++\rangle$ state and projecting onto the $+1$ eigenspace of ZZI and IZZ, since these two operators generate S.

$$\begin{aligned} |\phi_{GS}\rangle &= \frac{1}{\sqrt{4}}(III + ZZI)(III + IZZ)\,|+++\rangle = \frac{1}{\sqrt{4}}(III + ZZI)(|+++\rangle + |+--\rangle) \\ &= \frac{1}{\sqrt{4}}(|+++\rangle + |+--\rangle + |--+\rangle + |-+-\rangle) = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle) \end{aligned} \tag{1.25}$$

According to Theorem 1, the groundspace is degenerate since there are three physical degrees of freedom, but only two independent stabilizers. Indeed, applying ZZZ onto $|\phi_{GS}\rangle$ yields an orthogonal state that also satisfies $s\,|\phi'_{GS}\rangle = |\phi'_{GS}\rangle$ for all s. Therefore, the groundspace of the three-qubit code is given by:

$$GS_3 = span\{|+++\rangle, |---\rangle\} \tag{1.26}$$

This code space is protected against single-qubit flip errors. Assume that the state before the error was $|\phi_{GS}\rangle$ as computed above. An XII error will have the following effect.:

$$XII\,|\phi_{GS}\rangle = \frac{1}{\sqrt{2}}XII(|000\rangle + |111\rangle) = \frac{1}{\sqrt{2}}(|100\rangle + |011\rangle) \tag{1.27}$$

The expectation values of a measurement of the stabilizers in that state are:

$$\begin{aligned} &\frac{1}{2}(\langle 100| + \langle 011|)ZZI(|100\rangle + |011\rangle) = -1 \\ &\frac{1}{2}(\langle 100| + \langle 011|)IZZ(|100\rangle + |011\rangle) = 1 \end{aligned} \tag{1.28}$$

In this case, the measurement would reveal that an X error has occurred on the first qubit. Single Z operators are already logical operators for the three-qubit code. Therefore, an unwanted ZII operation will not be detectable. X errors on multiple sites are also not correctable. That is, because XXX is a logical operator on the three-qubit code. After an XXI error, the syndrome measurement returns the same outcome as an IIX error would. Hence, the "correction" of that error would lead to an unwanted logical operation.
As explored in this section, stabilizer codes protect their information by realizing computations in the symmetry protected groundspace $\mathcal{C}$. The symmetries of $\mathcal{C}$ form a group called the stabilizer group S. The dimension of $\mathcal{C}$ is directly related to the number of independent generators of S, see Theorem 1.

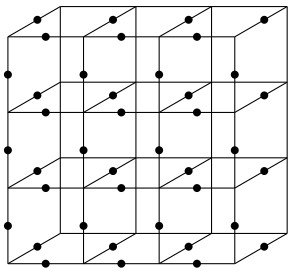


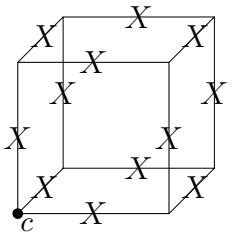
Figure 1.5: The physical degrees of freedom are represented as dots on the cubic lattice. For clarity, they are shown only on the front layer.

1.4 Introducing the Z_2 X-Cube Model

The X-Cube model, introduced in [15], is a stabilizer model defined on a cubic $N_x \times N_y \times N_z$ lattice with periodic boundaries. One physical qubit is located at each edge of the lattice as depicted in Fig. 1.5. To count the number of physical degrees of freedom, three edges can be assigned to each vertex. The Hilbert space of the model is therefore $(\mathbb{C}^2)^{\otimes(3N_xN_yN_z)}$. The Hamiltonian of the X-Cube model is defined as:

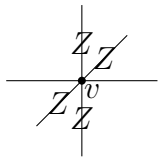
$$H = -\sum_c A_c - \sum_{v,i} B_{v,i}, \tag{1.29}$$

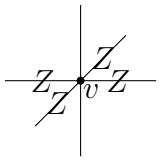
where A_c is defined as:

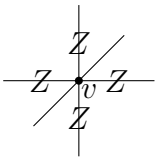
$A_c =$ 

(1.30)

and c runs over all cubes, and $B_{v,i}, i \in \{x, y, z\}$ are defined by:

$B_{v,x} =$ 

, $B_{v,z} =$ 

, $B_{v,y} =$ 

. (1.31)

Note, that the local Hilbert space $\mathbb{C}^2$ is isomorphic to span$\{|g\rangle, g \in Z_2\}$. Hence, we can view the X and Z operators in the A and B terms as representations of the cyclic group Z_2 and its character group, which is isomorphic to Z_2 such that

$$X_g Z_\chi = \chi(g) Z_\chi X_g. \tag{1.32}$$

More specifically, X is the regular representation of Z_2, which acts by left multiplication, so that

$$X_g \left|h\right\rangle = \left|gh\right\rangle . \tag{1.33}$$

In computations, the additive notation $X_g \left|h\right\rangle = \left|g \oplus h\right\rangle$ will be used. Z_χ is a representation of the character group of Z_2. Its action is defined by

$$Z_\chi \left|h\right\rangle = \chi(h) \left|h\right\rangle , \chi(e) = 1, \chi(g) = -1. \tag{1.34}$$

The X-Cube Hamiltonian consists of mutually commuting terms. The A terms trivially commute with each other, as the B terms do among themselves. The A terms also trivially commute with B terms supported at non-overlapping sites. The overlapping A and B terms commute because they always overlap on two sites and $[X \otimes X, Z \otimes Z] = 0$. Consequently, the Hamiltonian terms, together with the identity, form an abelian subgroup of the $3N_xN_yN_z$ folded Pauli group.
Therefore, the X-Cube model is a stabilizer code. The stabilizer group S is generated by the Hamiltonian terms:

$$S = \langle A_c, B_{v,i} \rangle. \tag{1.35}$$

A ground state of the X-Cube Hamiltonian must be in the $+1$ eigenspace of each stabilizer. that is

$$A \left|\varphi\right\rangle = \left|\varphi\right\rangle = B \left|\varphi\right\rangle . \tag{1.36}$$

The degeneracy of the groundspace can be obtained by counting the number of physical degrees of freedom and the number of independent stabilizers, as described in Section 1.3. Accoring to Theorem 1 the groundspace degeneracy is given by $\log_2 \mathrm{GSD} = n - k$, where n are the physical qubit and k the number of generators of the stabilizer. Here, the number of physical degrees of freedom is $n = 3N_xN_yN_z$. Three vertex stabilizers and one cube stabilizer can be associated to each vertex, therefore, there are a total of $4N_xN_yN_z$ stabilizers. Depending on the topology of the lattice, however, there are a number of constraints between these stabilizers. For periodic boundaries, the following constraints hold.
First, the product of all cube stabilizers in any plane P_i in the dual lattice perpendicular to the i-th coordinate direction is the identity.

$$\forall P_i \prod_{c \in P_i} A_c = \mathbb{1}. \tag{1.37}$$

There is one such constraint for each of the $N_x + N_y + N_z$ planes. However, three of those

constraints are not independent themselves, as

$$\prod_{P_x} \prod_{c \in P_x} A_c = \prod_{P_y} \prod_{c \in P_y} A_c = \prod_{P_z} \prod_{c \in P_z} A_c. \tag{1.38}$$

On every vertex v, the product of the three stabilizers $B_{v,i}$ is the identity:

$$\forall v \prod_i B_{v,i} = \mathbb{1}. \tag{1.39}$$

There are $N_x N_y N_z$ such constraints. Lastly, the product of all i-oriented vertex terms for all planes on the lattice, perpendicular to the i-direction is the identity:

$$\forall P_i \prod_{v \in i} B_{v,i} = \mathbb{1}. \tag{1.40}$$

There is one such constraint per plane. Note that the constraints $\prod_c A_c = \mathbb{1}$ and $\prod_{v,i} B_{v,i} = \mathbb{1}$ are already implied by the listed ones.
Taking these constraints into account, S can be generated by $3N_xN_yN_z - 2(N_x + N_y + N_z) + 3$ stabilizers. Therefore, the ground state degeneracy of the X-Cube model on the three-torus is:

$$\log_2 \mathrm{GSD} = 2(N_x + N_y + N_z) - 3, \tag{1.41}$$

which is in agreement with the literature [15].

2 X-Cube Tensor Network description

The goal of this chapter is to obtain the tensor network description of the X-Cube model and to analyze it with respect to its symmetries. These will be necessary to understand how the groundspace degeneracy can be understood purely from the tensor network.

2.1 Modified X-Cube Hamiltonian and Construction of the X-Cube Tensor

In order to obtain a tensor network description of the XCube model, we apply local isometries that double the physical degrees of freedom. Although this procedure modifies the Hamiltonian, it does so only locally and leaves global properties of the model, such as the GSD and the presence of topological excitations, unchanged.
On every edge, an additional physical qubit is added at $|0\rangle$. The resulting state is again a stabilizer state. To obtain the stabilizer group of the new state, one new Z stabilizer is added to the old stabilizer group. The qubits are added translationally invariantly, as depicted in Fig. 2.1 on the left. One added Z term is presented on the right side of that figure.

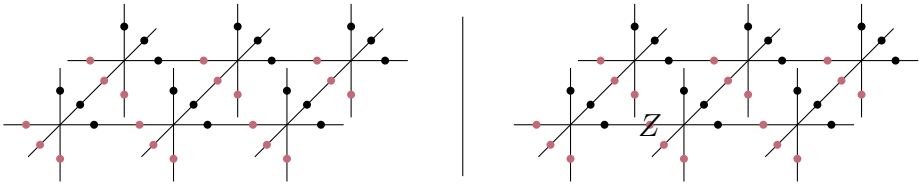


Figure 2.1: The left panel illustrates how the added qubits (depicted in red) are incorporated into the model. The right panel shows, in addition, a single Z term added to the Hamiltonian.

Then, a CX gate is applied, originating from the original qubit and targeting the added qubit on every edge, shown in Fig. 2.2. The action of the controlled X gate is given by

$$CX\,|a,b\rangle = |a, a \oplus b\rangle\,, \quad a,b \in \{0,1\}. \tag{2.1}$$

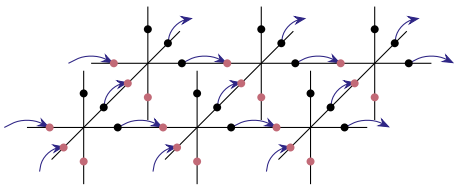


Figure 2.2: The action of CX gates is indicated by blue arrows pointing toward the target qubit. Gates in the z-direction are omitted for clarity.

$\oplus$ denotes addition mod 2. The choice on which side the new qubits are added does not change the result of the procedure in the end. Before acting with the CX gates, the modified stabilizer is spanned by all deformed A'_c and $B'_{v,i}$ terms as well as the single Z terms for every edge, denoted by $E'_{v,i}$. This intermediate stabilizer after doubling, but before applying the CX gates is denoted by S'

$$S' = \langle A'_c, B'_{v,i}, E'_{v,i} \rangle \tag{2.2}$$

After applying the CX gates, the state $CX\,|\varphi\rangle$ is stabilized by the conjugated stabilizer $CXS'CX^\dagger$, because

$$S''CX\,|\varphi\rangle = (CXS'CX^\dagger)CX\,|\varphi\rangle = CX\,|\varphi\rangle\,. \tag{2.3}$$

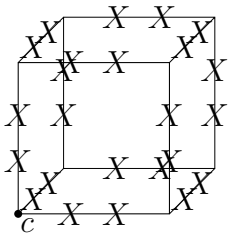
The new stabilizer S'' is calculated using

$$CX(\mathbb{1} \otimes Z)CX^\dagger = (Z \otimes Z^\dagger), \tag{2.4}$$

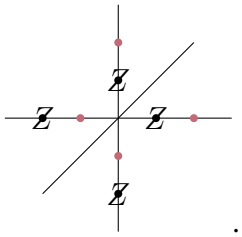
$$CX(Z \otimes \mathbb{1})CX^\dagger = (Z \otimes \mathbb{1}), \tag{2.5}$$

$$CX(X \otimes \mathbb{1})CX^\dagger = (X \otimes X). \tag{2.6}$$

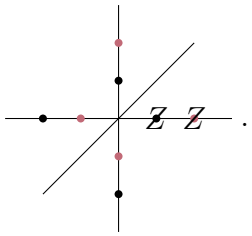
The new cube stabilizers A''_c act with an X on all 24 sites around a cube:

$A_c =$  . (2.7)

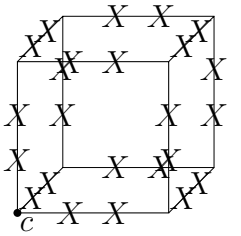
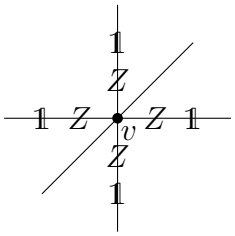
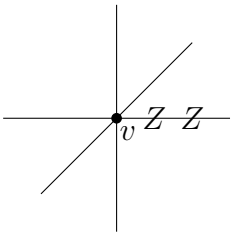
Vertex terms appear deformed, but are unaffected by the action of the CX gates:

$B''_{v,i} =$  . (2.8)

Single site Z stabilizers become $Z \otimes Z$ edge stabilizers $E''_{v,i}$:

$E''_{v,i} =$  . (2.9)

A new equivalent set of generators of S'' is obtained by multiplying each $B''_{v,i}$ with appropriate edge stabilizers, so that

$\tilde{A}_c =$  , $\tilde{B}_{v,i} =$  , $\tilde{E}_{v,i} =$  (2.10)

generate S''. Hence, the stabilized subspace of S'' is the groundspace of:

$$\tilde{H} = -\sum_c \tilde{A}_c - \sum_{v,i} \tilde{B}_{v,i} - \sum_{v,i} \tilde{E}_{v,i}. \tag{2.11}$$

The procedure described above acts only locally on each edge. The addition of qubits and application of CX gates afterwards implements the map

$$V = |00\rangle\langle 0| + |11\rangle\langle 1|\,, \tag{2.12}$$

which is an isometry. Therefore, global properties of the model, such as the ground state degeneracy, remain unchanged. For the ground state degeneracy, this can be verified easily using the same method as before. The number of physical degrees of freedom was doubled to: $n = 6N_xN_yN_z$. At the same time a total of $n/2$ new edge stabilizers were

introduced. The constraints for generating stabilizers on periodic boundary conditions are slightly altered in one case. Eq. (1.39) becomes

$$\forall P_i \prod_{v \in P_i, i} \tilde{B}_{v,i} = \prod_{v \in P_i, i} E_{v,i}. \tag{2.13}$$

As expected, the modified Hamiltonian has the same groundspace degeneracy.

$$\log_2 \text{GSD} = 6N_xN_yN_z - (6N_xN_yN_z - 2(N_x + N_y + N_z) - 3) = 2(N_x + N_y + N_z) - 3 \tag{2.14}$$

From this modified Hamiltonian, a tensor network representation of the ground state of the system is obtained in a similar way as described in [2]. Any ground state of the system can be written as starting from a generic state and projecting it onto the $+1$ eigenspace of all stabilizers.

$$|\phi_{\text{GS}}\rangle = \prod_{v,i} \frac{1}{2}(\mathbb{1} + \tilde{E})_{v,i} \prod_{v,i} \frac{1}{2}(\mathbb{1} + \tilde{B})_{v,i} \prod_{c} \frac{1}{2}(\mathbb{1} + \tilde{A})_{c} \, |\varphi\rangle \, . \tag{2.15}$$

Let us choose the starting state φ as $|+...+\rangle$. This is a $+1$ eigenstate of all cube stabilizers. Consequently, all cube stabilizers can be trivially applied, and thus

$$|\phi_{\text{GS}}\rangle = \prod_{v,i} \frac{1}{2}(\mathbb{1} + \tilde{E})_{v,i} \prod_{v,i} \frac{1}{2}(\mathbb{1} + \tilde{B})_{v,i} \, |+...+\rangle \, . \tag{2.16}$$

At every edge, the projection onto the edge stabilizer acts by:

$$\frac{1}{2}(\mathbb{1} + \tilde{E}) \, |++\rangle = \frac{1}{2}(|++\rangle + |--\rangle) = \frac{1}{2}(H \otimes H)(|00\rangle + |11\rangle) = \frac{1}{2} \sum_{i=0}^{1} |ii\rangle \, , \tag{2.17}$$

where the Hadamard gate H, defined as

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{2.18}$$

was used. At every vertex, the projections onto the $+1$ eigenspace of $B_{v,i}$ act on one part

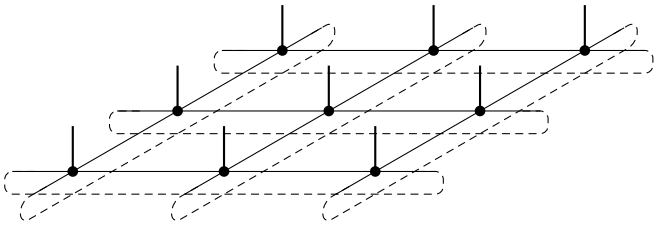


Figure 2.3: This two-dimensional illustration demonstrates how the ground state of the X-Cube model can be obtained by contracting tensors along all connected edges. In three dimensions, each tensor is connected to six neighboring tensors via virtual bonds. The physical degrees of freedom are grouped into a single edge per tensor.

of each of the six connected edges. Thus, the tensor placed on every vertex is

$$\begin{aligned} T &= \sum_{i_1,i_2,i_3,i_4,i_5,i_6=0}^{1} \prod_i (\mathbb{1} + \tilde{B})_{v,i} \left|i_1,i_2,i_3,i_4,i_5,i_6\right\rangle \otimes \left\langle i_1,i_2,i_3,i_4,i_5,i_6\right| \\ &= \sum_{i_1,i_2,i_3,i_4,i_5,i_6=0}^{1} P \left|i_1,i_2,i_3,i_4,i_5,i_6\right\rangle \otimes \left\langle i_1,i_2,i_3,i_4,i_5,i_6\right| , \end{aligned} \tag{2.19}$$

where P is

$$P = \frac{1}{4} \sum_{k,j=0}^{1} (\mathbb{1}^{\otimes 2} \otimes Z^k \otimes Z^k \otimes Z^k \otimes Z^k) \times (Z^j \otimes Z^j \otimes Z^j \otimes Z^j \otimes \mathbb{1}^{\otimes 2}). \tag{2.20}$$

The labeling of the indices is illustrated in Fig. 1.2. Only two projectors are applied in the second line, since the third one is the product of the other two. The contraction of the tensor network of the tensors T will represent the ground state of $\tilde{H}$. Fig. 2.3 presents a two-dimensional illustration of how the contraction results in a physical state, similar to the one-dimensional case shown in Section 1.2. In this section, we have obtained a tensor network description of the X-Cube model. We have done so by implementing local isomorphisms that double the degree of freedom on every edge.

