Università degli Studi di Padova

Department of Physics and Astronomy "Galileo Galilei"

Master Thesis in Physics of Data

Non-Abelian Tensor Network Methods for Many-body Quantum Systems

Supervisor
Prof. Simone Montangero
Università degli Studi di Padova

Co-supervisor
Dr. Daniel Jaschke
Ulm University

Master Candidate
Gerardo Carmona

Student ID
2005005

Academic Year
2021-2022

To my parents, my siblings, and my girlfriend. The warmth of your affection and support was far wider than the ocean between us.

You encompass radical love.

Acknowledgments

My gratitude goes to Professor Simone Montangero and Dr. Daniel Jaschke for their support and mentorship throughout the development of this work and even before it started.

I also thank Dr. Pietro Silvi for his insightful suggestions on relevant models to explore with our implementation.

Abstract

Non-Abelian symmetries are one of the most challenging forms of symmetries to be used in tensor network methods. Their promised advantages in terms of accessible physical phenomena and computational scaling makes them a meaningful tool for many-body quantum physics. We implement tensor network methods with non-Abelian symmetries to explore the bilinear biquadratic spin chain model for up to 50 sites, and compare their benefits with respect to the same system simulated without symmetries. We motivate the Hamiltonian and highlight the benefits in terms of group theory in the study of many-body systems in one dimension and possibly beyond. In particular, we integrate the non-Abelian symmetry into the existing library Quantum TEA, which consists of over twenty modules and covered up to now only Abelian symmetries. The integration exploits code design principles ranging from unit testing to dependency management.

Contents

Listing of figures

1 Introduction

Quantum computers, devices that perform numerical tasks with quantum bits, i. e. qubits, instead of bits, have captivated the efforts and resources of academia, public and private sectors. From emulator systems [1] to quantum circuits [2, 3], the possibility of entanglement between qubits unlocks huge dimensional spaces for the system state in-between gates up until measurement [4]. Since the operational principles of quantum computers are those of quantum systems, these devices could outperform classical computation in quantum many-body (QMB) simulations by means of analogue models in quantum processing units as envisioned by R. Feynman [5].

Advantages in problems beyond quantum systems simulations are also theoretically achievable. Shor's algorithm [6, 7], for example, could perform large prime factorization in polynomial time on the input's order. Item search in unstructured databases can be achieved with $\sqrt{N}$ complexity using Grover's algorithm [8]. There are also proposals to exploit the potential of quantum computers in discrete optimization tasks [9, 10].

Although promising, even the Noisy-Intermediate Scale Quantum era (NISQ) [11] for quantum technologies is yet to come. The study of quantum systems relying on classical resources remains essential for developing general quantum technologies. In the pursuit of modelling QMB systems with strong correlations, clever numerical strategies, such as density functional theory [12] or quantum Monte Carlo [13],

offer remarkable results. However successful these numerical procedures are, their shortcomings remain an issue [14].

Even if not essential to quantum advantage [15], entanglement is a distinguishing feature of quantum computers and thus one of the main challenges in the study of quantum systems on a classical computer. The full description of a system with N qudits, quantum bodies that can be observed in one out of d different states, implies an a priori d^N-dimensional Hilbert space [16]. Assuming single precision complex entries, more than eight thousand terabytes would go into the description of a generic, non-collapsed state for a system of 50 qubits. We are thus confronted with the so-called curse of dimensionality, which quickly rules out straightforward exact diagonalisation. Even the identification of symmetries, which often results in insightful constraints that prune the search space, falls short behind the growth of these exponentially scaling vector spaces.

Tensor network methods stand out as an approach that capitalises on the theory of quantum information [17, 18]. Rooted in the developments of Wilker [19], and White [20], and further formalized towards a class of solutions by the contributions of Vidal [21] and Schollwöck [22], tensor networks (TNs) furnish a variety of applications. The development of TNs has impacted the condensed matter field [23], quantum computation [24], as well as high energy physics [25]. In the growing field of machine learning, TNs have drawn attention due to their hyperparameter calibrating features [26]. The notation of tensor network methods provides a clear, diagrammatic perspective on the manipulation of quantum states and operators [27]. This intuitiveness may in part explain their growing success in the cited fields: the assumptions on model interactions are manifest in their structure.

With contemporary computational power, and the development of processing units particularly fit for tensor operations, such as Tensor Processing Units (TPUs) or high-end Graphical Processing Units (GPUs), TNs are a balanced and necessary approach to the simulation of many body systems, as they allow the exploitation of computational resources in an optimized fashion.

A growing variety of problems can be tackled by means of TNs. Ground state search is possible by means of either variational approaches [23] or imaginary time evolution, such as with the TEBD algorithm [28], which, as implied by the name, also enables unitary time evolution. Evaluation of observables other than energy is also possible, with loop-free TNs being particularly well-suited for efficient evalua-

tion of local observables and tensor product observables [29]. Moreover, correlators, which are of utmost relevance in the study of many-body quantum systems, can be measured efficiently by means of appropriate contraction procedures with tensor networks [30]. These features are not limited to ground states, since excited states can be obtained from the ground-state ansätze [31].

Tensor networks are best-suited for QMB low-dimensional lattice systems. Among such systems, we encounter the Bilinear Biquadratic (BLBQ) model [32, 33]. This chain model is of theoretical and practical relevance because of its variety of phase transitions, and the predicted Haldane phase [34]. Furthermore, promising proposals for quantum processors rely on qutrits [35, 36], embodied by spin-1 systems, whose interaction is approximated by the BLBQ model.

Although the nearest-neighbour BLBQ model proposes a spin interaction more elaborate than, e. g. the Heisenberg model [37], it still exhibits SU(2) invariance, both global and pointwise. In light of this trait, addressing the BLBQ model with a symmetry-aware approach is essential for an accurate assessment.

The explicit introduction of Abelian symmetries into tensor networks has proved advantageous resource, performance and accuracy wise [29]. Already efficient basic operations in tensor networks can be optimised further given the resulting structures, which are parallelisable symmetry-sector decompositions. Once non-Abelian symmetries are also taken into account [38], we can exploit further results from representation theory, leading to yet more efficient manipulations, for each instance in the degeneracy of a given sector accounts for multi-dimensional subspaces [39], instead of only a one-dimensional subspace for each instance of degeneracy.

In this work, we consider the different ingredients required for the construction of non-Abelian symmetric tensor network states, with a special focus on the Matrix Product State (MPS) [40] and Tree Tensor Network (TTN) ansatz [41], highlighting the emergence of both as intuitive decompositions.