Table 2.1: Symmetry generators of the tensor T: Each row is to be read as an operator acting on all physical and all virtual degrees of the tensor. They are ordered in the same way as in equation Eq. (2.21). Daggers are omitted as $X^\dagger = X$.

physical						*virtual*					
Z	Z	Z	Z	I	I	I	I	I	I	I	I
I	I	Z	Z	Z	Z	I	I	I	I	I	I
X	X	I	I	I	I	X	X	I	I	I	I
I	I	X	X	I	I	I	I	X	X	I	I
I	I	I	I	X	X	I	I	I	I	X	X
X	I	X	I	X	I	X	I	X	I	X	I
Z	I	I	I	I	I	Z	I	I	I	I	I
I	Z	I	I	I	I	I	Z	I	I	I	I
I	I	Z	I	I	I	I	I	Z	I	I	I
I	I	I	Z	I	I	I	I	I	Z	I	I
I	I	I	I	Z	I	I	I	I	I	Z	I
I	I	I	I	I	Z	I	I	I	I	I	Z

2.2 Symmetries of the X-Cube Tensor

In the following section, we will analyze the symmetries of T. For simplicity, the notation is written in a more compact form:

$$\begin{aligned} T &= \sum_{i_1,i_2,i_3,i_4,i_5,i_6=0}^{1} P\,|i_1,i_2,i_3,i_4,i_5,i_6\rangle \otimes \langle i_1,i_2,i_3,i_4,i_5,i_6| \\ &= \sum_i P\,|i\rangle\langle i|\,. \end{aligned} \tag{2.21}$$

A symmetry of the tensor is an operator whose action leaves the tensor invariant. The tensor admits twelve independent symmetries. They are listed in Table 2.1. Three examples of symmetries are calculated below. All other symmetries listed follow by analogous computations.

The first example of a symmetry of T is $ZZZZII$ acting on the physical level. This is the first row presented in Table 2.1. It is a symmetry because

$$\begin{aligned} &\sum_i (Z\otimes Z\otimes Z\otimes Z\otimes I\otimes I) P\,|i\rangle\langle i| = \\ &\sum_i P\,|i\rangle\langle i|\,. \end{aligned} \tag{2.22}$$

Here, we used:

$$(Z \otimes Z \otimes Z \otimes Z \otimes I \otimes I)P = P. \tag{2.23}$$

The second type symmetry in the table is acting with $XXIIII$ on the physical level and $XXIIII$ on the virtual level.

$$\begin{aligned} &\sum_i (X \otimes X \otimes I \otimes I \otimes I \otimes I)P \left|i\right\rangle\!\left\langle i\right| (X \otimes X \otimes I \otimes I \otimes I \otimes I) = \\ &\sum_i P(X \otimes X \otimes I \otimes I \otimes I \otimes I) \left|i\right\rangle\!\left\langle i\right| (X \otimes X \otimes I \otimes I \otimes I \otimes I) = \\ &\sum_i P \left|i\right\rangle\!\left\langle i\right| . \end{aligned} \tag{2.24}$$

The last equality holds, because

$$\sum_i (X \otimes X \otimes I \otimes I \otimes I \otimes I) \left|i\right\rangle\!\left\langle i\right| (X \otimes X \otimes I \otimes I \otimes I \otimes I) = \sum_i \left|i\right\rangle\!\left\langle i\right| . \tag{2.25}$$

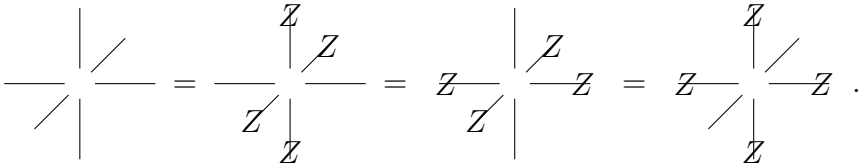
The third kind of symmetry of the tensor is the action of a single physical Z and a single virtual Z on the corresponding virtual degree of freedom. Similarly to the calculation above,

$$\begin{aligned} &\sum_i (Z \otimes I \otimes I \otimes I \otimes I \otimes I)P \left|i\right\rangle\!\left\langle i\right| (Z \otimes I \otimes I \otimes I \otimes I \otimes I) = \\ &\sum_i P(Z \otimes I \otimes I \otimes I \otimes I \otimes I) \left|i\right\rangle\!\left\langle i\right| (Z \otimes I \otimes I \otimes I \otimes I \otimes I) = \\ &\sum_i P \left|i\right\rangle\!\left\langle i\right| . \end{aligned} \tag{2.26}$$

Here, in the last step,

$$Z \left|i_1\right\rangle\!\left\langle i_1\right| Z = (-1)^{i_1+i_1} \left|i_1\right\rangle\!\left\langle i_1\right| = \left|i_1\right\rangle\!\left\langle i_1\right| \tag{2.27}$$

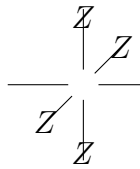
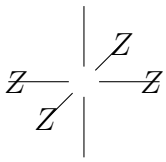
was used. Of all generating symmetries in Table 2.1, two can be written as an action purely on the virtual level. They generate the virtual symmetries:



(2.28)

These symmetries are the ones in which we are mainly interested. The other symmetries describe the encoding of physical degrees of freedom in the virtual ones. For example,

the last six symmetries tell us that a physical Z is equivalent to a virtual Z.
Note that the symmetries of Eq. (2.28) form a group, which will be denoted as G_{111}, where the indices indicate the size of the tensor network. The elements of this group are tensor products of regular representations of Z_2, the cyclic group. The operators

$L_x =$  $, L_z =$  (2.29)

are chosen as generators for the group:

$$G_{111} = \{(L_z^0, L_x^0), (L_z^0, L_x^1), (L_z^1, L_x^0), (L_z^1, L_x^1)\}. \tag{2.30}$$

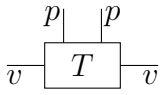
Since L_x and L_z are faithful representations of Z_2 each,

$$G_{111} \cong Z_2 \times Z_2. \tag{2.31}$$

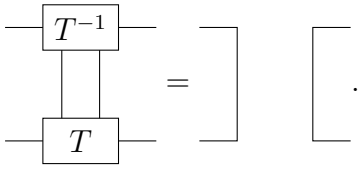
In summary, the single X-Cube tensor has G_{111} as its symmetry group. Acting with any $g \in G_{111}$ on the incoming virtual indices of T leaves T invariant. Each element $g \in G_{111}$ can be generated by the loop symmetries L_x and L_z.

2.3 Pseudo-Invertibility of T

Let us consider the PEPS tensor as a map from the virtual to the physical level. We are interested in the inverse of T on the invariant subspace of P. For that, we represent the tensor in a two-dimensional graphic notation as:

$T =$  (2.32)

An inverse of T would be of the form:

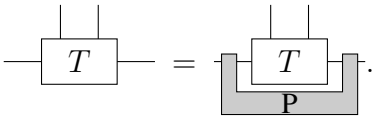
 (2.33)

However, since T is invariant under certain symmetries acting only on the virtual level, an inverse on the entire domain of T cannot exist. Instead, TT^{-1} can only act as the identity

on the range of P, where $P = 1/|G| \sum_{g \in G_{111}} g$. As $G \cong Z_2 \times Z_2$, we can write P as

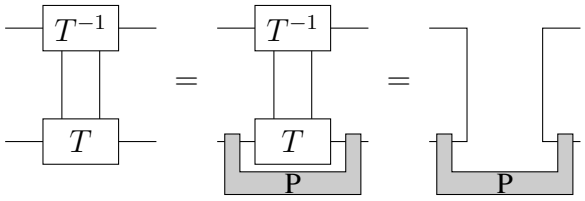
$$P = \frac{1}{4} \sum_{g,h \in \{\mathbb{1}, Z\}}$$. (2.34)

In the two-dimensional picture introduced above this invariance is



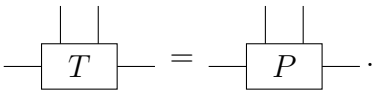
. (2.35)

The following equation illustrates why the inverse of T could only be defined on the subspace spanned by P. Assume that we have an inverse on the entire domain of T. Since T is invariant under P we can insert it before the contraction with T^{-1}. In this best case scenario, the resulting inverse is P, which is the identity on the subspace of P.



(2.36)

It turns out that the left pseudo-inverse of T is $T^\dagger$. This follows immediately from



. (2.37)

In formula, this is written as:

$$\begin{aligned} \mathrm{Tr}\{TT^\dagger\} &= \sum_{j,k} \mathrm{Tr}\{P\,|j\rangle\,\langle k|\,P\}\,\langle j| \otimes |k\rangle \\ &= \sum_{j,k} \langle k|\,PP\,|j\rangle \times |k\rangle\langle j| \\ &= \sum_{j,k} \langle k|\,P\,|j\rangle \times |k\rangle\langle j| \\ &= P, \end{aligned} \tag{2.38}$$

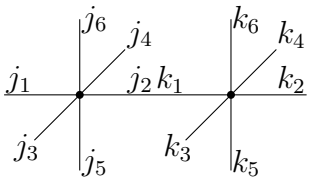
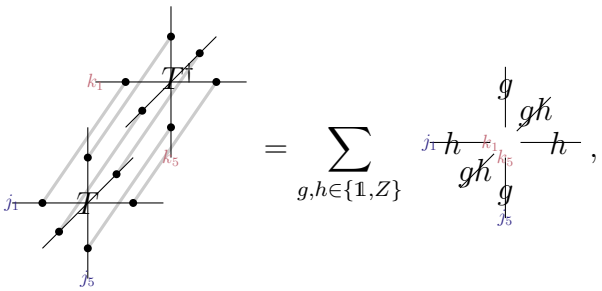


Figure 2.4: $2 \times 1 \times 1$ tensor blocking. The graph shows how the indices of the two tensors match together. The connected index j_2 and k_1 will be summed over.

where we used the projector property $P^2 = P$. In the three-dimensional picture this reads as



(2.39)

where the matching index from left to right is indicated by small indices j and k.

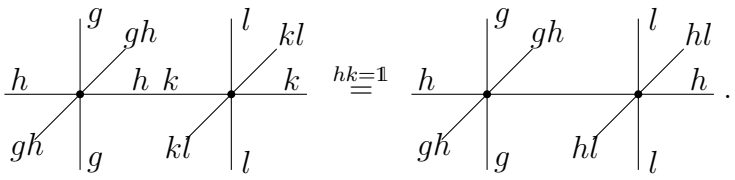
2.4 Blockings of T

In the previous sections, the symmetries and pseudo-invertibility of a single tensor were considered. This section expands this discussion to larger regions of the tensor network. In order to understand larger patches of Ts, let us start by considering two tensors contracted along one edge, as depicted in Fig. 2.4.
The edge connecting the two tensors represents a contraction of that virtual degree of freedom. In the computation, this trace yields a δ_{j_2,k_1}.

$$\begin{aligned} T_{211} &= \sum_{j,k} P\left|j\right\rangle P\left|k\right\rangle \otimes \mathrm{Tr}\left\{ \sum_i \left\langle k_1 | i\right\rangle \left\langle j_2 | i\right\rangle \right\} \times \left\langle j_1, j_3, j_4, j_5, j_6\right| \left\langle k_2, k_3, k_4, k_5, k_6\right| \\ &= \sum_{j,k} \delta_{j_2,k_1} P\left|j\right\rangle P\left|k\right\rangle \otimes \left\langle j_1, j_3, j_4, j_5, j_6\right| \left\langle k_2, k_3, k_4, k_5, k_6\right| \end{aligned} \tag{2.40}$$

The $\sum_i \left|i, i\right\rangle$ appearing in the trace is the maximally entangled pair state used to contract the two tensors, as described in Section 1.2.
Similarly to the single tensor case, we are mainly interested in symmetries acting on the virtual degrees of freedom.

These symmetries can be obtained by repeating the idea of Section 2.2. However, the virtual symmetries of T_{211} can also be obtained in a different way, that will help in the generalization to arbitrarily sized patches. As established in the previous sections, T is invariant under the action of P. We consider the $2 \times 1 \times 1$ blocking and apply P on both tensors individually. The symmetries supported only on the free virtual edges are obtained by constraining to the case where the symmetries are only acting as the identity on the contracted edge. After applying P to both tensors, any element in the sum can be constrained as follows:

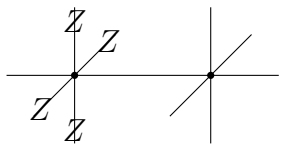


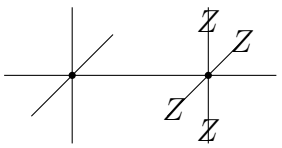
(2.41)

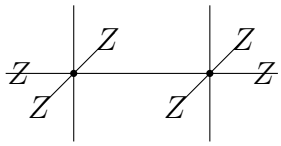
Note that $hk = \mathbb{1} \implies h = k$ for $h, k \in \{\mathbb{1}, Z\}$. Effectively, this condition leads to the merging of two generating loop symmetries.
Let G_{211} denote the symmetry group of the $2 \times 1 \times 1$ tensor network. It consists of eight elements, which are the eight possible configurations of $g, h, l \in \{\mathbb{1}, Z\}$ in Eq. (2.41). Note that G_{211} is generated by three symmetry operators, each acting on edges around a closed loop. Due to the faithfulness of the L_i representations,

$$G_{211} = \langle L_{x_1}, L_{x_2}, L_z \rangle \cong Z_2 \times Z_2 \times Z_2, \tag{2.42}$$

where

$L_{x_1} =$ 

, $L_{x_2} =$ 

, $L_z =$ 

.

(2.43)

The invertibility of T_{211} is subject to the same restrictive consideration as before. The invariance of T_{211} under the action of $P_{211} = 1/8 \sum_{g \in G_{211}} g$ implies that only an inverse

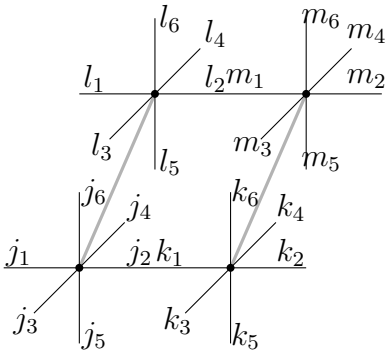
on the P_{211} subspace can be obtained. Again, the inverse is the complex conjugate of T_{211}.

$$\begin{aligned}\mathrm{Tr}\Big\{T_{211}T_{211}^{\dagger}\Big\}_p &= \sum_{j,k,l,m} \delta_{j_2,k_1}\delta_{l_2,m_1}\,\mathrm{Tr}\{P\,|j\rangle\langle l|\,P\}\times\mathrm{Tr}\{P\,|k\rangle\langle m|\,P\}\\ &\quad\times|l_1,l_3,l_4,l_5,l_6\rangle\langle j_1,j_3,j_4,j_5,j_6|\,|m_2,m_3,m_4,m_5,m_6\rangle\langle k_2,k_3,k_4,k_5,k_6|\\ &= \sum_{j,k,l,m} \delta_{j_2,k_1}\delta_{l_2,m_1}\,\langle l|\,P\,|j\rangle\times\langle m|\,P\,|k\rangle\\ &\quad\times|l_1,l_3,l_4,l_5,l_6\rangle\langle j_1,j_3,j_4,j_5,j_6|\,|m_2,m_3,m_4,m_5,m_6\rangle\langle k_2,k_3,k_4,k_5,k_6|\\ &= P_{211}.\end{aligned} \tag{2.44}$$

In the last step, we have used that the solutions of

$$\begin{aligned}P\,|j\rangle &= |l\rangle\\ P\,|k\rangle &= |m\rangle\end{aligned} \tag{2.45}$$

are constrained by the deltas. The action on the contracted edge between the two tensors, is constrained to the identity. With the goal of generalizing to arbitrary system sizes in mind, it is useful to think of this computation as an inversion of each tensor individually followed by a trace of the contracted edge:



$$= \frac{1}{16}\sum_{g,h,f,k\in\{\mathbb{1},Z\}} \cancel{h}\;\Big|_{\cancel{gh}}^{g}\;\cancel{h}\;\;\cancel{k}\;\Big|_{\cancel{fk}}^{f}\;\cancel{k}. \tag{2.46}$$

The procedure of blocking the tensors, finding the corresponding symmetry group $G_{N_xN_yN_z}$ and obtaining an inverse on the respective subspace can be repeated for any system size. However, it is not immediately clear how $G_{N_xN_yN_z}$ will scale with system size. In order to write down a recipe to generate $G_{N_xN_yN_z}$, let us first define a generalization of the L_{x_n} generators used for G_{111} and G_{211}.

Definition 3. *Let x_n denote the plane perpendicular to the x-direction with x coordinate*

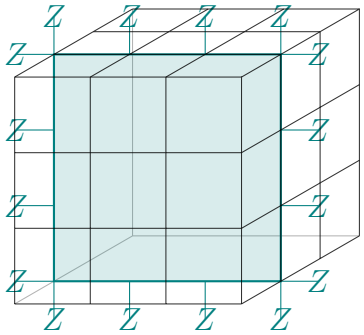


Figure 2.5: For a fixed plane y_2, the operator L_{y_2} acts on all non-contracted edges within that plane with an Z.

n. Then:

$$L_{x_n} = \bigotimes_{i \in x_n} Z_i \tag{2.47}$$

is the loop operator acting with an Z on all free edges, indexed by i, in the x_n plane, as illustrated in Fig. 2.5.

L_{z_n} and L_{y_n} are defined equivalently. With those we can now define the symmetry group of an arbitrarily sized tensor network of T.

Theorem 2. *Let $G_{N_x N_y N_z}$ be the symmetry group of an $N_x \times N_y \times N_z$ tensor network of X-Cube tensors T. Then:*

$$G_{N_x N_y N_z} = \langle L_{x_1}, L_{x_2}, ..., L_{x_{N_x}}, L_{z_1}, L_{z_2}, ..., L_{z_{N_z}}, L_{y_1}, L_{y_2}, ..., L_{N_y - 1} \rangle, \tag{2.48}$$

where L are loop operators consisting of tensor products of X as defined above.

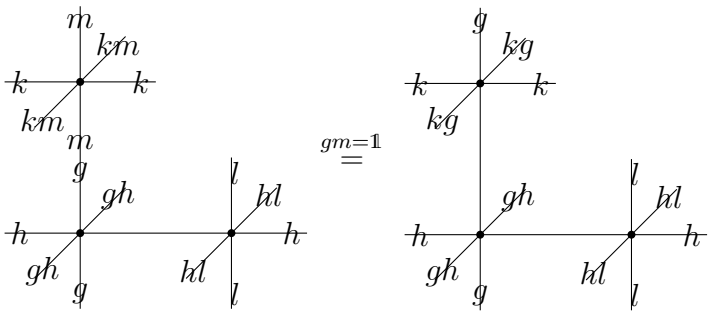
In y-direction one less generator is chosen because the set with $N_x + N_y + N_z$ generators would be over-complete, since

$$\prod_i^{N_x} L_{x_i} \prod_i^{N_y} L_{y_i} \prod_i^{N_z} L_{z_i} = \mathbb{1}. \tag{2.49}$$

Before presenting the proof, we provide some intuition for the content of Theorem 2. Previously, we discussed how the symmetry group of a single tensor is generated by loop symmetries L_x and L_z. We then extended this analysis to a $2 \times 1 \times 1$ patch of the tensor network, where the symmetry group G_{211} is generated by loop symmetries L_{x_1}, L_{x_2} and L_z. Theorem 2 generalized this result to arbitrarily sized networks. The symmetry group always consists of loop symmetries of the form Fig. 2.5 as well as products of such loop operators.

Proof. For G_{111} and G_{211} this theorem holds, see Eq. (2.30) and Eq. (2.42). The proof of this theorem for arbitrary system size is divided into three steps. Starting from the $1 \times 1 \times 1$ case, the theorem is first proven for systems of size $N_x \times 1 \times 1$. Next, the proof is expanded to systems of size $N_x \times 1 \times N_z$ and finally to arbitrary system sizes.
By repeating the procedure used to acquire G_{211} from G_{111}, it is straightforward to obtain the symmetry group $G_{N_x 11}$, which is generated by $L_{x_1}, L_{x_2} ... L_{x_{N_x}}, L_z$. Although the length of the calculations increases, the principle applied remains unchanged. Starting from the $1 \times 1 \times 1$ case, we add $N_x - 1$ tensors in x-direction one after the other. After every added tensor, the symmetry group is generated by one extra loop symmetry L_{x_i}. That is, because every tensor added to the blocking will come with two symmetry generators, chosen to be L_x^a and L_z^a. The contraction will constrain one of them, as in Eq. (2.41), effectively merging the L_z^a symmetry with the L_z loop symmetry of the tensor network before the addition of that new tensor. The L_x^a symmetry remains unconstrained and therefore acts as a new generator of the enlarged symmetry group. That proves the $N_x \times 1 \times 1$ case.

The next step is expanding the $N_x \times 1 \times 1$ tensor network in z-direction. First, the addition of one layer will be discussed, which can then be applied $N_z - 1$ times. For the first added tensor, the same construction of merging one loop holds. The following shows how the idea of contractions acting as constraints on added loop symmetries works also for this case:



(2.50)

Only two tensors are presented in the lower level, but this holds independently for any N_x. The symmetry group of this particular network cannot be written as G_{212}, since a corner of the lattice is missing. However, it is isomorphic to G_{212}, as will be shown in the following. After adding this first tensor, the neighboring one is added to the system. In contrast to the addition of new tensors so far, this one is contracted with the existing tensor network along two edges. As the following equation shows, this constrains both

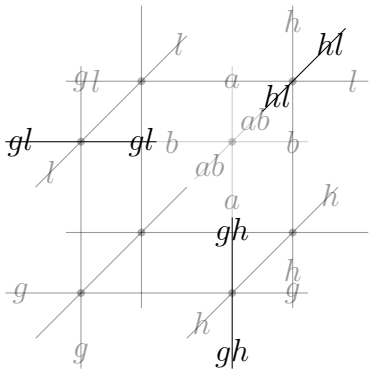
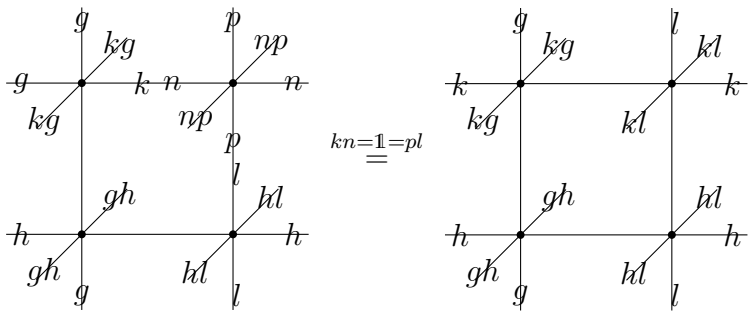


Figure 2.6: The relevant symmetry generators of the existing tensor network are shown. Adding a tensor in the missing position to complete the cube yields $a = gh$, $b = gl$, and $ab = hl$. Any two of these relations imply the third.

symmetries of the newly added tensor to merge with existing ones.



(2.51)

This argument can be repeated until the entire row is filled with tensors. The generating loop symmetries merge, but no additional generators need to be added. This proves the $N_x \times 1 \times 2$ case. Repeating this layer by layer proves the $N_x \times 1 \times N_z$ case.
Similarly to this last step, the addition of tensors is done in y-direction. The tensors are added in a connected way row after row. Every tensor after the first will be constraint by two edges, up until the last tensor in a plane is added. For the final tensor in every layer, three constraints arise from the three contractions. However, one of these constraints is redundant, as any two of these constraints imply the third one. This is shown for the smallest system size where this occurs in Fig. 2.6. To summarize, the addition of a $N_x \times 1 \times N_z$ tensor network layer onto the $N_x \times 1 \times N_z$ tensor network results only in one additional L_{y_1} symmetry. Repeating this layer by layer adds a total of $N_y - 1$ generators to the group. Consequently, Theorem 2 holds for any system size. □

Note that in the graphical computations shown, the labeling was chosen so that the computations did not require any relabeling to make the statements obvious. Choosing generators perpendicular to the direction in which the network is expanded achieves that. A

different choice of generators would then require an extra step of changing labels to arrive at the shown results.
Theorem 2 implies

$$G_{N_x N_y N_z} \cong Z_2 \times Z_2 \times ... \times Z_2 = (Z_2)^{\times(N_x+N_y+N_z-1)} \tag{2.52}$$

and therefore also

$$|G_{N_x N_y N_z}| = 2^{N_x+N_y+N_z-1}. \tag{2.53}$$

For the $1 \times 1 \times 1$ and the $2 \times 1 \times 1$ tensor network we found that the pseudo-inverse of T on the subspace of the corresponding P was $T^\dagger$. The following theorem generalizes this result to arbitrarily sized patches of T.

Theorem 3. *Let $G_{N_x N_y N_z}$ be the symmetry group of an $N_x \times N_y \times N_z$ tensor network of X-Cube tensors T. Denote the blocked tensor of that size by $T_{N_x N_y N_z}$. Then:*

$$T_{N_x N_y N_z} T^\dagger_{N_x N_y N_z} = \frac{1}{|G_{N_x N_y N_z}|} \sum_{g \in G_{N_x N_y N_z}} g \tag{2.54}$$

The proof of this theorem follows the general idea of the proof of Theorem 2.

Proof. Starting with a single tensor, we first expand the network in x-direction to the required size N_x. Then we expand layer by layer in z-direction. Finally, we reach the desired system size by adding layers in y-direction. After every added layer, Theorem 3 holds. The argument why Eq. (2.54) holds is the following: The symmetry group $G_{N_x N_y N_z}$ was obtained by constraining the symmetries of every added tensor so that they merge with the existing symmetries. The merging constraint arises from every contracted edge. Similarly, the inversion of a patch of tensors is achieved by inverting every individual one, then tracing the contracted edges. This trace enforces the merging of loops as seen in Eq. (2.46). Hence, Theorem 3 holds after every step of enlarging the tensor network, which achieves any system size. □

The main achievements of this chapter were the results of Theorem 2 and Theorem 3. These describe the structure of the symmetries of an X-Cube tensor network of arbitrary size. Unless relevant, from now on, the size of the tensor network will no longer be explicitly denoted for G.

2.5 Intersection Property and Closure Property

This section closely follows the approach of [11]. The goal is to prove two properties of the tensor network which will be required to obtain the groundspace degeneracy of a

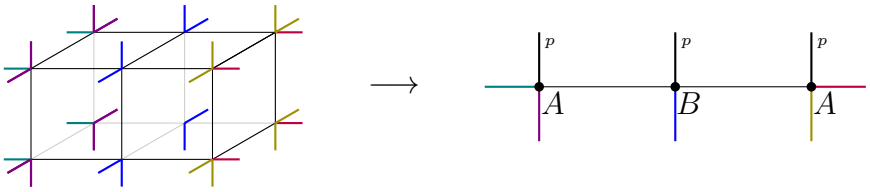
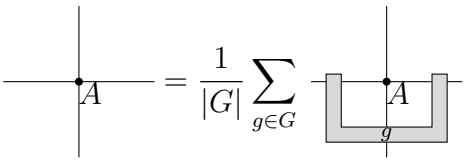


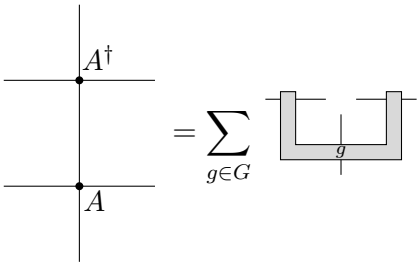
Figure 2.7: This figure illustrates how indices are blocked for the Intersection property proof. The indices that are blocked together on the outside are presented in different colors. The main distinction is between the block on the left A (in this case $1 \times 2 \times 2$) the central B (in this case $1 \times 2 \times 2$) and the added layer A (in this case $1 \times 2 \times 2$).

$N_x \times N_y \times N_z$ tensor network with periodic boundary conditions. The result is expected to be in agreement with the counting approach applied in Section 1.4.
The idea is as follows: A local Hamiltonian is considered, which has the $2 \times 2 \times 2$ X-Cube tensor network as a ground state. From there, a larger system size is considered, whose Hamiltonian is the sum of $2 \times 2 \times 2$ Hamiltonians. The Intersection Property guarantees that the ground state of that larger system is represented by the appropriately sized X-Cube tensor network. Finally, the closure to periodic boundary conditions has degrees of freedom, which are directly related to the groundspace degeneracy of the resulting Hamiltonian. These degrees of freedom are described by the Closure Property.
Before proving the Intersection Property and the Closure Property some properties of the X-Cube tensor will be recalled in a functional two-dimensional graphical notation. Indices of a larger tensor network will be blocked together as illustrated in Fig. 2.7 for a $3 \times 2 \times 2$ blocking size. For larger tensor networks, the blocking is done equivalently, where A's size remains $1 \times N_y \times N_z$ and B is a $(N_x - 2) \times N_y \times N_z$ network. For y and z-direction this is done equivalently.
Theorem 2 established the invariance of an arbitrary sized tensor network under the action of the appropriate $P = 1/|G| \sum_g g$:



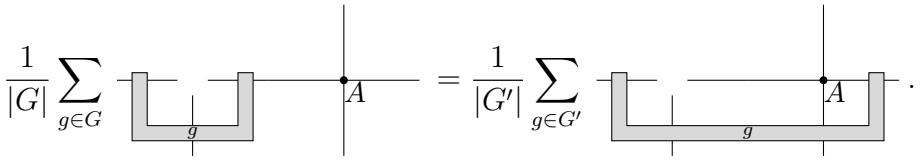
(2.55)

$A^\dagger$ is the pseudo-inverse of A, see Theorem 3:



(2.56)

One more property is needed for the proof. We call it the push through property, since it describes that an X acting on the contracted edges between two patches of the tensor network can be pushed through to the free edges of one of the patches. In graphical notation it reads:



(2.57)

The dimensions of edges around the left "empty" vertex are $N_x \times N_y \times N_z$, the dimensions of A are $N'_x \times N_y \times N_z$. G is the symmetry group corresponding to the $N_x \times N_y \times N_z$ system size, G' the symmetry group of the $N'_x \times N_y \times N_z$ tensor network.
For an arbitrary system size, the argument is as follows. Let P_1 be the operator acting on the left side of Eq. (2.57).

$$P_1 = \frac{1}{|G|} \sum_{g \in G} g \tag{2.58}$$

On the right side, define

$$P_2 = \frac{1}{|G'|} \sum_{\prime \in G'} g'. \tag{2.59}$$

Note, that the system sizes of the two patches must match in x-direction. Any element g of G can be written as

$$g = \prod_{i=1}^{N_x} L_{x_i}^{a_i} \prod_{j=1}^{N_y} L_{y_i}^{b_j} \prod_{k=1}^{N_z} L_{z_i}^{c_k}, \tag{2.60}$$

for some $a \in \{0,1\}^{N_x}, b \in \{0,1\}^{N_y}, c \in \{0,1\}^{N_z}$. Note, that for each sum element on the right of Eq. (2.57), the configuration of Xs acting on the contracted edges between the two blocks is completely determined by

$$\prod_{j=1}^{N_y} L_{y_i}^{b_j} \prod_{k=1}^{N_z} L_{z_i}^{c_k}. \tag{2.61}$$

In order to cancel this contribution of Xs on the connected edge, we apply the symmetry

$$\frac{1}{2^{N'_x}} \sum_{a'_{i'}=0}^{1} \prod_{i'=1}^{N'_x} \tilde{L}_{x'_i}^{a'_{i'}} \prod_{j=1}^{N_y} \tilde{L}_{y_j}^{b_j} \prod_{k=1}^{N_z} \tilde{L}_{z_k}^{c_k}, \tag{2.62}$$

on A. The tilde denotes that the operators act on the edges of A. The summation only runs over the $\tilde{L}_x$ elements in the second block, whereas the $\tilde{L}_y$ and $\tilde{L}_z$ elements are fixed to match the operators acting on the contracted edge.
Doing this for all $g \in G$ results in the following operator acting on the connected blocks:

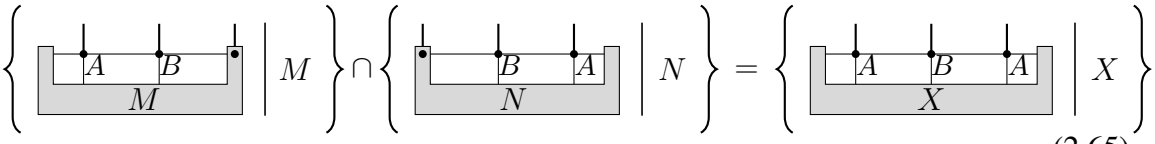
$$\frac{1}{2^{N'_x+N_x+N_y+N_z-1}} \sum_{a,a',b,c=0}^{1} \prod_{i=1}^{N_x} L_{x_i}^{a_i} \prod_{i'=1}^{N'_x} \tilde{L}_{x'_i}^{a'_{i'}} \prod_{j=1}^{N_y} L_{y_j}^{b_j} \prod_{j=1}^{N_y} \tilde{L}_{y_j}^{b_j} \prod_{k=1}^{N_z} L_{z_k}^{c_k} \prod_{k=1}^{N_z} \tilde{L}_{z_k}^{c_k} \tag{2.63}$$

Merging the operators L_{y_j} with their corresponding $\tilde{L}_{y_j}$ gives $\hat{L}_{y_j}$ loop symmetries of the entire system and equivalently for the L_z operators. Then the equation above simplifies to

$$\frac{1}{2^{\hat{N}_x+N_y+N_z-1}} \sum_{\hat{a},b,c=0}^{1} \prod_{i=1}^{\hat{N}_x} \hat{L}_{x_i}^{\hat{a}_i} \prod_{j=1}^{N_y} \hat{L}_{y_j}^{b_j} \prod_{k=1}^{N_z} \hat{L}_{z_i}^{c_k}, \tag{2.64}$$

where $\hat{N}_x = N_x + N'_x$ and $\hat{a} \in \{0,1\}^{\hat{N}_x}$. The operators L_i and $\tilde{L}_i$ were grouped into $\hat{L}_i$. $\hat{P}$ is the operators acting on the right side of Eq. (2.57).
With the two-dimensional notation clarified and all required properties established we can now formally state the Intersection Property.