The efforts hereby reported are part of the multipurpose tensor network project, the Quantum TEA Library. In light of this, the search for numerical stability, efficiency, modularity and in general, the best practices of scientific software development play a central role throughout the implementation. Building on top of the developments on non-Abelian TNs in Ref. [42], we take further steps in the development towards test coverage. There are already useful references for the programming of non-Abelian symmetric TNs; see for example Ref. [43]. We believe that our

exposition underlines the theoretical aspects relevant not only to address the object-oriented definition of non-Abelian tensor networks but also to establish a framework inspired by test-driven development (TTD), in the pursuit of maintainable, reliable software. We also build towards increased compatibility with TN structures, like the TTN, to exploit efficient, pre-defined procedures. Furthermore, we underline manipulation steps which are often overlooked as implementation-specific or obvious.

Chapter 2 is a review of symmetries in quantum systems, with a focus on the key results of representation theory. This chapter introduces the relevant terminology, theorems and notation that go into the modelling of symmetric tensor network states. We devote chapter 3 to general TNs, detailing their justification from a quantum information point of view, to then list the standard operations and manipulations. After discussing different classes of loop-free TNs, we wrap up that section with an example of time evolution and ground-state search algorithm: TEBD. With these foundations, we finally present non-Abelian symmetric TNs in chapter 4, where we point out their analogies with regular TNs as well as their differences. We test our implementation with a SU(2) symmetry and contrast the behaviour described by TEBD in a regime of sites beyond exact diagonalisation. Specifically, we tackle the BLBQ model as an application in the study of phase transitions. We simulate for up to 50 qutrits, and study the convergence for a growing bond dimension. Along with closing remarks and some implementation details, in chapter 5, we set forth an outlook of potential improvements based on the current design choices of the implementation, as well as specific model simulation requirements.

2

Symmetries in quantum mechanics

It is hard to overstate the impact that the study of symmetries has played in the development of theoretical physics. Symmetries result from invariance (or covariance) of equations, variables or, more generally, operators under transformations. In physics, symmetries can often be related to conserved quantities. For example, U(1) invariance is associated with charge (or particle number) conservation, while SO(3) symmetries point to angular momentum conservation.

Transformations can be mapped to the *action of a group*. Since we describe quantum mechanics in Hilbert spaces, understanding how to express the action of a group in vector spaces is key to exploiting the symmetric properties of quantum systems.

In section 2.1 we review the definition of general groups, the notion of linear representations and their connection to vector spaces, and exploit this formalism for the notion of Lie groups, groups from which the physical symmetries we focus on arise. Section 2.2 connects the tensor product of representations with the properties that define a tensor. We present the structural implications of the fusion of irreducible representations, which results in Clebsch-Gordan coefficients. To wrap up, in section 2.2.3 we focus on the decomposition that follows from the Wigner-Eckart theorem, which is of central relevance in the study of quantum systems.

2.1 Group theory

Groups are algebraic structures formed by a set $\mathbb{G}$ and an associative binary operation $\circ$ which together satisfy the following axioms:

1. *closure*: $\forall\ a, b \in \mathbb{G},\ a \circ b \in \mathbb{G}$.
2. *neutral element*: $\exists\ e \in \mathbb{G} \mid \forall\ a \in \mathbb{G},\ e \circ a = a \circ e = a$
3. *inverse element*: $\forall a \in \mathbb{G},\ \exists a^{-1} \in \mathbb{G} | a \circ a^{-1} = a^{-1} \circ a = e$.

This small collection of axioms endows the construct of a group with useful properties. As they stand, groups comprehend a wide variety of systems.

As may immediately come to mind, the pair $(\mathbb{R}, +)$, that is, real numbers under addition, form a group. In fact, this group is of *Abelian* nature, given the commutation of all elements. In contrast, a *non-Abelian* group is such that not all of its elements commute. This feature plays a key role in the applications of group theory to quantum systems.

Yet another Abelian group is formed when we consider the set $\mathbb{R}^\times \equiv \mathbb{R}/\{0\}$ with the regular product $\times$, where we remove the 0 element, as it is the only element that lacks an inverse under multiplication. When we consider the set $\mathbb{C}$ instead of $\mathbb{R}$, we also form Abelian groups: $(\mathbb{C}, +)$ and $(\mathbb{C}^\times, \times)$, which combined form the *field* $(\mathbb{C}, +, \times)$. In a field, a set $\mathbb{F}$ with two binary operations, known as *addition* and *product*, are defined so that each operation forms an Abelian group with the set $\mathbb{F}$ (without the addition neutral element in the case of the product group). Furthermore, the two operations are *distributive*: $a \times (b + c) = a \times b + a \times c,\ \forall a, b \in \mathbb{F}$.

A subset $\mathbb{H}$ of $\mathbb{G}$ forms a subgroup of $(\mathbb{G}, \circ)$ if it is also a group under the same operation, $\circ$. Should a subgroup $\mathbb{H}$ of $\mathbb{G}$ be closed under the action of $\mathbb{G}$, that is, $g \circ h \circ g^{-1} \in \mathbb{H}\ \forall h \in \mathbb{H},\ \forall g \in \mathbb{G}$, then $\mathbb{H}$ is an *invariant subgroup* of $\mathbb{G}$.

Let $\mathbb{H}_1$ and $\mathbb{H}_2$ be two subgroups of $\mathbb{G}$. The group $\mathbb{G}$ is a *direct product* $\mathbb{H}_1 \otimes \mathbb{H}_2$ of $\mathbb{H}_1$ and $\mathbb{H}_2$ if

- $\mathbb{H}_1 \cap \mathbb{H}_2 = \{e\}$, the subgroups only share the identity element,
- $\forall\ (h_1, h_2) \in \mathbb{H}_1 \times \mathbb{H}_2,\ h_1 \circ h_2 = h_2 \circ h_1$, the elements of a subgroup commute with those in the other subgroup,
- $\forall g \in \mathbb{G},\ \exists!\ (h_1, h_2) \in \mathbb{H}_1 \times \mathbb{H}_2 \mid g = h_1 \circ h_2$, each element in the group can be expressed in exactly one combination of elements of the subgroups.

We now turn to the set of non-singular N-square matrices over a field $(\mathbb{F}, +, \times)$, with the usual matrix product. We denote such a structure by $\mathrm{GL}(\mathbb{F}, N)$. This structure forms a group, given a square matrix is non-singular if and only if its determinant is different from zero, and for any two N-dimensional square matrices, $\det\{AB\} = \det\{A\}\det\{B\}$. $\mathrm{GL}(\mathbb{F}, N)$ is known as the *general linear group*. Since it is packed with the matrix product, $\mathrm{GL}(\mathbb{F}, N)$ is by construction a non-Abelian group.

Rotations on a plane, given by the matrix

$$R(\theta) = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}, \tag{2.1}$$

form an Abelian subgroup of $\mathrm{GL}(\mathbb{R}, 2)$, since $R(0) = I_2$ is the identity, and,

$$R(\theta_1)R(\theta_2) = \begin{pmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{pmatrix}\begin{pmatrix} \cos\theta_2 & \sin\theta_2 \\ -\sin\theta_2 & \cos\theta_2 \end{pmatrix} \tag{2.2}$$

$$= \begin{pmatrix} \cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2 & \cos\theta_1\sin\theta_2 + \sin\theta_1\cos\theta_2 \\ -\cos\theta_1\sin\theta_2 - \sin\theta_1\cos\theta_2 & \cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2 \end{pmatrix} \tag{2.3}$$

$$= \begin{pmatrix} \cos(\theta_1+\theta_2) & \sin(\theta_1+\theta_2) \\ -\sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) \end{pmatrix} \tag{2.4}$$

$$= R(\theta_1+\theta_2) = R(\theta_2)R(\theta_1). \tag{2.5}$$

This is the special orthogonal group $\mathrm{SO}(2)$.