Theorem 4 (Intersection Property)**.** *Let A and B be X-Cube tensor networks of sizes $1 \times N_x \times N_z$ and $(N_x - 2) \times N_y \times N_z$ respectively. Let M,N and X be arbitrary tensors of dimensions matching those of A and B along the contracted edges as well as appropriate physical dimensions for M and N. Then,*

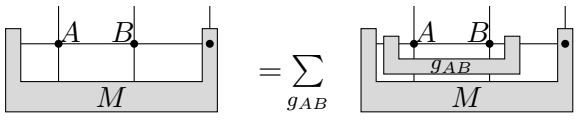


(2.65)

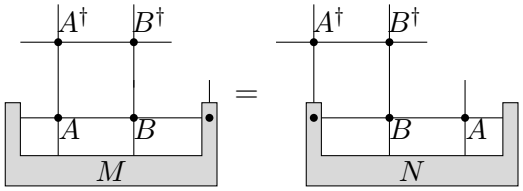
Although the theorem is formulated for the x-direction which is the one connecting A and B, the theorem holds equally for A and B contracted along the y or z-direction.
Intuitively, the Intersection Property ensures the consistency of the tensor network across overlapping boundary regions.

Proof. First note that M, N and X can be assumed to be invariant under the action of the

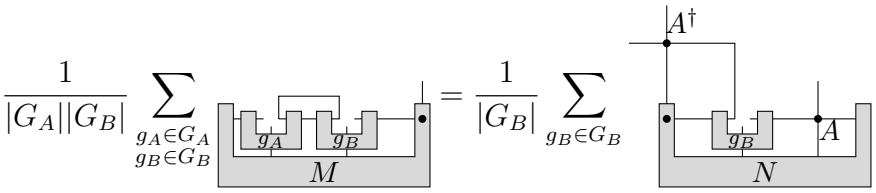
corresponding Ps. We can symmetrize M with the symmetries of M,



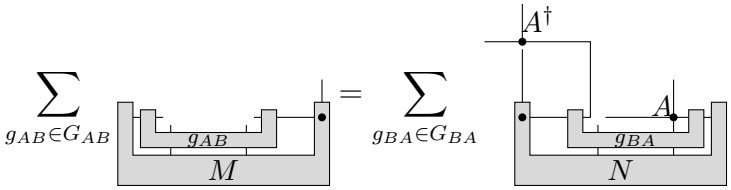
Hence, contracting AB with M is the same as contracting AB with PM, and PM is symmetric. We choose an arbitrary element within the intersection on the left side. Then we apply the pseudo-inverses $A^\dagger$ and $B^\dagger$ on A and B,



. (2.66)

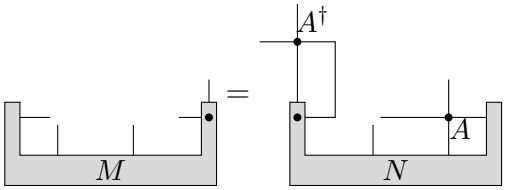


Next, we merge the two symmetries traced over on the left and apply the push through property, see Eq. (2.57), on the right side.



(2.67)

Finally, we use the invariance of M and N under the action of the present symmetries to obtain M on the right. The left side is of the desired form in Theorem 4. As we started with an arbitrary element in the intersection, this proves inclusion of the left side.



(2.68)

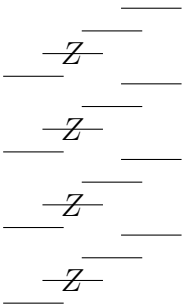
Inclusion in the opposite direction is trivially fulfilled by choosing N and M appropriately for the given X. This concludes the proof of the Intersection Property. □

Since the Closure Property depends on how symmetries act across different directions of the tensor network, we first introduce a decomposition of symmetries into their action along the x, y and z-directions.

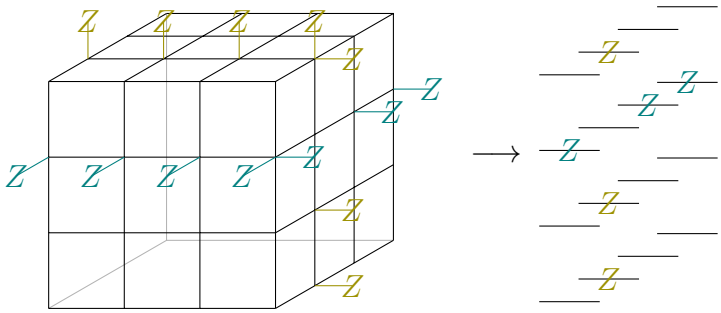
Definition 4. *Let F be an $N_y \times N_z$ array of edges pointing in x-direction, representing $\mathbb{C}^2$ degrees of freedom. Let $l^z_{y_i}$ denote a line in z-direction with y-coordinate i. Then, define the operator $S^F_{l^z_{y_i}}$*

$$S^F_{l^z_{y_i}} = \bigotimes_{i \in l^z_{y_i}} Z_i. \tag{2.69}$$

In graphical notation:

$S^F_{l^z_{y_2}} =$  , (2.70)

where F is a 4×4 array. Similarly, the operator $S^F_{l^y_{z_i}}$ can be defined, as well as equivalent operators acting on arrays of edges in y and z directions.
Note that for every symmetry of a tensor network the operators appearing on one side of the network are products of such operators, as depicted below:

 (2.71)

These products are denoted by E^x:

$$(E^x)^{b_i,c_i} = \prod_i (S^F_{l^y_{y_i}})^{b_i} \prod_i (S^F_{l^z_{z_i}})^{c_i}. \tag{2.72}$$

The number of distinct E^x, E^y and E^z is counted by three quotient groups of G, which will be introduced in the following. In order to do so, let us start by defining their corresponding subgroups:

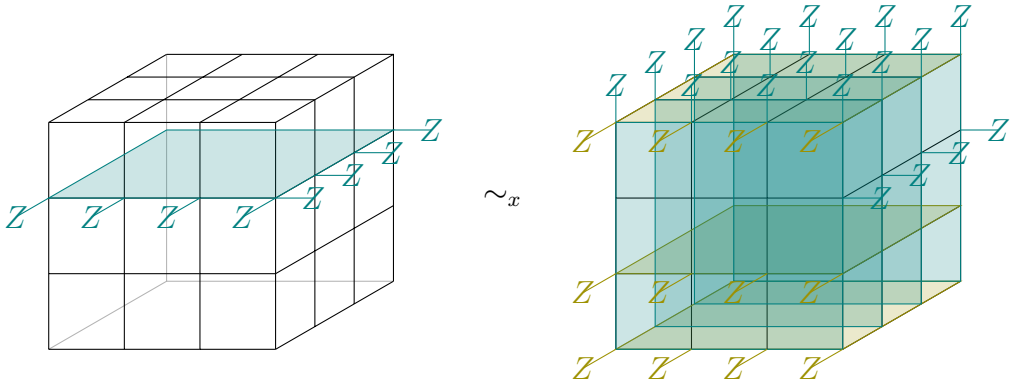


Figure 2.8: This illustration depicts two distinct symmetries in G_{444}. For clarity, the Z operators acting on opposite faces of the cube are omitted. The two symmetries belong to the same equivalence class under $\sim_x$, as their product corresponds to a group element generated by the L_{x_n} symmetries. Intuitively, the equivalence relation $\sim_x$ checks whether the Z operators acting on links along the x-direction coincide.

Definition 5. *Let L_i be as defined in Definition 3. Then the subgroup G_x is defined as*

$$G_x = \langle L_{x_1}, L_{x_2}, ..., L_{x_{N_x}} \rangle. \tag{2.73}$$

Its associated equivalence relation $\sim_x$ is

$$g \sim_x h \Leftrightarrow g^\dagger h \in G_x \Leftrightarrow \exists (a_1, a_2 ... a_{N_x}) \in \{0,1\}^{N_x} : gh = \prod_{i=1}^{N_x} L_{x_i}^{a_i}. \tag{2.74}$$

Since G is an abelian group, all subgroups of G are normal subgroups and thus is G_x. The cardinality of G_x is

$$|G_x| = 2^{N_x}. \tag{2.75}$$

The equivalence classes $[g]_x$ are sets with equal action on the free edges in the x-direction. This is illustrated in Fig. 2.8. Intuitively, two elements of G multiplying to an element of G_x must match the action of Zs in the x-direction. Consequently, their product only acts on links perpendicular to the x-direction.

Definition 6. *For G_x the quotient group G/G_x is defined as*

$$G/G_x = \Big\{ hG_x : h \in G \Big\}. \tag{2.76}$$

For finite groups, the cardinality of G/G_x is directly related to the order of G and G_x:

$$|G/G_x| = \frac{|G|}{|G_x|} = 2^{N_y+N_z-1} \tag{2.77}$$

Every equivalence class $[g] \in G/G_x$ corresponds one to one to a product of S operators, which will be shown in the following. Every distinct equivalence class $[g]_x$ can be represented by a $g \in G$, for which all a_i are chosen to be zero so that:

$$g = \prod_{i=1}^{N_y} L_{y_i}^{b_i} \prod_{i=1}^{N_z} L_{z_i}^{c_i}, \tag{2.78}$$

for some $b \in \{0,1\}^{N_y}$ and $c \in \{0,1\}^{N_z}$. Any other element $h \in [g]_x$ must have matching b and c, which follows from

$$gh = \prod_i L_{x_i}^{a_i}. \tag{2.79}$$

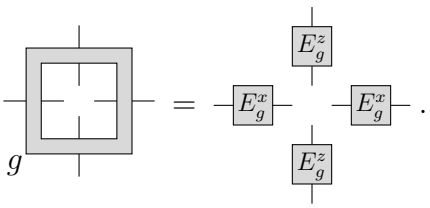
Consequently, their corresponding E^xs must match. Therefore, they are denoted with an additional index g, which indicates their corresponding $g \in G/G_x$:

$$h \sim_x g \implies E_g^x = E_h^x. \tag{2.80}$$

The opposite direction follows from the fact that E_g^x and E_h^x are completely specified by their corresponding b and c. Hence, there is a one to one correspondence between $[g]_x$ and E_g^x. The analysis above holds equivalently for the y and z-directions, for which the resulting operators acting on the edges in the y and z-directions are denoted by E_g^y and E_g^z, respectively. Then, the action of a symmetry on a patch of the tensor network can be decomposed into:

$$g = E_g^x \otimes E_g^x \otimes E_g^y \otimes E_g^y \otimes E_g^z \otimes E_g^z, \tag{2.81}$$

which is represented graphically as:



(2.82)

To improve readability, the edges in y direction have been omitted.
With all required descriptions, we are now in a position to describe why distinct ground states differ by a global non-localized operator. We refer to this fact as the Closure Property. For the statement and the proof, we again use a two-dimensional graphical notation.

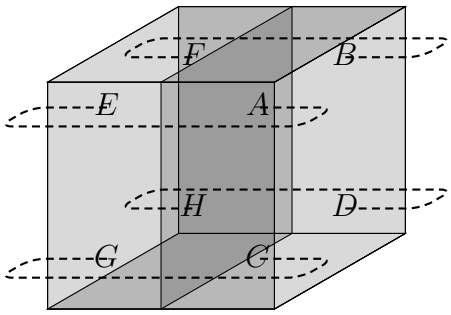
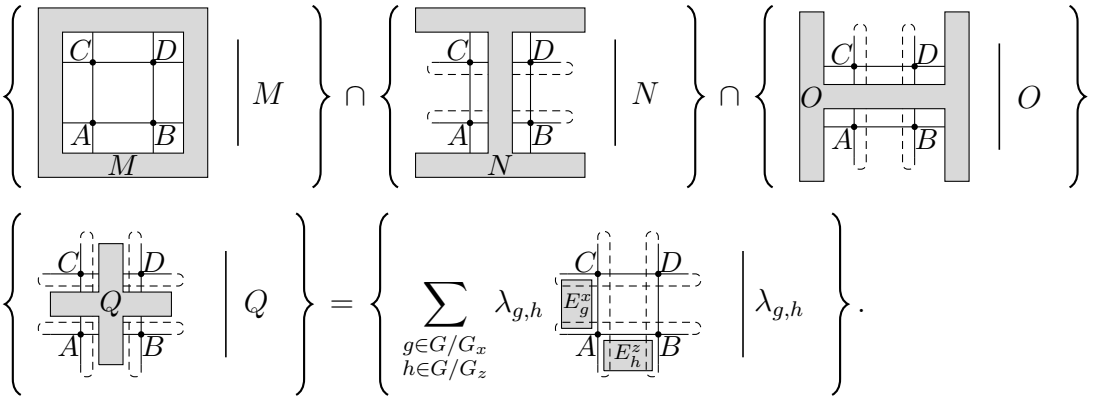


Figure 2.9: This graphic shows one three-dimensional term equivalent to the N term presented in the Closure Property. The connecting lines within the cube are not drawn for better readability. A, for example, is connected to B, C, and E with three of its links. The other three links are connected to the shaded region, representing the closure N.

However, in the figures, the physical indices will be implicitly denoted by dots on the tensors, not explicitly by drawing a physical edge. We do this, because a fourth edge on every tensor is necessary to represent the two-dimensional version. Three-dimensional versions of the figures are hard to draw and understand, so they will only be explained verbally.
It is useful to think of the images presented as projections of the three-dimensional version onto the x-z-plane, where the tensors have been blocked in the third dimension already.

Theorem 5 (Closure Property)**.** *Let A, B, C, and D be arbitrarily sized X-Cube tensor networks so that their dimensions fit along the contracted edges. Let M, N, O, and Q be arbitrary tensors with dimensions that match along the contracted edges. Let E_g^x and E_h^z be operators as in Eq.* (2.72)*. Then, the following property holds:*

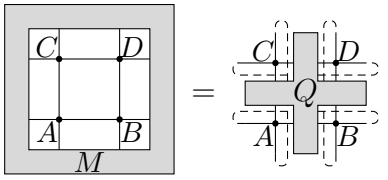


$$\left\{ \; \left| \; M \right. \right\} \cap \left\{ \; \left| \; N \right. \right\} \cap \left\{ \; \left| \; O \right. \right\} \left\{ \; \left| \; Q \right. \right\} = \left\{ \sum_{\substack{g\in G/G_x \\ h\in G/G_z}} \lambda_{g,h} \; \left| \; \lambda_{g,h} \right. \right\}. \tag{2.83}$$

Before proving this theorem, let us address the differences to the three-dimensional version. The biggest difference is that in the three-dimensional version there are eight terms

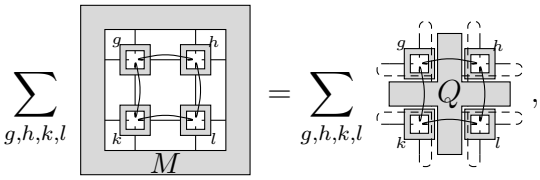
on the left side instead of four. Each term being a different configuration of the border. The three-dimensional term corresponding to the N term above is shown in Fig. 2.9. Another notable difference is that the term on the right has three operators E_g^x,E_h^z, and E_k^y acting in x,z and y-direction respectively. Despite these differences the two-dimensional proof does generalize in these aspects and therefore suffices.

Proof. Let us start by showing the inclusion of the left side. First, we choose an arbitrary element in the intersection of the M and the Q term:



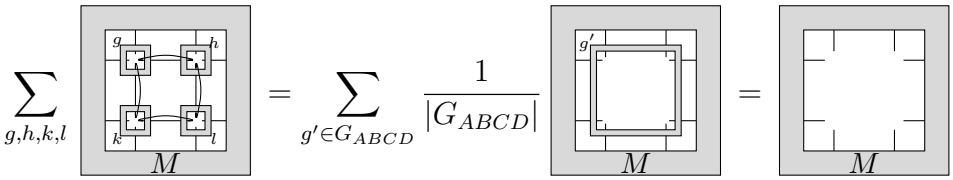
(2.84)

Next, we apply the pseudo-inverse of $ABCD$, so that



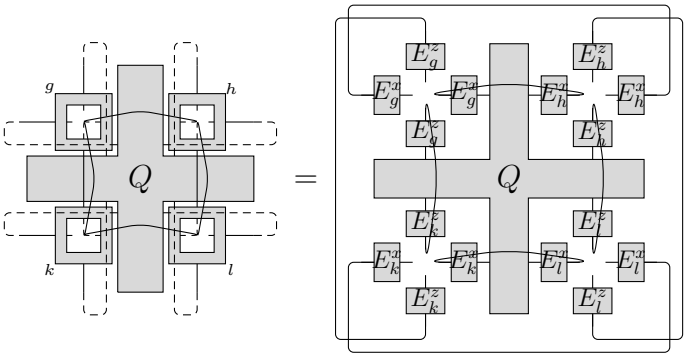
(2.85)

where g, h, k, l are the elements of the symmetry groups of A, B, C and D respectively. Next, the symmetries on the left hand side are merged:



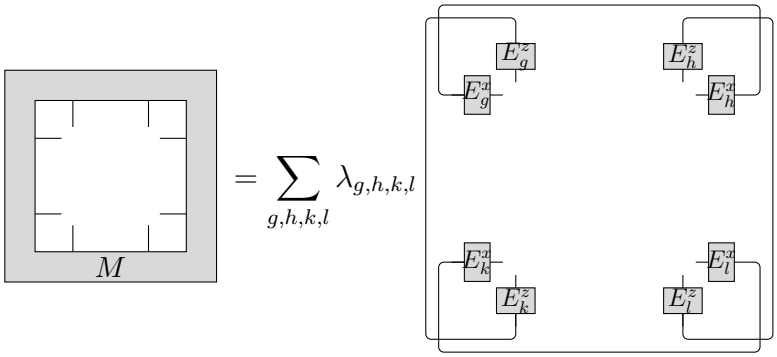
(2.86)

where the last equality follows from the symmetrization of M, as in the proof of Theorem 4. On the right hand side of Eq. (2.85), we use the decomposition of each element in the sum as in Eq. (2.82):



(2.87)

Note that all edges of Q are contracted. We denote this trace as $\lambda_{g,h,k,l}$, then recombine the left and the right side to:



$$= \sum_{g,h,k,l} \lambda_{g,h,k,l} \quad . \tag{2.88}$$

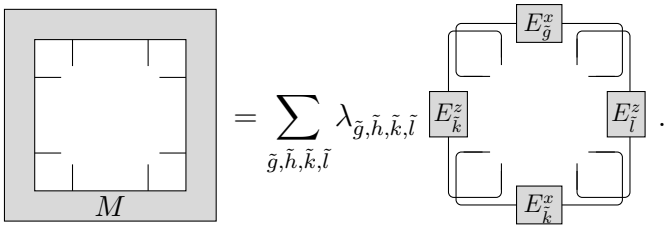
Although g and k are part of different groups, the quotient groups G_A/G_A^z and G_C/G_C^x are equal. Therefore, E_g^z and E_k^z can be multiplied and we denote their product by E_g^z, k:

$$E_{g,k}^z = E_g^z \cdot E_k^z. \tag{2.89}$$

Furthermore, since

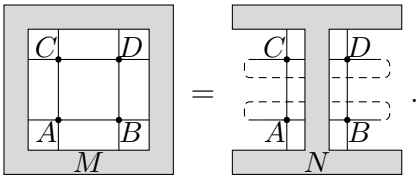
$$\sum_{\substack{g \in G_C \\ k \in G_A}} E_{g,h}^z = \sum_{\tilde{g} \in G_A/G_A^z} E_{\tilde{g}}^z, \tag{2.90}$$

we simplify M to:



$$= \sum_{\tilde{g},\tilde{h},\tilde{k},\tilde{l}} \lambda_{\tilde{g},\tilde{h},\tilde{k},\tilde{l}} \quad . \tag{2.91}$$

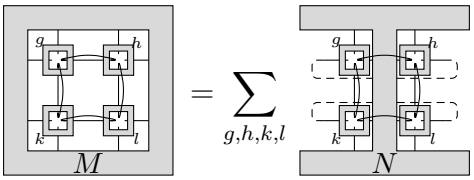
Next, we show that $E_{\tilde{g}}^x$ and $E_{\tilde{k}}^x$ are further constrained. For that, we start with an arbitrary element in the intersection of the M and the N term.



$$= \quad . \tag{2.92}$$

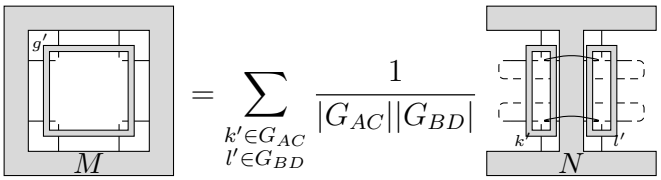
Then, we apply the inverse of A, B, C, D on both sides.

$$\sum_{g,h,k,l} \quad = \sum_{g,h,k,l} \quad . \tag{2.93}$$



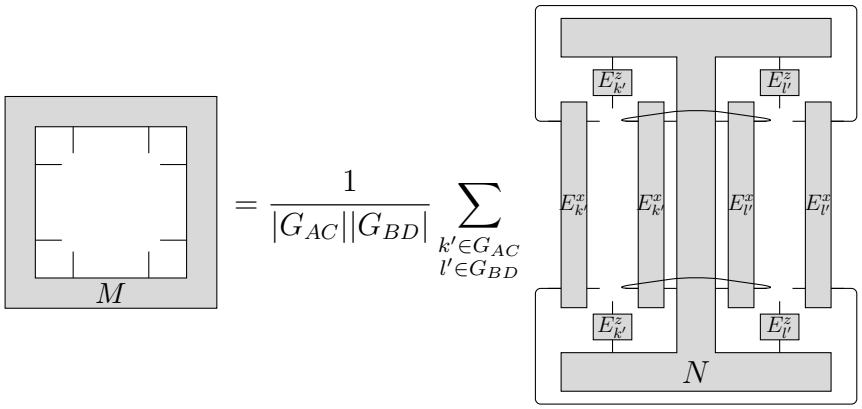
Here, g, h, k, l are elements of the respective symmetry groups of A,B,C and D as above. Next, we simplify both sides by merging symmetries.

$$\sum_{g' \in G_{ABCD}} \frac{1}{|G_{ABCD}|} \quad = \sum_{\substack{k' \in G_{AC} \\ l' \in G_{BD}}} \frac{1}{|G_{AC}||G_{BD}|} \quad \tag{2.94}$$



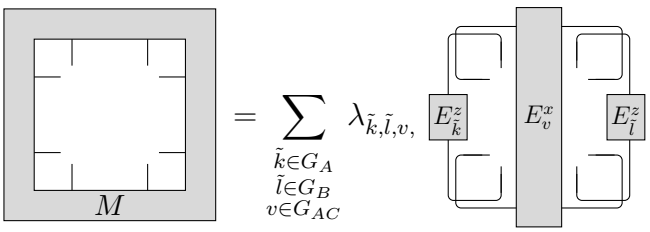
G_{ABCD}, G_{AC} and G_{BD} refer to the symmetry groups of the contracted blocks $ABCD, AC$ and BD respectively. On the left, we use the symmetrization of M again. On the right, we decompose the action of k' and l', as in Eq. (2.87).

$$\quad = \frac{1}{|G_{AC}||G_{BD}|} \sum_{\substack{k' \in G_{AC} \\ l' \in G_{BD}}} \quad . \tag{2.95}$$



As above, we simplify the E^x terms. Then we compare Eq. (2.91) to Eq. (2.95) and obtain:

$$\quad = \sum_{\substack{\tilde{k} \in G_A \\ \tilde{l} \in G_B \\ v \in G_{AC}}} \lambda_{\tilde{k},\tilde{l},v,} \quad . \tag{2.96}$$



This argument can be repeated for the z-direction using an element in the intersection of an M term and an O term in Theorem 5 and in the y-direction using the corresponding

terms in the third dimension. This concludes the proof for the inclusion of the left side. The inclusion of the right side is trivially fulfilled by the proper choice of M, N, O and Q, which proves Theorem 5. □

The two theorems proven in this section will be necessary to find the groundspace degeneracy of the parent Hamiltonian of the X-Cube tensor network. Theorem 4 will be used to show how the groundspace of the parent Hamiltonian corresponding to a region is related to smaller, intersecting subregions of the tensor network. Theorem 5 provides the tool necessary to close the boundary and quantify the related degrees of freedom.

3 Analysis of the X-Cube Tensor Network

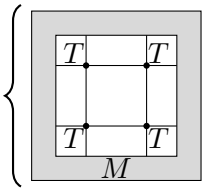
Within the three sections of this chapter, three main known results of the X-Cube Stabilizer model will be shown from the perspective of the tensor network formalism. The sub-extensive groundspace degeneracy of the model is derived in the following section. Next, excitations with restricted mobility known as fractons, planons and lineons will be discussed in Section 3.2. The layered structure of the X-Cube model in terms of coupled toric code layers will be the content of Section 3.3.

3.1 Groundspace Degeneracy of the Modified X-Cube Hamiltonian

In this section, the groundspace degeneracy of the parent Hamiltonian of a periodic X-Cube tensor network will be obtained purely from the tensor network description. Although for our X-Cube tensor network a Hamiltonian was the starting point, this approach showcases how the properties of the X-Cube model appear in its tensor network description. Similarly to the last section, two-dimensional plots will be used to illustrate the arguments. Three-dimensional plots are provided whenever necessary. This approach follows closely the strategy applied in [11].
As a first step, the $2 \times 2 \times 2$ subspace of an X-Cube tensor network is defined. A corresponding Hamiltonian is defined, the groundspace of which is the $2 \times 2 \times 2$ subspace defined before. We can then grow the $2 \times 2 \times 2$ block and show, using the Intersection Property, that the groundspace of the larger block is equal to the intersection of the groundspaces of all $2 \times 2 \times 2$ subspaces. Finally, the boundary is closed. The degrees of freedom attached to the choice of different closures correspond to degenerate ground states.
Let us start by defining the $2 \times 2 \times 2$ subspace of T with arbitrary closure. $S_{2\times2\times2}$ is

defined as

$$S_{2\times2\times2} = \left\{ \quad \middle| \; M \right\}. \tag{3.1}$$



It is the subspace spanned by eight X-Cube tensors (only four depicted in the two-dimensional plot), closed with an arbitrary tensor M with dimensions matching on contracted edges. Let $\Pi_{S_{2\times2\times2}}$ be the projector onto this space. Then, the above subspace is the groundspace of

$$H = \mathbb{1} - \Pi_{S_{2\times2\times2}}. \tag{3.2}$$

Similarly, let $S_{N_x\times N_y\times N_z}$ be the space spanned by the $N_x \times N_y \times N_z$ tensor network with arbitrary closure. For each n, k, l define the corresponding projector $\Pi_{S_{N_x\times N_y\times N_z}}$ on it. The first goal is to show that the groundspace of

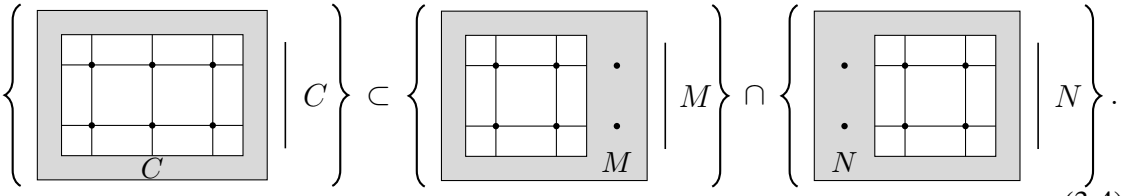
$$H_{N_x\times N_y\times N_z} = \sum_c (\mathbb{1} - \Pi_{S_{2\times2\times2}})_c \tag{3.3}$$

is $S_{N_x\times N_y\times N_z}$.

Theorem 6. *Let $H_{2\times2\times2}$ and $H_{N_x\times N_y\times N_z}$ be the Hamiltonians as defined above. Then, the groundspace of $H_{N_x\times N_y\times N_z}$ is $S_{N_x\times N_y\times N_z}$.*

Proof. Starting with the $2\times2\times2$ tensor network, layer after layer will be added as depicted in Fig. 3.1. After each layer Theorem 6 will be proven to hold using the Intersection Property.
After adding the first layer, the resulting tensor network lies in the intersection of subspaces spanned by the two $2 \times 2 \times 2$ blocks of Ts with arbitrary closure.

$$\left\{ \quad \middle| \; C \right\} \subset \left\{ \quad \middle| \; M \right\} \cap \left\{ \quad \middle| \; N \right\}. \tag{3.4}$$



Due to the Intersection Property, the inclusion of the left hand side also holds. Hence, Theorem 6 holds after this step. Theorem 6 can then be proven by induction. After every

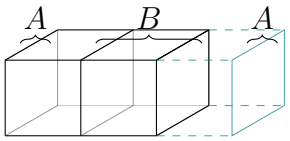
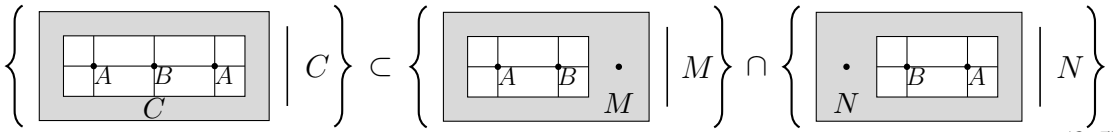


Figure 3.1: The illustration shows how the tensor network is expanded layer by layer. The new layer is marked in teal, and the connecting lines to the new layer are dashed. A and B indicate how the tensor network is divided into blocks to apply the Intersection Property, as presented in Theorem 4. The closure is not depicted.

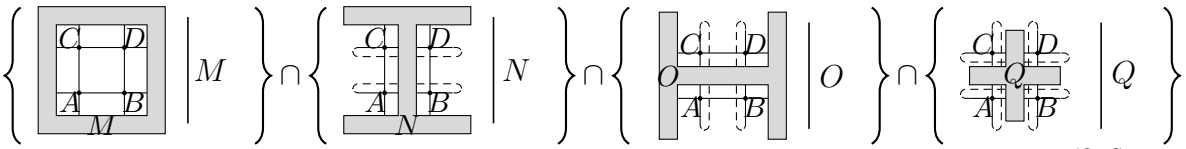
additional layer added to the system, the equation



(3.5)

holds, where A is the size of the added layer. The size of B is given implicitly as the remainder of the tensor network. Due to the Intersection Property, see Theorem 4, equality of Eq. (3.5) holds. The subspaces on the left can further be expressed as intersected $2 \times 2 \times 2$ subspaces, which proves Theorem 6 inductively for any system size. Fig. 3.1 presents exemplary how the division of the tensor network into A and B is done for the step from $3 \times 2 \times 2$ to $4 \times 2 \times 2$. □

Due to Theorem 6 any X-Cube tensor network of arbitrary size is $S_{N_x \times N_y \times N_z}$. The last step towards obtaining the groundspace degeneracy is closing the boundary. This Hamiltonian of the system with periodic boundary conditions is $H = H_1 + H_2 + ... + H_8$, whose individual terms are Hamiltonians with different configurations of boundaries. The groundspace of each of these Hamiltonians is considered below in Eq. (3.6). Note, that only four configurations are depicted in two dimensions. In the three-dimensional picture, eight such terms appear.



(3.6)

Theorem 5, specifies that any state in this intersection is closed by membranes of X operators. The number of different membranes are counted by G/G_x, G/G_y and G/G_z. Adding their orders, the resulting degeneracy of the Hamiltonian H obtained this way is

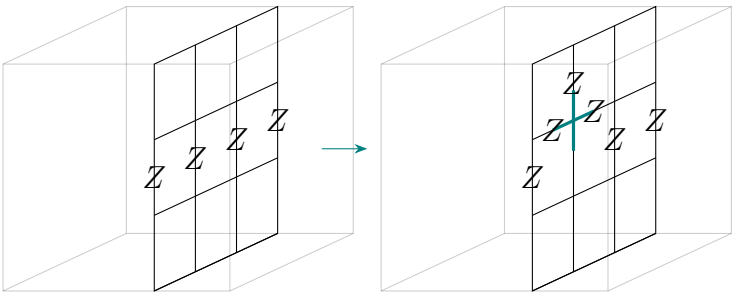


Figure 3.2: On the left, a ground state of the tensor network different from the "empty" network with no operators on the virtual edges is presented. The right depiction shows the tensor network after applying the symmetry marked in teal. Note that the symmetry does not change the parity of Zs around any cube.

given by:

$$\mathrm{GSD}_{XC} = |G/G_x| + |G/G_y| + |G/G_z| = 2^{2(N_x+N_y+N_z)-3} \tag{3.7}$$

This matches the known degeneracy of the X-Cube model, obtained in section Section 1.4. Naturally, the question arises how these different ground states are related. Since they appear locally indistinguishable, they are necessarily related by global operators. The degeneracy arises from the possibility of closing the boundary with different membranes. These membranes E^i themselves are generated by loop symmetries.
After the boundary is closed, the remainders of these loops are loops around one of the three holes in the three-torus $\mathbb{T}^3$. In each plane there are two non-equivalent loops. Fig. 3.2 shows one of these loops. It is mobile in the sense that it can be freely deformed within its plane using symmetries of the tensors within that plane. However, it is not possible to generate or remove the loop entirely by applying symmetries. This must be true, since all symmetries leave the parity of Z operators along all grid lines of the network invariant. Adding a string of Zs as depicted in Fig. 3.2 changes the parity along all grid lines perpendicular to it within its plane.
In total, there are $2(N_x + N_y + N_z)$ different loops of that form, two for every plane. There are three global constraints on them, and that is where the term -3 in Eq. (3.7) comes from. One such constraint is presented in Fig. 3.3.

3.2 Excitations of the X-Cube Tensor Network

This section discusses virtual excitations in the X-Cube tensor network. Virtual excitations in tensor networks are localized operators acting on virtual degrees of freedom. The localization is their distinguishing feature, separating them from logical operators as used above.

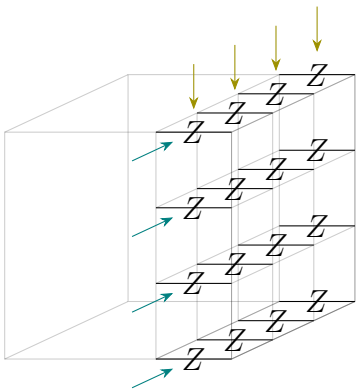


Figure 3.3: Applying a loop in the y-direction in all X–Y planes (indicated by the teal arrows) is equivalent to applying a loop in the z-direction in all X–Z planes (indicated by the olive arrows).