In three dimensions, rotations form the special orthogonal group $\mathrm{SO}(3)$, a subgroup of $\mathrm{GL}(\mathbb{R}, 3)$. In this case, since general three-dimensional rotations do not commute, $\mathrm{SO}(3)$ is non-Abelian.

2.1.1 Linear representations

Consider a map ρ which transforms elements g of a group $\mathbb{G}$ into linear operators $\rho(g)$, that act on a space $\mathbb{V}$. If the following relation holds

$$\rho(g_1 \circ g_2) = \rho(g_1)\rho(g_2), \ \forall g_1, g_2 \in \mathbb{G}, \tag{2.6}$$

then we have a *linear representation* of the group $\mathbb{G}$, which is a group itself. Notice that, since $\rho(e \circ g) = \rho(e)\rho(g) = \rho(g)$, we necessarily have $\rho(e) = I$, the identity

operator. Furthermore, since $\rho(g)\rho(g^{-1}) = \rho(e) = I$, it follows that $\rho(g^{-1}) = \rho(g)^{-1}$.

Note that there is no requirement on the injectivity of the so-called representation. Neither is there any condition for the dimension of the map's vector space $\mathbb{V}$. Thus, the identity in a vector space of arbitrary dimension, $\rho(g) = I,\ \forall g \in \mathbb{G}$, is a valid representation for any group. This is a *trivial representation*. If instead, we impose the one-to-one property to the representation, we talk about a *faithful representation*. To fulfil the non-commutative property, faithful representations of non-Abelian groups necessarily have $\dim(\mathbb{V}) \geq 2$.

By choosing a basis for the space $\mathbb{V}$, $\rho(\mathbb{G})$ forms a subgroup of $\mathrm{GL}(\mathbb{F}, N)$. The convenience of this is self-evident: many results from linear algebra can be applied to group representations. From the study of vector spaces, we know that similarity transformations take a matrix operator into a different basis. Representations related by a similarity transformation are *equivalent*. Under the right basis, representations may become block diagonal, and thus act invariantly on non-trivial subspaces; such is the defining trait of *reducible representations*: they can be decomposed into smaller representations. Instead, representations for which no non-trivial invariant subspaces exists are *irreducible* (irreps). We make explicit the fact that a representation is irreducible by using Λ instead of ρ.

2.1.2 Schur's Lemma

The beacon of representation theory, Schur's lemma plays a pivotal role in the study of symmetries. This lemma states that, given two irreps $\Lambda_1 : \mathbb{G} \to \mathbb{V}_1 \times \mathbb{V}_1$, $\Lambda_2 : \mathbb{G} \to \mathbb{V}_2 \times \mathbb{V}_2$, and a linear map $\hat{\Phi} : \mathbb{V}_2 \to \mathbb{V}_1$, such that

$$\Lambda_1(g)\hat{\Phi} = \hat{\Phi}\Lambda_2(g),\ \forall g \in \mathbb{G} \tag{2.7}$$

then $\hat{\Phi}$ is either null or an *isomorphism*. In the special case where $\mathbb{V}_1 = \mathbb{V}_2$, a linear operator $\hat{H}$, invariant under the action of the group $\mathbb{G}$, $\hat{H}$ is then either zero or proportional to the identity for an irrep of $\mathbb{G}$. That is, for a representation ρ,

$$\hat{H}\rho(g) = \rho(g)\hat{H},\ \forall g \in \mathbb{G} \tag{2.8}$$

$$\implies \hat{H} = hI,\ \text{when } \rho \text{ is irreducible.} \tag{2.9}$$

If, in contrast, ρ is reducible, then both maps decompose into the following block diagonal matrices

$$\hat{H} = \begin{pmatrix} h_1 I_1 & & & \\ & h_2 I_2 & & \\ & & \ddots & \\ & & & h_N I_N \end{pmatrix} \tag{2.10}$$

$$\rho(g) = \begin{pmatrix} \Lambda_1(g) & & & \\ & \Lambda_2(g) & & \\ & & \ddots & \\ & & & \Lambda_N(g) \end{pmatrix}, \tag{2.11}$$

with I_μ the identity in the space $\mathbb{V}_\mu$ where irrep Λ_μ acts. The notation is deliberate. Assume $\hat{H}$ to be the Hamiltonian of a quantum system, invariant under the unitary transformations ρ. Then, each h_μ has a degeneracy of $\dim(\mathbb{V}_\mu)$. This is a key application of group theory and symmetries to quantum systems. Furthermore, it may be the case that for two μ, ν in the decomposition, $\Lambda_\mu = \Lambda_\nu$, i. e. the same irrep can show up multiple times.

2.1.3 Dual representations

The set of linear maps $f : \mathbb{V} \to \mathbb{C}$ behaves in a way that mirrors the properties of objects $v \in \mathbb{V}$. Consider the basis $|j\rangle$, so that $v = \sum_j v^j |j\rangle$. Since f is linear, we have

$$f(v) = f\left(\sum_j v^j |j\rangle\right) = \sum_j v^j f(|j\rangle) \tag{2.12}$$

in particular, we focus on the function $f = \langle k|$, so that $\langle k| \left(\sum_j v^j |j\rangle\right) = v^k$. A linear combination $f = \sum_k f_k \langle k|$ of such functions is also a linear map f. We can in fact represent any such map in this way, and the set inherits all the properties of a vector space. In light of this, we identify this space as the *dual space* $\mathbb{V}^*$.

A representation can also be defined in a dual space $\mathbb{V}^*$. A dual representation ρ^* can be defined so that, acting on $\langle j|$, the outcome remains dual to $\rho(g)|j\rangle$. In matrix notation $f(v) = f^T v$, the transpose vector f^T multiplies v. Therefore, we see

that

$$(\rho^*(g)f)^T(\rho(g)v) = f^T\rho^{*T}(g)\rho(g)v = f^Tv \tag{2.13}$$

$$\iff \rho^{*T}(g) = \rho(g^{-1}). \tag{2.14}$$

Which under unitary representations, is precisely the complex conjugate, $\rho^{*T}(g) = \rho^\dagger(g)$.