The discussion in Section 2.3 showed that the physical space is isomorphic to the virtual one. Hence, the excitations should match the ones known for the X-Cube stabilizer model closely.

3.2.1 Z Excitations - Fractons

When discussing virtual excitations, it can be useful to consider how these virtual operators are obtained from physical operations. For that, Table 2.1 is referenced. Symmetries that act on both the physical and virtual levels contain the information that a virtual Z can be generated by applying a physical Z_p operator on one of the physical qubits of an edge. The index p denotes the action on the physical level.
Virtual Z operators in the tensor network are mobile in the sense that they can be moved from their link via the symmetries of the single tensors. Despite that, a single Z is localized in the sense that it cannot be moved away from the four surrounding cubes of the lattice. Any cube in the ground state has an even number of Zs on its edges. Applying any symmetry of the tensor network will not change the parity of Zs around any cube. However, adding a single Z changes the parity of Zs on the four cubes adjacent to the edge. Hence, Z is localized in these four cubes, which are therefore called excited cubes. This is illustrated by the four cubes marked in teal below.

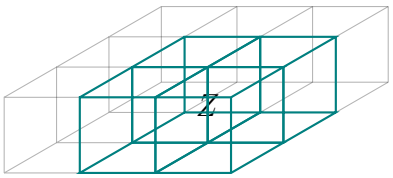


Figure 3.4: A single Z operator on an edge is said to excite the four adjacent cubes, each of which then has an odd number of Z operators on its edges.

Any Z added to the tensor network necessarily changes the parity of four cubes. Hence, it is never possible to move a single cube as that would require erasing one cube excitation and simultaneously creating one new excitation. In contrast, pairs of neighboring excited cubes are mobile within a plane. They can be moved by adding further Zs to the tensor network. The following figures show that two adjacent cubes can move within a plane.

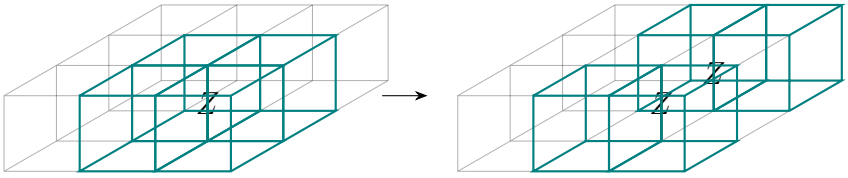


Figure 3.5: Two cube excitations, adjacent in the x-direction can move in y-direction by inserting an Z as depicted here.

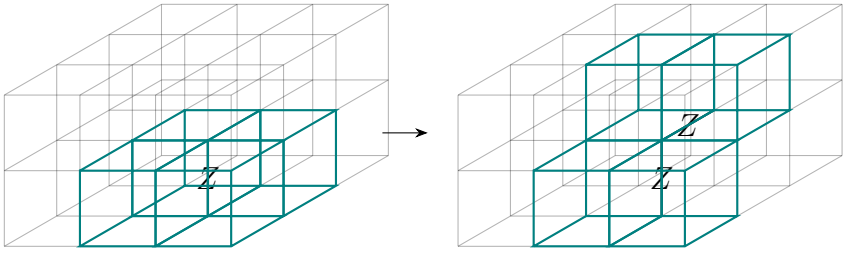


Figure 3.6: Two cubes excitations, adjacent in the x-direction can move in the z-direction by inserting an Z as depicted here.

Two adjacent cube excitations are mobile in the plane perpendicular to their common face. If the two cubes are separated as depicted below, they maintain this mobility. This follows trivially when viewing the two separated excited cubes as the product of a string of adjacent cubes that cancel everywhere but at the endpoints, as depicted below.

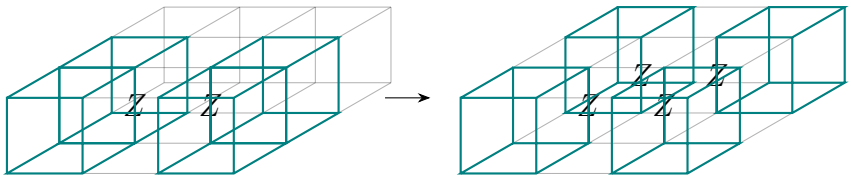
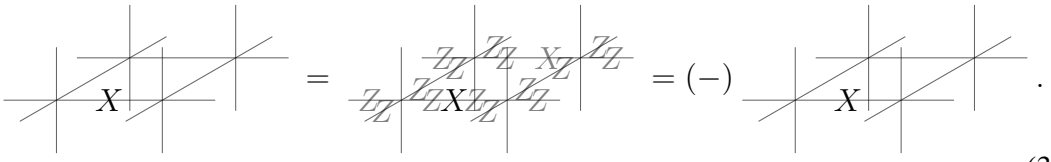


Figure 3.7: Separated cubes maintain their movability as depicted here.

As discussed in the previous section, logical operators between the different ground states are loops of Z operators. These correspond to taking a pair of cube excitations, moving them around the torus, and recombining them with the other two excitations. These observations match the known features of fractal excitations in the stabilizer X-Cube model, as described in [15]. Since these virtual Z operators are equivalent to physical Z_p operators, matches our expectation.

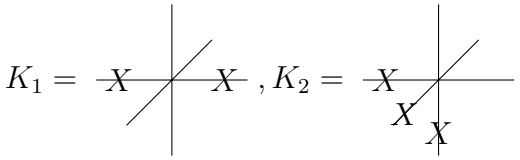
3.2.2 X Excitations - Lineons

Similarly to the previous study of virtual Z operators, virtual X operators can be analyzed with respect to their mobility in the network.
Virtual X operators cannot be generated individually by a physical operation. It turns out that due to the symmetries of the tensor network, virtual X operators must come in even numbers in every plane of the system, since



(3.8)

The global minus sign originates from the commutation of $ZXZ = -X$.
Instead, multiple virtual Xs can be obtained by applying X_p on the matching physical degrees of freedom. Fig. 3.8 illustrates this. Note that the exclusion argument from Eq. (3.8) does not apply to Xs generated that way, since only even numbers of Xs appear in every plane. Consequently, virtual X operators only appear in even numbers in all lattice planes of the tensor network. The only two local operations applied to a tensor that follow this constraint are presented in Eq. (3.9). They correspond to the virtual part of the symmetry configurations presented in Fig. 3.8.



(3.9)

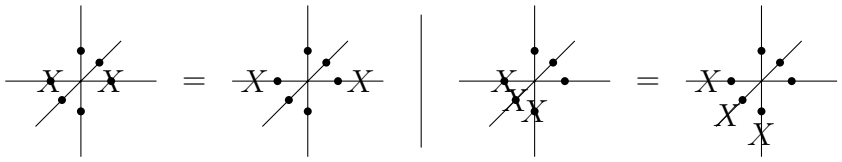


Figure 3.8: The dots represent the physical degrees of freedom which have mostly been omitted in the previous sections. The tensor T has symmetries under the action of both the physical and the virtual degrees of freedom. Two representative ones for physical Xs and virtual Zs are shown here.

Since virtual X operators cannot be moved by symmetries of the adjacent tensors, they are localized at the edges they are acting on. They can be moved in a straight line by applying K_1 on a neighboring tensor. This will erase the old X as it squares to the identity and create a new X one edge further. This works independently for all three directions and is illustrated for the y-direction below.

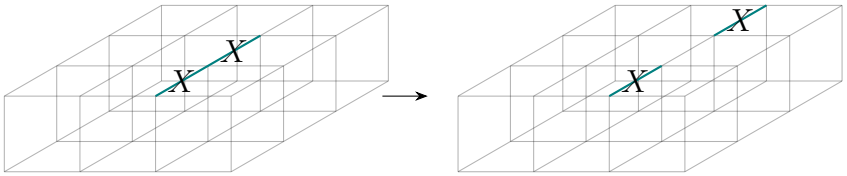


Figure 3.9: A pair of X excitations can move within their line by applying K_1 repeatedly.

The K_2 operation does not enable a single X to move. Instead, this operation allows three X operators in different directions to merge at a tensor. Tracing the virtual X operators involved back to their physical operations explicitly shows that no stabilizers are violated on the merging tensor. Therefore, it makes sense to attribute the X operators not only to their respective links, but also the neighboring tensors. This is related to the doubling procedure, which introduced a ZZ stabilizer on each edge. We therefore call an edge excited when an X acts on it. A vertex is considered to be excited when it has an odd number of X operators in any plane around it.

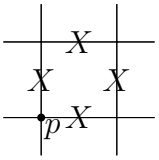
3.3 Layer Structure of the X-Cube Tensor Network

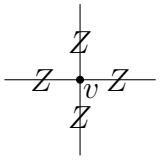
It has been discovered that the X-Cube model can be understood as a layering of coupled toric codes [9]. This structure can also be seen in the tensor network, as will be described in the following section.
In order to describe the layering of toric codes a short introduction to the tensor network representation of the toric code groundspace is required. The toric code, originally intro-

duced by Kitaev [8], is a stabilizer code on a square lattice with periodic boundary conditions. One physical $\mathbb{C}^2$ degree of freedom is located at every edge of a two-dimensional rectangular lattice. The Hamiltonian of the toric code model consists of stabilizer terms on plaquettes and vertices:

$$H = -\sum_{p} A_p - \sum_{v} B_v, \tag{3.10}$$

where

$A_p =$ 

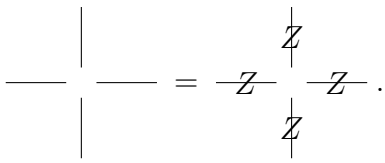
$, B_v =$ 

. (3.11)

All Hamiltonian terms are local and commute with each other. A tensor network description of the toric code groundspace can be acquired via a similar procedure used for the X-Cube model. The degrees of freedom are doubled, edge stabilizers are introduced, leading to a modified Hamiltonian. Then the projection into the ground state leads to the toric code tensor T^{TC}:

$$T^{TC} = \sum_{i_1,i_2,i_3,i_4} P^{TC} \left|i_1, i_2, i_3, i_4\right\rangle \otimes \left|i_1, i_2, i_3, i_4\right\rangle, \quad P^{TC} = \frac{1}{4}\sum_{k=0}^{1} (Z^k)^{\otimes 4}. \tag{3.12}$$

T^{TC} can then be analyzed with respect to its eight symmetries, see Table 3.1, and pseudo-invertibility. The virtual symmetry group G^{TC} of the toric code tensor is isomorphic to Z_2. The two symmetries of it are the following:



. (3.13)

The pseudo-inverse of T^{TC} is $(T^{TC})^\dagger$:

$$T^{TC}(T^{TC})^\dagger = P^{TC}. \tag{3.14}$$

For any system size, the symmetry group remains isomorphic to Z_2. Analogously to the X-Cube case, any added tensor brings along one loop symmetry, which is constrained to match the existing loop symmetry. Hence, the system size does not need to be specified for G^{TC}. The two elements of G^{TC} for an arbitrary system size are the identity acting on all free edges and virtual Z operators acting on all free edges. Repeating the analysis of the Intersection Property and the Closure Property for the toric code yields the result that the groundspace degeneracy of the toric code with periodic boundary conditions is

phyiscal				*virtual*			
Z	Z	Z	Z	I	I	I	I
X	X	I	I	X	X	I	I
X	I	X	I	X	I	X	I
X	I	I	X	X	I	I	X
Z	I	I	I	Z	I	I	I
I	Z	I	I	I	Z	I	I
I	I	Z	I	I	I	Z	I
I	I	I	Z	I	I	I	Z

Table 3.1: Symmetries of the toric code tensor. Each row is a symmetry consisting of the four left operators acting on the physical degrees of freedom and the four right operators on the corresponding virtual edges.

constant:

$$\text{GSD}_{TC} = 2^2. \tag{3.15}$$

Intuitively, the layering of toric codes in the X-Cube model can be anticipated in the groundspace degeneracy scaling. Every layer introduces a factor of 2^2, which matches the groundspace degeneracy of a single toric code layer. The procedure of adding a toric code layer [12] can be slightly modified to work for the doubled X-Cube and the doubled toric code. However, in the tensor network description, virtual degrees of freedom also have to be considered. The procedure for the addition of a single tensor is described below. In order to add a layer of toric code to a larger X-Cube tensor network, the same steps have to be done translationally invariant for the entire layer. First, one edge of the X-Cube tensor is elongated. Then a toric code tensor is placed on that stretched edge, as depicted on the left of Fig. 3.10. Two more physical degrees of freedom are added in the $|0\rangle$ state, also illustrated on the left of Fig. 3.10. Then, two sets of CX gates are applied. The gates are shown in the middle and the right side below.

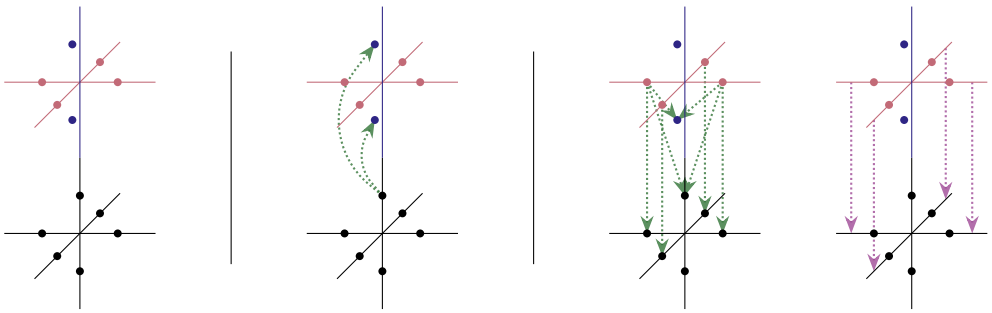


Figure 3.10: On the left, the setup before the application of CX gates is shown. The bottom depicts the X-Cube tensor, while the added toric code tensor is shown in red. Two $|0\rangle$ states are highlighted in blue, placed adjacent to the elongated line to indicate their independence prior to coupling. In the center, the first two CX gates are shown; their color indicates that they act on physical degrees of freedom. The right side presents the second set of CX gates, divided into two figures for the physical (green) and virtual (violet) gates.

The following computation proves that the upper part consisting of added $|0\rangle$ states and a toric code is equivalent to an X-Cube tensor after applying the CX gates. The symmetries of the tensor network after the application are compared to the symmetries of the $1 \times 1 \times 2$ X-Cube tensor network. If they match, both tensor networks represent the same state. The single X-Cube tensor carries twelve degrees of freedom, six physical and six virtual ones. Due to its twelve independent symmetries, it uniquely specifies one state. Blocking two X-Cube tensors changes these numbers to twenty-two degrees of freedom and twenty-two independent symmetries. Two less than twice as much due to the contraction of two virtual degrees of freedom. The toric code layer with doubled degrees of freedom has a total of eight symmetries, presented in table Table 3.1. The $|0\rangle$ states are only symmetric under a physical Z operation. Hence, a total of ten symmetry generators are supported on the added degrees of freedom. The conjugate action of the gates CX transforms these ten new symmetries, as well as the twelve symmetries of the X-Cube tensor. Two examples are shown in Figs. 3.11 and 3.12, the first of which additionally highlights that the order in which the physical CX are applied matters.

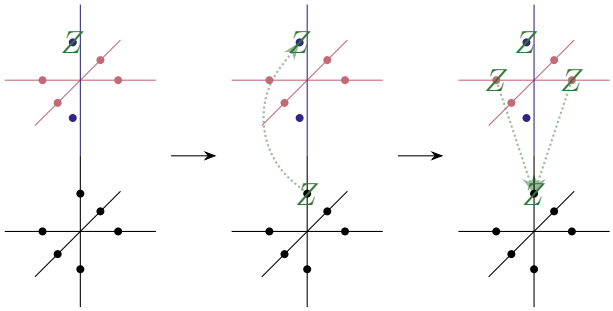


Figure 3.11: This figure illustrates the transformation of the symmetry of the added $|0\rangle$ state (left) following the application of the first set of CX gates (center) and the second series of CX gates (right), alongside the relevant gates.

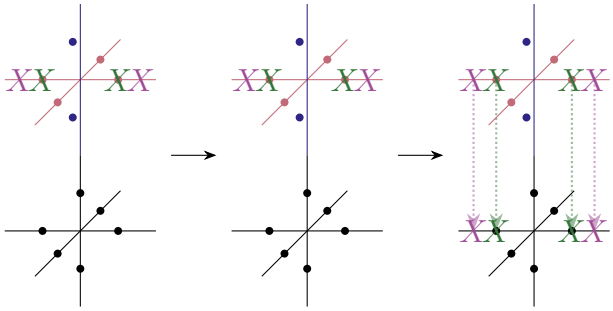
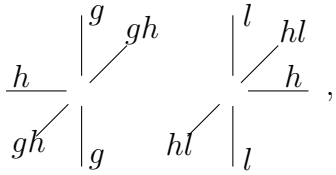


Figure 3.12: This figure illustrates the transformation of the symmetry of the toric code (left) following the application of the first set of CX gates (center) and the second series of CX gates (right), alongside the relevant gates.

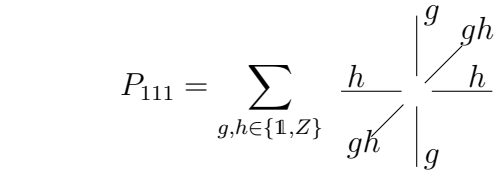
This computation is done for all 22 symmetry generators. By choosing specific products of the resulting symmetries, all generators of the $G_{2\times1\times1}$ are obtained. Hence, the tensor network after the application of the CX gates represents the same state as the $2 \times 1 \times 1$ X-Cube tensor network.