2.1.4 Lie Groups

Starting from the notion of continuity, we focus on unitary transformations arbitrarily close to the identity, so that a first-order approximation is valid

$$W(\epsilon) = I + \mathrm{i}\epsilon X \tag{2.15}$$

$$W^\dagger(\epsilon) = I - \mathrm{i}\epsilon X \tag{2.16}$$

$$W^\dagger W = (I - \mathrm{i}\epsilon X)(I + \mathrm{i}\epsilon X) \tag{2.17}$$

$$= I + O(\epsilon^2) \tag{2.18}$$

where we set X to be Hermitian, and is recovered by $X = -\mathrm{i}\frac{\mathrm{d}W}{\mathrm{d}t}\Big|_{t=0}$. Successive applications of W are equivalent, to first order, to a transformation with parameter $n\epsilon$. Going from an infinitesimal transformation to a finite one, we may then apply the following limit

$$W(t_a) = \lim_{n\to\infty}\left(I + \mathrm{i}\frac{tX}{n}\right)^n \tag{2.19}$$

$$= e^{\mathrm{i}tX}. \tag{2.20}$$

The X is the *generator* of a *Lie group*.

Recall a rotation $R(\theta) \in \mathrm{SO}(2)$ is given as

$$R(\theta) = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}. \tag{2.21}$$

Differentiation w. r. t. θ leads to

$$\left.\frac{\mathrm{d}R(\theta)}{\mathrm{d}\theta}\right|_{\theta=0} = \left.\begin{pmatrix} -\sin\theta & \cos\theta \\ -\cos\theta & -\sin\theta \end{pmatrix}\right|_{\theta=0} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \tag{2.22}$$

$$\implies X = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}. \tag{2.23}$$

Consider now a collection of operators X^a that forms a closed algebra under commutation. These relations are mediated by *structure constants*, compactly denoted by f_c^{ab}:

$$\left[X^a, X^b\right] = \mathrm{i} f_c^{ab} X^c. \tag{2.24}$$

Take $U_1(t_1) = e^{\mathrm{i}t_1 X_1}$, $U_2(t_2) = e^{\mathrm{i}t_2 X_2}$, and $U_3(t_3) = e^{\mathrm{i}t_3 X_3}$. By the Baker-Campbell-Hausdorff formula, successive application of U_3 and U_2 leads to a series of commutations between operators X_a and commutators of them:

$$e^X e^Y = e^Z \tag{2.25}$$

$$Z = X + Y + \frac{1}{2}[X, Y] + \frac{1}{12}[X, [X, Y]] - \frac{1}{12}[Y, [X, Y]] + \dots \tag{2.26}$$

Given the algebra of the generators is closed under commutation, it follows that a general application of the U_a transforms has the following form:

$$e^{\mathrm{i}t_1 X_1} e^{\mathrm{i}t_2 X_2} e^{\mathrm{i}t_3 X_3} = e^{\mathrm{i}(t_1' X_1 + t_2' X_2 + t_3' X_3)} \tag{2.27}$$

$$= e^{\mathrm{i}t' \cdot \mathrm{X}}, \tag{2.28}$$

with t_j generally different from t_j'. We therefore arrive to a multi-parameter generalization of the Lie group, in which an infinitesimal transformation is given as

$$U(\epsilon) \approx I + \mathrm{i}\epsilon \cdot \mathrm{X}, \tag{2.29}$$

from which we recover the X_a by $X_a = -\mathrm{i}\left.\frac{\partial U}{\partial t_a}\right|_{t_a=0}$. These operators are known as the *generators* of the Lie group. As mentioned, they form a closed algebra under commutation, known as the *Lie algebra* of the group.

Now, starting from a general rotation in three dimensions,

$$R(\phi,\theta,\psi)=\begin{pmatrix}\cos\phi\cos\psi-\cos\theta\sin\phi\sin\psi & -\cos\phi\sin\psi-\cos\theta\sin\phi\cos\psi & \sin\phi\sin\theta\\ \sin\phi\cos\psi+\cos\theta\cos\phi\sin\psi & -\sin\phi\sin\psi+\cos\theta\cos\phi\cos\psi & -\cos\phi\sin\theta\\ \sin\psi\sin\theta & \cos\psi\sin\theta & \cos\theta\end{pmatrix},$$

we obtain the following generators:

$$L_x=\begin{pmatrix}0&0&0\\0&0&i\\0&-i&0\end{pmatrix},\qquad L_y=\begin{pmatrix}0&0&-i\\0&0&0\\i&0&0\end{pmatrix},\qquad L_z=\begin{pmatrix}0&i&0\\-i&0&0\\0&0&0\end{pmatrix}. \tag{2.30}$$

Together, these generators form $\mathrm{L}=(L_x,L_y,L_z)$, the angular momentum operator for quantum systems.

Generators are defined by their relation, as described by Eq. (2.24). The generators we have obtained in the form of matrices are in fact representations over specific vector spaces. We distinguish between the representation of a Lie Group $\mathbb{G}$ by $\mathbf{\Pi}\left(e^{\mathrm{i}t\cdot\mathrm{X}}\right)$ and that of the Lie algebra $\mathfrak{g}$, $\pi(X_a),\ X_a\in\mathfrak{g}$, both to a given vector space $\mathbb{V}$.

2.2 Product of representations

We can multiply two representations together by means of a tensor product. For the generators of a Lie group, this is given as

$$\pi_1(X_1)\otimes\pi_2(X_2)=-i\frac{\mathrm{d}}{\mathrm{d}t}\left(\Pi_1\left(e^{itX_1}\right)\otimes\Pi_2\left(e^{itX_2}\right)\right)\Big|_{t=0} \tag{2.31}$$

$$=-\mathrm{i}\lim_{\epsilon\to0}\frac{\Pi_1\left(e^{\mathrm{i}\epsilon X_1}\right)\otimes\Pi_2\left(e^{\mathrm{i}\epsilon X_2}\right)-I_1\otimes I_2}{\epsilon} \tag{2.32}$$

$$=-\mathrm{i}\lim_{\epsilon\to0}\frac{\Pi_1\left(e^{\mathrm{i}\epsilon X_1}\right)\otimes\Pi_2\left(e^{\mathrm{i}\epsilon X_2}\right)+I_1\otimes\Pi_2\left(e^{\mathrm{i}\epsilon X_2}\right)-I_1\otimes\Pi_2\left(e^{\mathrm{i}\epsilon X_2}\right)-I_1\otimes I_2}{\epsilon} \tag{2.33}$$

$$=-\mathrm{i}\lim_{\epsilon\to0}\frac{\Pi_1\left(e^{\mathrm{i}\epsilon X_1}\right)-I_1}{\epsilon}\otimes\Pi_2\left(e^{\mathrm{i}\epsilon X_2}\right)-\mathrm{i}\lim_{\epsilon\to0}I_1\otimes\frac{\Pi_2\left(e^{\mathrm{i}\epsilon X_2}\right)-I_2}{\epsilon} \tag{2.34}$$

$$=\pi_1\left(X_1\right)\otimes I_2+I_1\otimes\pi_2\left(X_2\right) \tag{2.35}$$

This result is the composition rule for observables arising from a Lie algebra on the Hilbert space.