This can also be seen on the level of the purely virtual symmetries of the X-Cube tensor. Applying only the virtual CX gates presented above gives rise to an isomorphism between P_{211} and $P_{111} \otimes P_{011}^{TC}$, where

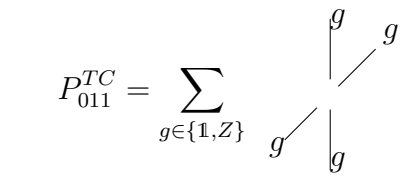
$$P_{211} = \sum_{g,h,l\in\{\mathbb{1},Z\}}$$



, (3.16)

$$P_{111} = \sum_{g,h\in\{\mathbb{1},Z\}}$$  , (3.17)

and

$$P_{011}^{TC} = \sum_{g\in\{\mathbb{1},Z\}}$$  . (3.18)

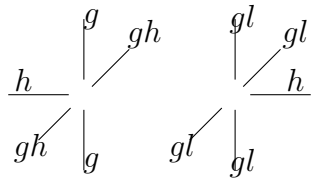
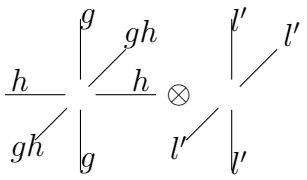
Now, we apply the virtual CX gates from the left to right side of P_{211}. The conjugate action on the symmetry operators will effectively act as a multiplication to the right, since

$$CX(g \otimes l)CX^{\dagger} = (g \otimes gl), g, l \in \{\mathbb{1}, Z\}. \tag{3.19}$$

The sum over g on the right side can then be absorbed into the sum over l:

$$l' = gl. \tag{3.20}$$

By reordering the indices, the free edge on the right side can be attributed to the left part forming an X-Cube tensor symmetry:

$$\sum_{g,h,l\in\{\mathbb{1},Z\}}$$  $$\cong \sum_{g,h,l'\in\{\mathbb{1},Z\}}$$  . (3.21)

In this last expression, the layer of toric code in the symmetry group can be seen explicitly. Under conjugate action of certain CX gates, the projector P_{211} of the X-Cube model is isomorphic to the tensor product of P_{111} and P^{TC}.

$$P_{211} \cong P_{111} \otimes P_{011}^{TC} \tag{3.22}$$

4 Generalization to Z_n Symmetries

The X-Cube model can be generalized to symmetries that are representations of groups different from $Z_2 = \{e, g\}$. One such generalization is to $G = Z_n = \{e, g, g^2 ... g^{n-1}\}$, which will be considered in this chapter.

4.1 Generalizing the X-Cube Stabilizers

The generalization can be made in various ways. The ones presented here will start on the level of the stabilizer code. Even then, no clear unique generalization exists. The first step is to define the local Hilbert space to be $\mathbb{C}^n \cong \text{span}\{|g\rangle, g \in Z_n\}$. Next, the operators X and Z are defined as representations of Z_n. X is given as the generalized shift operator and Z as the generalized clock operator:

$$X_g = \sum_{k=0}^{n-1} |k \oplus 1\rangle\langle k|, \quad Z_\chi = \sum_{k=0}^{n-1} \omega^k |k\rangle\langle k|. \tag{4.1}$$

For $n = 3$ their matrix representation in the Z basis is

$$X_g = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad Z_\chi = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \omega & 0 \\ 0 & 0 & \omega^2 \end{pmatrix}. \tag{4.2}$$

By straightforward computation,

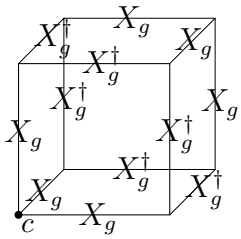
$$Z_\chi X_g = \chi(g) X_g Z_\chi = \omega X_g Z_\chi. \tag{4.3}$$

With these operators, we can define a generalized Z_n X-Cube Hamiltonian:

$$H^n = -\frac{1}{n} \sum_c \sum_{k=0}^{n-1} A_c^k - \frac{1}{n} \sum_{v,i} \sum_{k=0}^{n-1} B_{v,i}^k, \tag{4.4}$$

where

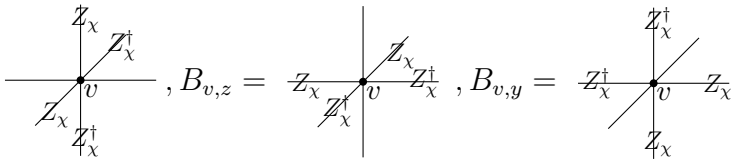
$$A_c = \quad ,$$



(4.5)

and

$$B_{v,x} = \quad , B_{v,z} = \quad , B_{v,y} = \quad ,$$



(4.6)

The Stabilizers are expressed without additional n index to maintain clarity. Within this chapter A and B terms as well as Hamiltonians always refer to Z_n operators. Note that the vertex stabilizers were chosen such that

$$\forall v : \prod_i B_{v,i} = \mathbb{1}. \tag{4.7}$$

All Hamiltonian terms commute, which can be verified using Eq. (4.3). Although this model is not a stabilizer code as described in Section 1.3 since it is not defined on qubits, the analysis regarding the ground state degeneracy can be considered similarly. The ground state of this system is the $+1$ eigenstate of all stabilizers, where

$$S = \langle A_c, B_{v,i} \rangle. \tag{4.8}$$

This is equivalent to the Z_2 case, where the generalized stabilizers of Eqs. (4.5) and (4.6) are used. The stabilizers in this caseb are no longer dividing the Hilbert space into two $\mathbb{C}$ eigenspaces but instead into n $\mathbb{C}$ eigenspaces. Hence, Theorem 1 generalizes to the Z_n case and

$$\log_n \text{GSD} = \tilde{n} - k. \tag{4.9}$$

Counting the number of independent generators of S works equivalently to the Z_2 case, see Eqs. (1.37) to (1.40). This results in the same number of independent generators of S. Therefore, the groundspace degeneracy scales similarly to the Z_2 case as:

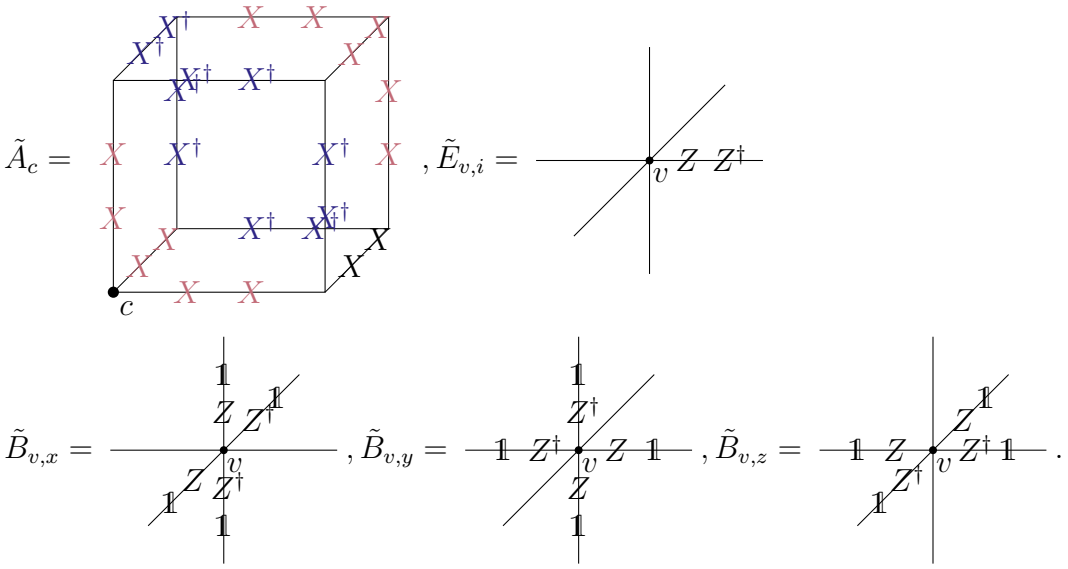
$$\log_n \text{GSD} = 2(N_x + N_y + N_z) - 3. \tag{4.10}$$

4.2 The Z_n X-Cube Tensor.

In the following section, a Z_n X-Cube tensor is derived from the generalized Z_n Hamiltonian, see Eq. (4.4). Then, the discussions regarding the symmetries and properties of the tensor in the $n = 2$ case are shown to also hold for the $n > 2$ case. The computations from Chapter 2 hold in generalized forms. The core differences are noted in the text of this section.
In order to obtain a Z_n X-Cube tensor, the procedure of doubling degrees of freedom is applied to the Z_n model, where the generalized CX gates are defined as:

$$CX = \sum_k |k \oplus 1\rangle\langle k| \,, \tag{4.11}$$

where $\oplus$ denotes addition mod n. The qubits are added in the same pattern as in the Z_2 case, see Fig. 2.1. After adding degrees of freedom, applying CX gates as in Fig. 3.10 and changing generators, the stabilizers are:



(4.12)

Here and from now on, the indices g and χ are omitted for simplicity. For $\tilde{A}_c$ the X and $X^\dagger$ operators are additionally color coded for clarity. The isometry applied by this construction is the following:

$$V_n = \sum_i |i, i\rangle\langle i| \,. \tag{4.13}$$

To obtain the Z_n X-Cube tensor, we project onto the $+1$ eigenspace of the modified Z_n

Hamiltonian:

$$\tilde{H}^n = -\frac{1}{n}\sum_c \sum_{k=0}^{n-1} \tilde{A}_c^k - \frac{1}{n}\sum_{v,i} \sum_{k=0}^{n-1} \tilde{B}_{v,i}^k - \frac{1}{n}\sum_{v,i} \sum_{k=0}^{n-1} \tilde{E}_{v,i}^k. \tag{4.14}$$

The projector on the $+1$ eigenspace of X is

$$P_X = \frac{1}{n}\sum_{k=0}^{n-1} X^k. \tag{4.15}$$

Similarly,

$$P_Z = \frac{1}{n}\sum_{k=0}^{n-1} Z^k \tag{4.16}$$

is the projector in the $+1$ eigenspace of Z. Since all stabilizers are tensor products of powers of X and Z, the projectors on the $+1$ eigenspaces of the stabilizers are

$$P_{s_i} = \frac{1}{n}\sum_{k=0}^{n-1} s_i^k. \tag{4.17}$$

For the projection onto a groundstate of $\tilde{H}_n$, we choose

$$|+...+\rangle = H^{\otimes 6N_xN_yN_z} |0...0\rangle \tag{4.18}$$

as our starting state, where

$$H = \frac{1}{\sqrt{n}}\sum_{jk=0}^{n-1} \omega^{jk} |j\rangle\langle k|, H|0\rangle = |+\rangle \tag{4.19}$$

is the generalized Hadamard gate. Next, we apply the projections into the $+1$ eigenspaces of all generators of S_n:

$$|\phi_{GS}\rangle_n = \prod_{s_i} P_{s_i} |+...+\rangle. \tag{4.20}$$

The projectors onto the $+1$ eigenstates of the $\tilde{A}_c$ terms act trivially, hence

$$|\phi_{GS}\rangle_n = \prod_{\tilde{B}_{v,i}} P_{\tilde{B}_{v,i}} \prod_{\tilde{E}_{v,i}} P_{\tilde{E}_{v,i}} |+...+\rangle. \tag{4.21}$$

Similarly to the Z_2 case, we now consider the action of $P_{\tilde{E}_{v,i}}$ on every edge:

$$\begin{aligned} P_{\tilde{E}_{v,i}} |++\rangle &= \frac{1}{n}\sum_{k=0}^{n-1}(Z^k \otimes (Z^\dagger)^k) |++\rangle \\ &= \frac{1}{n}\sum_{k=0}^{n-1}(Z^k \otimes (Z^\dagger)^k)(H \otimes H) |00\rangle \\ &= \frac{1}{n}\sum_{k=0}^{n-1}(H \otimes H)(X^k \otimes (X^\dagger)^k) |00\rangle \\ &= \frac{1}{n}\sum_{k=0}^{n-1}(H \otimes H) |k, n \ominus k\rangle \\ &= \frac{1}{n}\sum_{k=0}^{n-1}(\mathbb{1} \otimes HH^T) |k, n \ominus k\rangle \\ &= \frac{1}{n}\sum_{k=0}^{n-1} |k, k\rangle \,. \end{aligned} \tag{4.22}$$

Here, we used

$$ZH = HX \tag{4.23}$$

and

$$HH^T = \sum_{k=0}^{n-1} |n-k\rangle\langle k| \,. \tag{4.24}$$

On every vertex, we place a tensor T^n:

$$T^n = \sum_{i_1,i_2,i_{3,4},i_5,i_6=0}^{n-1} P^n \, |i_1, i_2, i_3, i_4, i_5, i_6\rangle\langle i_1, i_2, i_3, i_4, i_5, i_6| \,, \tag{4.25}$$

where

$$P^n = \frac{1}{n^2}\sum_{k,j=0}^{n-1} (\mathbb{1}^{\otimes 2} \otimes (Z^\dagger)^k \otimes Z^k \otimes Z^k \otimes (Z^\dagger)^k) \times ((Z^\dagger)^j \otimes Z^j \otimes Z^j \otimes (Z^\dagger)^j \otimes \mathbb{1}^{\otimes 2}). \tag{4.26}$$

The contraction of a tensor network of T^n tensors results in the desired groundstate of the modified Z_n Hamiltonian.

Before discussing properties of T^n let us clarify a subtlety of the Z_n tensor network: contracted edges represent a trace over $\mathbb{C}^n$ and its dual $(\mathbb{C}^n)^*$. Therefore, moving an

Table 4.1: Symmetry generators of the Z_n X-Cube tensor T^n.

physical						*virtual*					
$Z^\dagger$	Z	Z	$Z^\dagger$	I	I	I	I	I	I	I	I
I	I	$Z^\dagger$	Z	Z	$Z^\dagger$	I	I	I	I	I	I
X	X	I	I	I	I	$X^\dagger$	$X^\dagger$	I	I	I	I
I	I	X	X	I	I	I	I	$X^\dagger$	$X^\dagger$	I	I
I	I	I	I	X	X	I	I	I	I	$X^\dagger$	$X^\dagger$
X	I	X	I	X	I	$X^\dagger$	I	$X^\dagger$	I	$X^\dagger$	I
Z	I	I	I	I	I	$Z^\dagger$	I	I	I	I	I
I	Z	I	I	I	I	I	$Z^\dagger$	I	I	I	I
I	I	Z	I	I	I	I	I	$Z^\dagger$	I	I	I
I	I	I	Z	I	I	I	I	I	$Z^\dagger$	I	I
I	I	I	I	Z	I	I	I	I	I	$Z^\dagger$	I
I	I	I	I	I	Z	I	I	I	I	I	$Z^\dagger$

operator O from $\mathbb{C}^n$ to $(\mathbb{C}^n)^*$ results in their transposed action O^T. For Z:

$$Z^T = Z, \tag{4.27}$$

whereas for X:

$$X^T = X^\dagger. \tag{4.28}$$

Hence, in depictions of the tensor network:

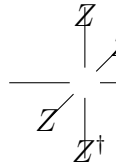
(4.29)

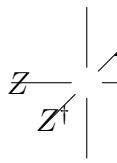
and

. (4.30)

In depictions with virtual X operators, we will specify which half of the edge they are acting on to avoid ambiguities. We now proceed to the discussion of symmetries of T^n.
The symmetry group of T^n is generated by operators, which are very similar to the ones in the Z_2 case. These operators are presented in Table 4.1.
The calculations for the symmetries proceed analogously as in the $n = 2$ case, see Eqs. (2.22), (2.24) and (2.26).
This allows us to define the symmetry group G^n_{111}, the group of operators acting on the six virtual degrees of freedom, which leave T^n invariant under their action. We do so by

defining the generators of G_{111}, L_x^n and L_z^n:

$L_x^n =$ 

, $L_z^n =$ 

. (4.31)

Then,

$$G_{111}^n = \langle L_x^n, L_z^n \rangle. \tag{4.32}$$

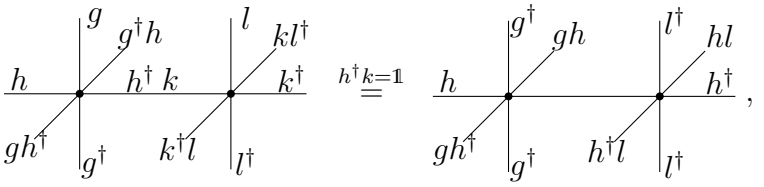
Since L_x^n and L_z^n are faithful representations of Z_n,

$$G_{111}^n \cong Z_n \times Z_n. \tag{4.33}$$

The discussion on the pseudo-inverse of T^n is precisely the same as for the $n = 2$ case. The pseudo-inverse of T^n is $(T^n)^\dagger$, so that

$$T^n (T^n)^\dagger = P^n. \tag{4.34}$$

The analysis of larger patches of T^n tensors leads to generalized versions of Theorems 2 and 3, as outlined in the following. Adding tensors to the network forces the generating loop symmetries to merge with existing loops of the tensor network. However, one has to be careful with how the merging works. The contracted edge represents a trace over $\mathbb{C}^n$ and $(\mathbb{C}^n)^*$. As discussed earlier, this means that moving a virtual operator from one side of the edge to the other corresponds to taking the transpose. Hence, the merging constraint becomes:



, (4.35)

where $g, h, k, l \in \{\mathbb{1}, Z, Z^2, ..., Z^{n-1}\}$. The constraint $h^\dagger k = \mathbb{1}$ enforces that the operator on the contracted edge is of the form $(Z^\dagger \otimes Z)^k, k \in \{0, 1, ..., n-1\}$, which is the identity according to Eq. (4.29). The generalization of Definition 3 are $L_{x_i}^n$ operators:

Definition 7. *Let x_k denote the plane perpendicular to the x-direction with x-coordinate k. Divide the non-contracted edges in x_k into two sets where those pointing to the back or to the bottom are labeled by a and those pointing to the front or the top are labeled by*

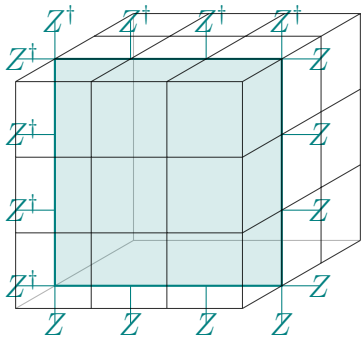


Figure 4.1: For a chosen plane y_2, the operator $L^n_{y_2}$ acts on all edges within that plane. It acts with an $Z^\dagger$ on edges pointing to the top or to the left and with an Z on edges pointing to the right or downwards.

b. Then, define the loop operator associated with x_k:

$$L^n_{x_k} = \bigotimes_{a \in x_k} Z_a \bigotimes_{b \in x_k} Z^\dagger_b \tag{4.36}$$

meaning Z acts on all edges labeled by a, and $X^\dagger$ on all edges labeled by b.

This definition extends to the y and z-direction, where partitioning into a and b corresponds to Eq. (4.31) with $(L^n_y)^\dagger = L^n_x L^n_z$. For the y-direction, Fig. 4.1 presents an example $L^n_{y_2}$. With this definition of L^n operators, the equivalent of Theorem 2 holds for the Z_n case. The virtual symmetries of an $N_x \times N_y \times N_z$ Z_n X-Cube tensor network is:

$$G^n_{N_x N_y N_z} = \langle L^n_{x_1}, ..., L^n_{x_{N_x}}, L^n_{z_1}, ..., L^n_{z_{N_z}}, L^n_{y_1}, ..., L^n_{y_{N_y-1}} \rangle. \tag{4.37}$$

Again, due to the faithfulness of the representations,

$$G^n_{N_x N_y N_z} \cong (Z_n)^{\times (N_x+N_y+N_z-1)}, |G^n_{N_x N_y N_z}| = n^{N_x+N_y+N_z-1}. \tag{4.38}$$

Theorem 3 generalizes as well to

$$T^n_{N_x N_y N_z} (T^n_{N_x N_y N_z})^\dagger = P^n_{N_x N_y N_z} = \frac{1}{|G^n_{N_x N_y N_z}|} \sum_{g \in G^n_{N_x N_y N_z}} g. \tag{4.39}$$

Eqs. (2.60) to (2.64) hold for the Z_n case, where the L operators are replaced with their generalized L^n operators and $a_i, b_j, c_k \in \{0, 1, 2, ..., n-1\}$. With that, all the properties required in the proof of Theorem 4 generalize to the Z_n case. Consequently, the Intersection Property holds for Z_n X-Cube tensor networks as well. By generalizing the operators E^i_g, the decomposition of symmetries into tensor products acting on the edges

in the different directions generalizes to

$$g = E_g^x \otimes (E_g^x)^\dagger \otimes E_g^y \otimes (E_g^y)^\dagger \otimes E_g^z \otimes (E_g^z)^\dagger. \tag{4.40}$$

Similarly to the Z_2 case, these correspond one to one to elements in G^n/G_x^n, where G_x^n denotes the generalization of G_x, which is done by allowing $a \in \{0, 1, ..., n-1\}$ in the Definition 5. Theorem 5 generalizes to the Z_n X-Cube tensor network using the generalized E^i operators.

4.3 Analysis of the Z_n X-Cube Tensor Network

This section discusses the equivalent of Chapter 3 for the Z_n case. The groundspace degeneracy of the modified Z_n X-Cube Hamiltonian is discussed, followed by an analysis of Z and X excitations. Finally, the layered toric code structure is discussed.

4.3.1 Groundspace Degeneracy of the Z_n X-Cube Tensor Network

Since the Intersection Property holds for the Z_n case, the generalization of Theorem 6 and its proof are straightforward. Similarly, the argument for Eq. (3.7) generalizes to the Z_n case, for which:

$$\text{GSD}_{H^n} = |G^n/G_x^n| + |G^n/G_y^n| + |G^n/G_z^n| = n^{2(N_x+N_y+N_z)-3}. \tag{4.41}$$

Loops of Z operators along the holes of the three-torus $\mathbb{T}^3$ are logical operators, as in the Z_2 case.

4.3.2 Z and X Excitations of the Z_n X-Cube Tensor Network

In the Z_2 case, a cube of the tensor network is said to carry an excitation if an odd number of Z operators act on its edges. For the Z_n generalization, a weighted product of $Z^k, k \in \{0, 1, 2, ..., n-1\}$ edge operators is introduced that measures the carried Z excitation of every cube.

Definition 8. *Consider a cube c in the tensor network. We divide the operators acting on the edges of the cube into two sets, denoted by A (red) and B (blue). For a fixed orientation of the lattice, the partition is made according to Fig. 4.2. We then define the product π_c:*

$$\pi_c = \prod_{a \in A} O_a \prod_{b \in B} O_{e'}^\dagger, \tag{4.42}$$

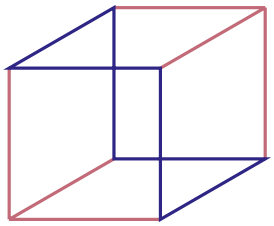


Figure 4.2: Operators on the red edges appear without a dagger in π_c, whereas operators on the blue edges appear with a dagger.

where O_a and O_b are operators of the form $Z^k, k \in \{0, 1, 2, ..., n-1\}$.

Since $\pi_c = Z^k$ for some $k \in \{0, 1, ..., n-1\}$, the excitation of a cube is no longer a binary, but an n-ary variable for the Z_n case. Moreover, π_c is invariant under the application of $g \in G^n_{111}$ applied to any vertex of the tensor network. For any groundstate represented by the Z_n X-Cube tensor network:

$$\forall c : \pi_c = \mathbb{1}. \tag{4.43}$$

Introducing a single Z operator into the network excites the four surrounding cubes, as can be seen below.

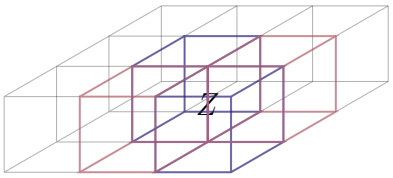


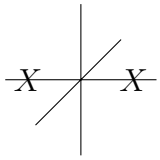
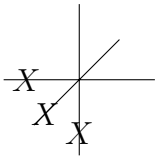
Figure 4.3: The central Z excites the four surrounding cubes, whose product π_c is indicated by their coloring: blue corresponds to $\pi_c = Z^\dagger$ and red corresponds to $\pi_c = Z$.

The discussion on the mobility of these excitations, see Section 3.2.1, is analogous to the $n = 2$ case. Single excited cubes are immobile, since any Z changes the value of π_c on all four surrounding cubes. Pairs of excited cubes, differently colored in the Z_n case are mobile within the plane corresponding to their common face.
Virtual X excitations in the Z_n X-Cube model behave similarly to the ones discussed in Section 3.2.2. In the Z_2 case every plane had to have an odd number of X operators, see Eq. (3.8). In the Z_n X-Cube tensor network, the generalization of this exclusion argument is slightly more complicated since X operators moved along an edge become $X^\dagger$. The X operators in a plane must come in configurations that commute through the local symmetries of T^n, or the argument of the Z_2 case applies. Locally, this constrains

physical				*virtual*			
Z	$Z^\dagger$	$Z^\dagger$	Z	I	I	I	I
X	X	I	I	$X^\dagger$	$X^\dagger$	I	I
I	I	X	X	I	I	$X^\dagger$	$X^\dagger$
X	I	X	I	$X^\dagger$	I	$X^\dagger$	I
Z	I	I	I	$Z^\dagger$	I	I	I
I	Z	I	I	I	$Z^\dagger$	I	I
I	I	Z	I	I	I	$Z^\dagger$	I
I	I	I	Z	I	I	I	$Z^\dagger$

Table 4.2: Symmetries of the Z_n toric code tensor $T^{TC,n}$.

the generation of virtual Z operators to operations generated by:

$K_1^n =$  , $K_2^n =$  (4.44)

With that, the mobility and merging properties of lineons discussed in Section 3.2.2 generalize to the Z_n case. In order to move an excitations within a line, K_1^n is applied repeatedly. Note, that the X operators of K_1^n represents a $X^\dagger$ acting on the dual vector space to the left. Therefore, applying it repeatedly will cancel the X and $X^\dagger$ between the endpoints. K_2^n describes how X and $X^\dagger$ operators coming from different directions can merge, as in the Z_2 case.

4.3.3 Layer Structure of the Z_n X-Cube Tensor Network

In order to see the layered toric code structure of the Z_n X-Cube tensor network, the toric code is generalized in a very similar way. Following the same strategy used to generalize the X-Cube tensor, we obtain the Z_n toric code tensor $T^{TC,n}$:

$$T^{TC,n} = \sum_{i_1,i_2,i_3,i_4=0}^{n-1} P^{TC,n} \left|i_1,i_2,i_3,i_4\right\rangle \otimes \left\langle i_1,i_2,i_3,i_4\right| , \tag{4.45}$$

where

$$P^{TC} = \sum_{k=0}^{n-1} Z^k \otimes (Z^\dagger)^k \otimes (Z^k)^\dagger \otimes Z^k. \tag{4.46}$$

The symmetries of $T^{TC,n}$ are listed in Table 4.2. All properties of the Z_2 toric code generalize, as the properties of the Z_2 X-Cube model do.

In the generalized procedure of adding a layer to the X-Cube tensor network, described in Section 3.3, we make use of the twisted CX gate:

$$\tilde{CX} = (\mathbb{1} \otimes H^2)(CX)(\mathbb{1} \otimes H^{-2}). \tag{4.47}$$

The relevant differences to the CX gate in the following computations are

$$\tilde{CX}(X \otimes \mathbb{1})\tilde{CX}^\dagger = (X^\dagger \otimes X) \tag{4.48}$$

and

$$\tilde{CX}(\mathbb{1} \otimes Z)\tilde{CX}^\dagger = (Z \otimes Z). \tag{4.49}$$

In Fig. 4.4, the operation of coupling a toric code tensor to an existing X-Cube tensor is shown for the Z_n X-Cube tensor network.

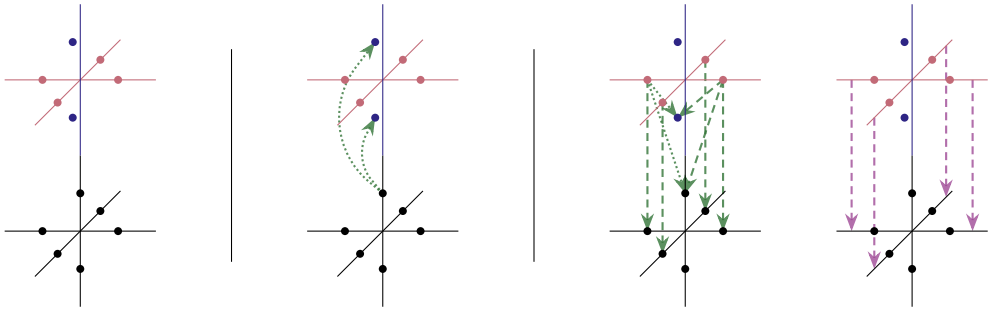


Figure 4.4: The left side presents the setup before the application of CX and $\tilde{CX}$ gates. The added toric code layer and the added $|0\rangle$ states are shown in red and blue respectively. The picture in the center and the two on the right side show the series of CX and $\tilde{CX}$ gates. CX gates are displayed as dotted arrows, while $\tilde{CX}$ gates are shown as dashed arrows. Their color indicates whether they act on the physical (green) or virtual (violet) level.

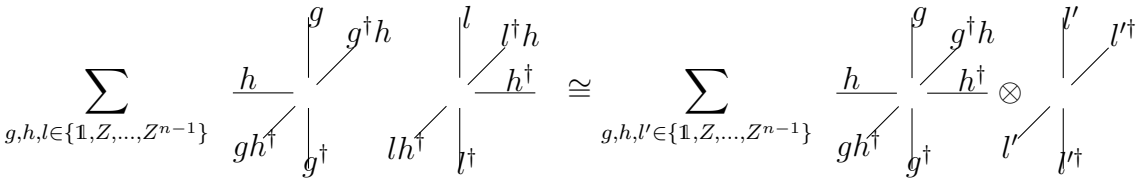
This set of gates transforms the symmetry generators of the Z_n X-Cube tensor, the Toric code tensor and the Z symmetries of the $|0\rangle$ states to the symmetry generators of the T^n_{112}, the Z_n X-Cube tensor network of size $1 \times 1 \times 2$.
As in Section 3.3, we can also see this pattern on the virtual level alone. The isomorphism

$$P^n_{211} \cong P^n_{111} \otimes P^{TC,n}_{011} \tag{4.50}$$

is realized by application of virtual $\tilde{CX}$ gates in the pattern of virtual gates depicted above. For that we use:

$$\tilde{CX}(g \otimes \mathbb{1})\tilde{CX}^\dagger = (g \otimes g^\dagger), \tag{4.51}$$

where $g = Z^k, k \in \{0, 1, ..., n-1\}$



(4.52)

That concludes the analysis of the Z_n X-Cube tensor network. We have shown that the topological groundspace degeneracy, the fractal nature of excitations and the layered toric code structure hold for the chosen generalization.

5 Conclusion

In this thesis, a tensor network representation of the X-Cube model was constructed and analyzed, offering a new perspective on fracton topological phases through tensor network methods. The representation captures key structural and entanglement properties of the ground state using a locally defined tensor network, enabling a more transparent view of the model's features.

The study focused on identifying and analyzing the symmetries present in the tensor network, which shows how they encode fundamental aspects of the X-Cube phase. Three defining features were demonstrated within this framework: the sub-extensive scaling of ground state degeneracy on the torus, the presence of excitations with restricted subsystem mobility and the layered structure resembling stacks of two-dimensional toric code systems.

Furthermore, a generalization of the construction to a Z_n version of the X-Cube model was proposed. This extension preserves the essential properties observed in the Z_2 case, indicating the robustness of the tensor network approach across different symmetry groups.

In summary, this thesis contributes a concrete tensor network construction of the X-Cube model and highlights how key features of fracton order are reflected in the network's structure. These results provide a solid foundation for future work on fractonic tensor networks, including extensions to other models or deeper analysis of their entanglement and symmetry properties.

Bibliography

[1] Xie Chen et al. “Symmetry protected topological orders and the group cohomology of their symmetry group”. In: *Phys. Rev. B* (2013). DOI: 10.1103/PhysRevB.87.155114.

[2] Clement Delcamp and Norbert Schuch. “On tensor network representations of the (3+1)d toric code”. In: *Quantum* (2021). DOI: 10.22331/q-2021-12-16-604.

[3] Jens Eisert, Marcus Cramer, and Martin Plenio. “Colloquium: Area laws for the entanglement entropy”. In: *Rev. Mod. Phys.* 82 (2010). DOI: 10.1103/RevModPhys.82.277.

[4] Daniel Gottesman. “Stabilizer Codes and Quantum Error Correction”. (1997). arXiv: quant-ph/9705052 [quant-ph].

[5] Daniel Gottesman. “Theory of fault-tolerant quantum computation”. In: *Phys. Rev. A* (1998). DOI: 10.1103/PhysRevA.57.127.

[6] Jeongwan Haah. “Local stabilizer codes in three dimensions without string logical operators”. In: *Phys. Rev. A* (2011). DOI: 10.1103/PhysRevA.83.042330.

[7] Matthew Hastings. “An area law for one-dimensional quantum systems”. In: *Journal of Statistical Mechanics: Theory and Experiment* (2007). DOI: 10.1088/1742-5468/2007/08/P08024.

[8] Alexei Yu Kitaev. “Fault-tolerant quantum computation by anyons”. In: *Annals of physics* (2003). DOI: 10.1016/S0003-4916(02)00018-0.

[9] Han Ma et al. “Fracton topological order via coupled layers”. In: *Phys. Rev. B* (2017). DOI: 10.1103/PhysRevB.95.245126.

[10] David Poulin et al. “Quantum Simulation of Time-Dependent Hamiltonians and the Convenient Illusion of Hilbert Space”. In: *Phys. Rev. Lett.* (2011). DOI: 10.1103/PhysRevLett.106.170501.

[11] Norbert Schuch, Ignacio Cirac, and David Pérez-García. “PEPS as ground states: Degeneracy and topology”. In: *Annals of Physics* (2010). DOI: 10.1016/j.aop.2010.05.008.

[12] Wilbur Shirley et al. “Fracton Models on General Three-Dimensional Manifolds”. In: *Phys. Rev. X* (2018). DOI: 10.1103/PhysRevX.8.031051.

[13] Frank Verstraete and Ignacio Cirac. “Matrix product states represent ground states faithfully”. In: *Phys. Rev. B* (2006). DOI: 10.1103/PhysRevB.73.094423.

[14] Frank Verstraete et al. “Criticality, the Area Law, and the Computational Power of Projected Entangled Pair States”. In: *Phys. Rev. Lett.* 96 (2006). DOI: 10.1103/PhysRevLett.96.220601.

[15] Sagar Vijay, Jeongwan Haah, and Liang Fu. “Fracton topological order, generalized lattice gauge theory, and duality”. In: *Phys. Rev. B* (2016). DOI: 10.1103/PhysRevB.94.235157.