2.2.1 Tensors as operators

With the result of representations product in mind and the description of dual spaces, we can put forward a definition of tensor from the representation theory perspective. Consider a vector space $\mathbb{V}$ is formed by

$$\mathbb{V} = \mathbb{W}_1^* \times \mathbb{W}_2^* \times \cdots \times \mathbb{W}_N^* \times \mathbb{V}_1 \times \mathbb{V}_2 \times \cdots \times \mathbb{V}_M. \tag{2.36}$$

A tensor $\mathcal{T} : \mathbb{V} \to \mathbb{F}$ is a linear map

$$\mathcal{T}\left(\alpha f_1 + \beta f_1', f_2, \ldots, f_N, v^1, \ldots, v^M\right) = \alpha \mathcal{T}\left(f_1, \ldots, f_N, v^1, \ldots, v^M\right) \tag{2.37}$$

$$+ \beta \mathcal{T}\left(f_1', \ldots, f_N, v^1, \ldots, v^M\right), \tag{2.38}$$

for scalars $\alpha, \beta \in \mathbb{F}$. The $f_r \in \mathbb{W}_r^*$, are dual vectors of vector space $\mathbb{W}_r$, while $v_s \in \mathbb{V}_s$. In particular, f_1 and f_1' are both dual vectors of $\mathbb{W}_1$, thus $\alpha f_1 + \beta f_1' \in \mathbb{W}_1^*$. We say a tensor of this sort is of rank (N, M).

Since the map is linear, we can choose bases $\{\langle k_1|\}, \ldots, \{\langle k_N|\}$ and $\{|j_1\rangle\}, \ldots, \{|j_M\rangle\}$, and focus on $\mathcal{T}^{j_1,\ldots,j_N}_{k_1,\ldots,k_M}$, the transformation that each element of the bases undergoes.

A tensor has special transformation properties. The action of a group $\mathbb{G}$ on its components is given by the tensor product representation $R = (\otimes_{k=1}^{N} \rho_k^*) \otimes (\otimes_{j=N+1}^{M+N} \rho_j)$. When the group is a (matrix) Lie group, the representation of the generators iterates what we obtained in the previous section,

$$\pi(X) = \sum_{j=1} I_{1|(j-1)} \otimes \pi_j^*(X) \otimes I_{(j+1)|(N+M)} \tag{2.39}$$

$$+ \sum_{j'=N+1} I_{1|(j'-1)} \otimes \pi_{j'}(X) \otimes I_{(j'+1)|(N+M)} \tag{2.40}$$

with $I_{l|k} = \bigotimes_{j=l}^{k} I_j$, where I_j is the identity operator in the space of the algebra representation j.

2.2.2 Decomposing a product of representations

Consider the product of irreps Λ_a and Λ_b of the same group $\mathbb{G}$, each acting on span $(\{|a\rangle\})$ and span $(\{|b\rangle\})$, respectively. A priori, the resulting representation, Θ_{ab} acts on a space span $(\{|a\rangle \otimes |b\rangle\})$. However, we can often find non-trivial invariant subspaces for Θ_{ab}, leading to

$$\Theta_{ab}(g) = \bigoplus_c \left(\mathbb{I}_{d_{\Lambda_c}} \otimes \Lambda_c(g)\right) \tag{2.41}$$

where Λ_c is an irreducible representation of the product group, and $\mathbb{I}_{d_{\Lambda_c}}$ accounts for the degeneracy of irrep Λ_c.

Given the decomposition, then only a subspace of span $\left(\{\left|j, m_j\right\rangle \otimes \left|j', m_{j'}\right\rangle\}\right)$ is relevant. It is possible to generate bases for these subspaces starting from the tensor product basis. Consider three irreps, Λ_j, $\Lambda_{j'}$ and Λ_J, with orthonormal bases $\{\left|j, m_j\right\rangle\}$, $\{\left|j', m_{j'}\right\rangle\}$ and $\{|J, M_J\rangle\}$, respectively. Then we can expand $|J, M_J\rangle$ as follows

$$|J, M_J\rangle = \sum_{m_a, m_b} C^{F[J], M_J}_{j, m_j; j', m_{j'}} \left|j, m_j; j', m_{j'}\right\rangle \tag{2.42}$$

$$C^{F[J], M_J}_{j, m_j; j', m_{j'}} = \left\langle j, m_j, j', m_{j'} \middle| J, M_J \right\rangle \tag{2.43}$$

The $C^{F[J], M_J}_{j, m_j; j', m_{j'}}$ are known as *Clebsch-Gordan coefficients*. These coefficients in fact form a rank $(2, 1)$ invariant tensor, since

$$C^{F[J], M'_J}_{j, m'_j; j', m'_{j'}} = W^{[J]\ M'_J}_{M_J} W^{*[j]\ m_j}_{m'_j} W^{*[j']\ m_{j'}}_{m'_{j'}} C^{F[J], M_J}_{j, m_j; j', m_{j'}} \tag{2.44}$$

Furthermore, a map $\mathcal{X}^{F[J], t_J}_{j, t_j; j', t_{j'}}$ is introduced to distinguish between degenerate contributions:

$$\mathcal{X}^{F[J], t_J}_{j, t_j; j', t_{j'}} = \begin{cases} 1 & t_j, t_{j'} \text{ contribute to } t_J \\ 0 & \text{otherwise.} \end{cases} \tag{2.45}$$

The tensors $\mathcal{X}^F$ and C^F combine to form Υ^F:

$$\Upsilon^F = \bigoplus_{[j, j', J]} \mathcal{X}^{F[J]}_{j, j'} \otimes C^{F[J]}_{j, j'} \tag{2.46}$$

We can also expand a product basis element in terms of the fused basis:

$$\left|j, m_j; j', m_{j'}\right\rangle = \sum_{M_J} C_{J,M_J}^{S[j,m_j;j',m_{j'}]} \left|J, M_J\right\rangle \tag{2.47}$$

And of course,

$$\mathcal{X}_{J,t_J}^{S[j,t_j;j',t_{j'}]} = \begin{cases} 1 \ t_J \text{ contributes to } t_j, t_{j'} \\ 0 \text{ otherwise.} \end{cases} \tag{2.48}$$

which together lead to an invariant split tensor Υ^S

$$\Upsilon^S = \bigoplus_{[j,j',J]} \mathcal{X}_{J,t_J}^{S[j,t_j;j',t_{j'}]} \otimes C_{J,M_J}^{S[j,m_j;j',m_{j'}]}. \tag{2.49}$$

For some groups decomposition rules are well known, e. g. for two irreps l and l' of the special unitary group, SU(2),

$$l \otimes l' = (|l - l'|) \oplus (|l - l'| + 1) \oplus \cdots \oplus (l + l'). \tag{2.50}$$

However, for the decomposition of compact Lie groups irrep. products, useful algorithms such as [39] can be used to find both the irreps and the Clebsch-Gordan coefficients.

2.2.3 Wigner-Eckart Theorem

When we deal with operators T that are invariant under SO(3) or $SU(2)$ transformations, Wigner-Eckart theorem states that we can decompose them as

$$T_{i_1,i_2,i_3} = \mathcal{R}_{j,t_j;j',t_{j'}}^{[J],t_J} C_{j,m_j;j',m_{j'}}^{F[J,m_J]}, \tag{2.51}$$

where $C_{j,j'}^{F[J]}$ are the Clebsch-Gordan coefficients to fuse j and j' into J, and $\mathcal{R}_{j,j'}^{[J]}$ is a *reduced tensor*, a tensor that contains information not covered by the symmetry sector. This is reminiscent of the decomposition in Eq. (2.52). We can therefore

state invariant *tree-tensors* as:

$$\mathcal{T} = \bigoplus_{[j,j',J]} \mathcal{R}^{F[J]}_{j,j'} \otimes \mathcal{C}^{F[J]}_{j,j'}. \tag{2.52}$$

3

Tensor Networks

The pure state of a quantum system is described by the wavefunction $|\psi\rangle$ in a Hilbert space $\mathcal{H}$. Assuming $\mathcal{H} \cong \mathbb{V}_1 \otimes \cdots \otimes \mathbb{V}_2$, a generic superposition of states in an N-sites quantum system can be described as a linear expansion over a basis $\{|j_1, ..., j_N\rangle\}$:

$$|\psi\rangle = \sum_{j_1,...,j_N=1}^{d} \mathcal{T}_{j_1,...,j_N} |j_1, ..., j_N\rangle \tag{3.1}$$

The basis $\{|j_1, ..., j_N\rangle\}$ naturally arises as the tensor product between the bases of each $\mathbb{V}_n$ in the decomposition, that is, $|j_1, ..., j_N\rangle = \otimes_{\mu=1}^{N} |j_\mu\rangle$. The symbol $\mathcal{T}_{j_1,...,j_N}$ which represents the coefficients of the expansion, is a map to $\mathbb{C}$ from $\mathbb{J}_1 \times \cdots \times \mathbb{J}_N$, which for simplicity we assume to be d-dimensional spaces.

Let $\mathbb{V}_n$ be the space where an irrep ρ_n of $\mathbb{G}$ acts. Then, unitary transformations are of the form $\rho_n(g) |j_n\rangle$. Now, in the conjugate state

$$\langle\psi| = \sum_{j_1,...,j_N=1}^{d} \mathcal{T}^*_{j_1,...,j_N} \langle j_1, \ldots, j_N| , \tag{3.2}$$

group $\mathbb{G}$ acts as $\langle j_n| \rho_n(g)^\dagger$. The structure of $\mathcal{T}_{j_1,...,j_N}$ therefore fits our notion of tensor expressed on a specific basis.

3.1 Tensor network notation and operations

We regard each index as a link, which results in the diagrammatic rendition of tensors in Fig. 3.1. The number of links is the tensor's rank, and each link i acts on a space $\mathbb{V}_i$. In this notation, we can represent tensors that arise from the contraction

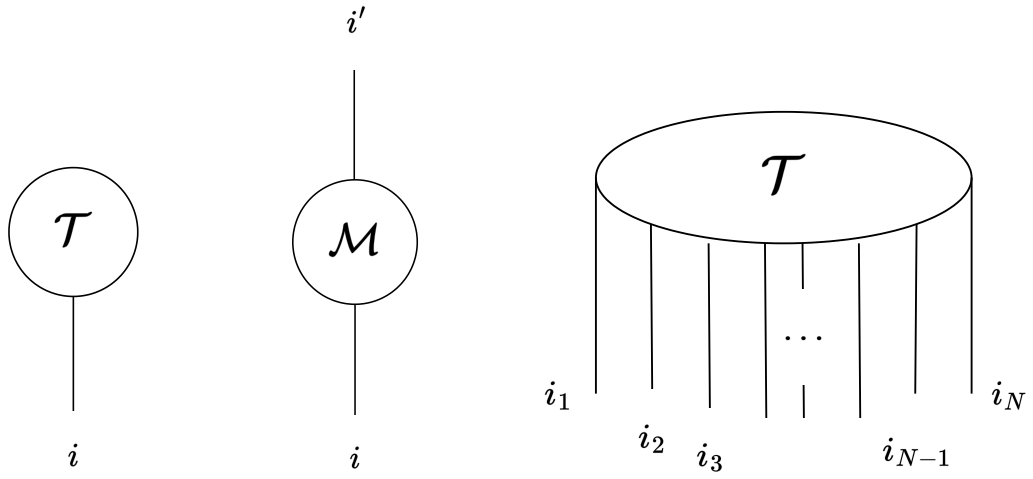

Figure 3.1: A vector is a tensor with only one link, whereas a matrix has two links. A generic tensor state of an N-site system, has a link i_n for each index.

of two (or more) tensors, e. g.

$$\mathcal{T}_{i_A,i_B,i_C,i_D,i_E} = \sum_i \Psi_{i_A,i,i_B} \Phi_{i,i_C,i_D,i_E}, \tag{3.3}$$

as Fig. 3.2. For the time being, we do not distinguish between covariant and con-

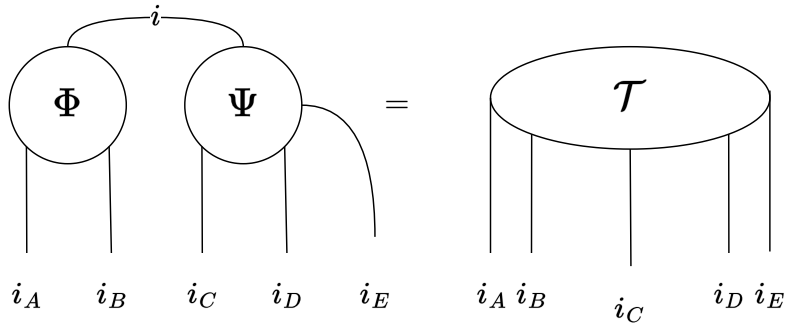

Figure 3.2: A tensor that results from contracting two tensors together.

travariant indices, and thus a rank-(n, m) tensor is described simply as rank-$(m+n)$. Duality of the link space is assumed for contractions.

A generic unitary transformation U can also be represented in this notation, Fig. 3.3. Whereas the $U = W_1 \otimes \cdots \otimes W_N$ transformation is conveniently depicted

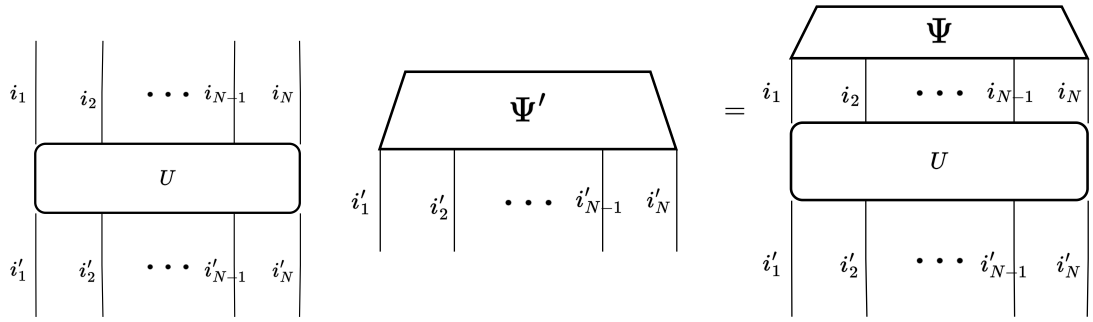

Figure 3.3: The diagrammatic representation of a unitary transformation.

as Fig. 3.4. This depiction of separability also applies to states, such as a three-qubit

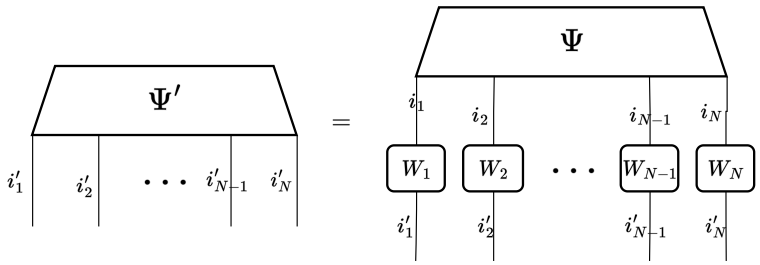

Figure 3.4: The diagrammatic representation of a group transformation. Decompositions into tensor products of different subsystems reduce to separate link actions.

state of the form:

$$|\psi\rangle = \frac{1}{\sqrt{2}}\left(\alpha\,|000\rangle_{ABC} + \alpha\,|011\rangle_{BC} + \beta\,|100\rangle_{ABC} + \beta\,|111\rangle_{ABC}\right) \tag{3.4}$$

$$= (\alpha\,|0\rangle_A + \beta\,|1\rangle_A) \otimes \frac{1}{\sqrt{2}}\left(|00\rangle_{BC} + |11\rangle_{BC}\right) \tag{3.5}$$

$$= (\alpha\,|0\rangle_A + \beta\,|1\rangle_A) \otimes \left|\Phi^+\right\rangle_{BC}. \tag{3.6}$$

This state is separable with respect to subsystem A, in contrast with the bipartite system BC. In fact, $|\Phi^+\rangle$ is a *Bell state*, which presents maximum entanglement.

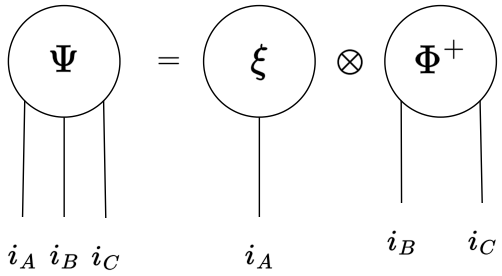

Figure 3.5: A state of three subsystems separable with respect to subsystem A, and fully entangled in BC.

With the listed depictions in mind in mind, we represent arbitrarily complex tensor decompositions, like in Fig. 3.6. We identify the "links and nodes" description of tensors as *tensor networks*.

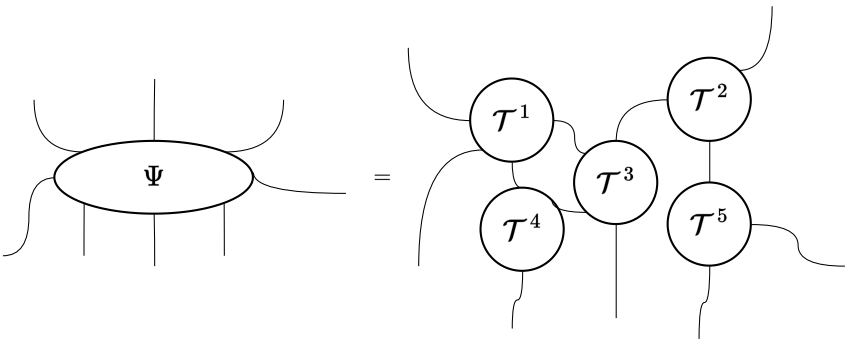

Figure 3.6: A tensor may decompose into an elaborate network.

3.1.1 Fundamental TN Operations

There exists a collection of low-level operations in the manipulation of tensor networks. These operations emerge frequently in many of the applications, and for this reason, their identification and potential optimization become essential in the development of TN numerical routines.

Split

The generic tensor of Eq. (3.1) comes from an expansion over the basis $|j\rangle = \{|j_1, j_2, \ldots, j_N\rangle\}$. When we make explicit the N subsystems that come from the sites, or when we treat it as a bipartite system, we are met with equivalent descriptions of the state. To obtain either description, we *split* the j link.

We can split a given link j into two or more links $j_1, j_2, \ldots, j_N$, as long as the following relation holds,

$$\dim(j) = \dim(j'_1) \times \dim(j'_2) \times \cdots \times \dim(j'_N). \tag{3.7}$$

This operation is computationally equivalent to reshaping an array. For this reason, it can be performed very efficiently, in negligible $O(1)$. Assuming, e. g., a column-major storing, $\Psi_{i_1,(i_4 \cdot d_{i_4}+i_5-1),i_3} = \Psi'_{i_1,i_4,i_5,i_3}$.

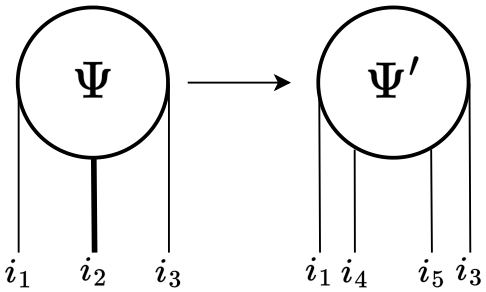

Figure 3.7: The link i_2 is split into the links i_4 and i_5. The resulting links satisfy $d_{i_4} \times d_{i_5} = d_{i_2}$.

Fusion

Likewise, two or more links can be combined into a single one. This is particularly useful when we require a specific rank for our tensor, as for the factorizations we discuss in section 3.1.2. Again, the following relation holds when fuse links $j_1, j_2, \ldots, j_N$ into j',

$$\dim(j_1) \times \dim(j_2) \times \cdots \times \dim(j_N) = \dim(j'). \tag{3.8}$$

Permutation

We can generalize the transpose operation of a matrix. For traceability, links in a tensor have a given order. We can alter this indexing sequence by means of link permutations. Since we are relocating entries, this operation is bounded linearly on the total size of the tensor, and records are related by

$$\Psi^{\sigma}_{p(i_1,i_2,\ldots,i_n)} = \Psi_{i_1,i_2,\ldots,i_n} \tag{3.9}$$

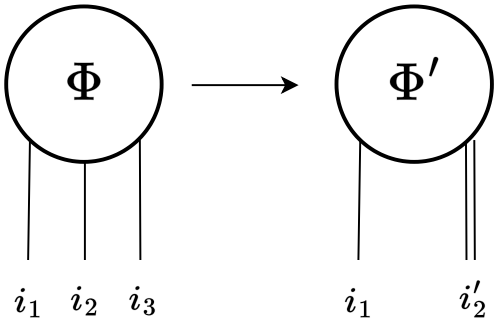

Figure 3.8: Links i_2 and i_3 are fused into link i''_2. These links satisfy $d_{i_2} \times d_{i_3} = d_{i''_2}$.

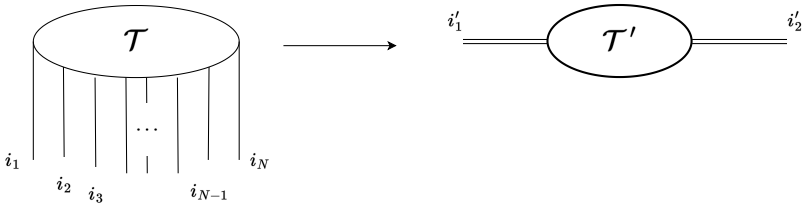

Figure 3.9: By fusing links to either side, a generic tensor can be turned into a matrix.

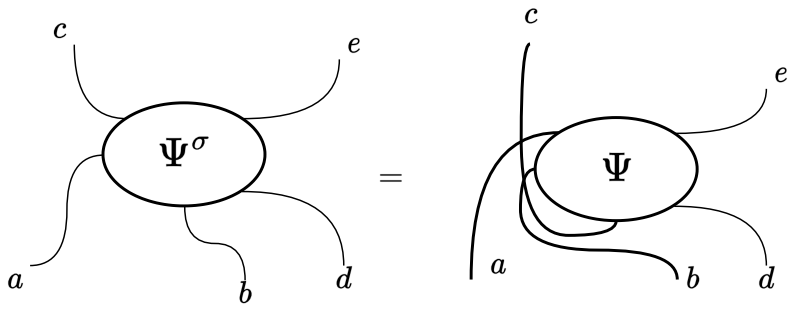

Figure 3.10: Links a, b and c of Ψ^σ and Ψ are related by a cyclic permutation $p(a, b, c, d, e) = (c, b, a, d, e)$.

Contraction

Summation over a given index is essential to the manipulation of tensors. Contraction, which generalizes the matrix product, is a distinguished operation. We can contract two tensors on compatible links, or compatible links within the same tensor.

As we have seen in Fig. 3.2, contraction is fundamental for the construction of tensor networks. Decompositions, norm calculation and trace all arise from contraction, Fig. 3.11.

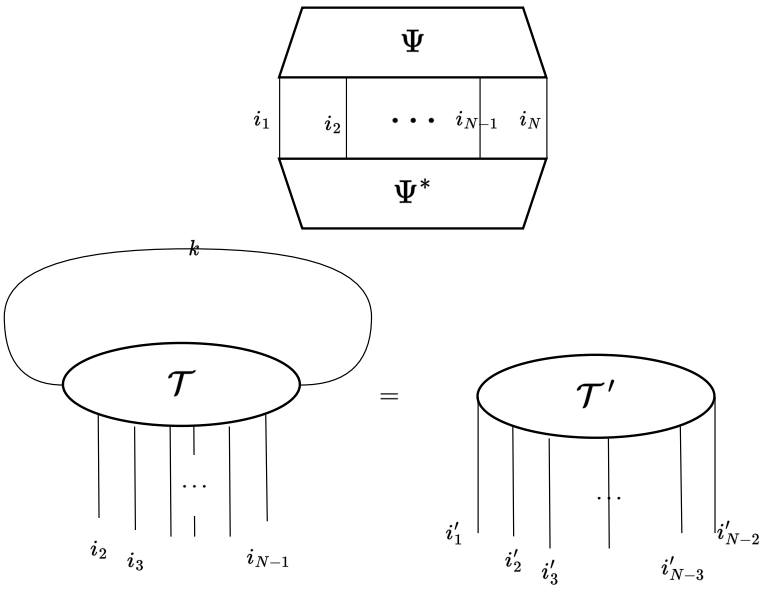

Figure 3.11: Norm calculation is a contraction that leads to a rank-0 tensor, while contracting two indices together in a rank-N tensor reduces its rank to $N-2$.

3.1.2 Decompositions

Rank-2 tensors can be interpreted as matrices, which transform under specific rules. As such, tensors of this sort can be subjected to matrix decompositions, resulting in new tensors, with useful properties such as being unitary, diagonal, or with manifest non-trivial kernels, which can be used to reduce dimensionality. Furthermore, from the basic tensor operations we have described, the links of any tensor can be manipulated to cast the tensor into a convenient rank-2 form, and recover arbitrary rank tensors after the matrix decompositions.

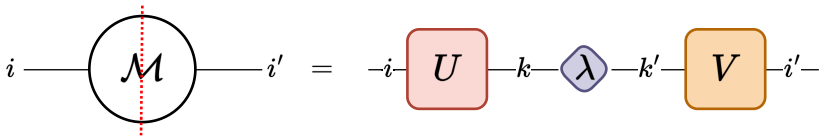

Figure 3.12: Singular values decomposition of a tensor.

Singular Value Decomposition

This decomposition breaks an $n \times m$ matrix into two unitary matrices U, V and a matrix S with a diagonal block and null otherwise,

$$T = USV \tag{3.10}$$

$$S = \begin{cases} \begin{bmatrix} \text{diag}(\lambda) \\ 0 \end{bmatrix}, & m > n \\ \begin{bmatrix} \text{diag}(\lambda), 0 \end{bmatrix}, & m < n, \end{cases} \tag{3.11}$$

where $\text{diag}(\lambda)$ is a diagonal matrix whose entries are those of the λ vector. U is $m \times m$, while V is $n \times n$. The S matrix has shape $m \times n$. The singular values vector, λ is of dimension $\min(m, n)$. Note that, when $m = n$, we get the eigendecomposition and $V = U^\dagger$.

Valuable properties arise from this structure. Consider the squared norm of T:

$$\begin{aligned} \|T\|^2 &= \text{Tr}\{T^\dagger T\} && (3.12) \\ &= \text{Tr}\{V^\dagger S^\dagger U^\dagger USV\} && (3.13) \\ &= \text{Tr}\{S^\dagger S\} = \|\lambda\|^2 && (3.14) \end{aligned}$$

where the last line follows from the unitarity of both U and V. This property can be exploited in tensor networks, setting the *isometry centre* in a convenient way to spare computational resources while normalising states or computing expectations of local observables. As we discuss in section 3.2.1, the entries of λ are real and non-negative, and thus can be sorted. This feature is central in delimiting a relevant subspace of the Hilbert space spanned by the system's tensor product basis.

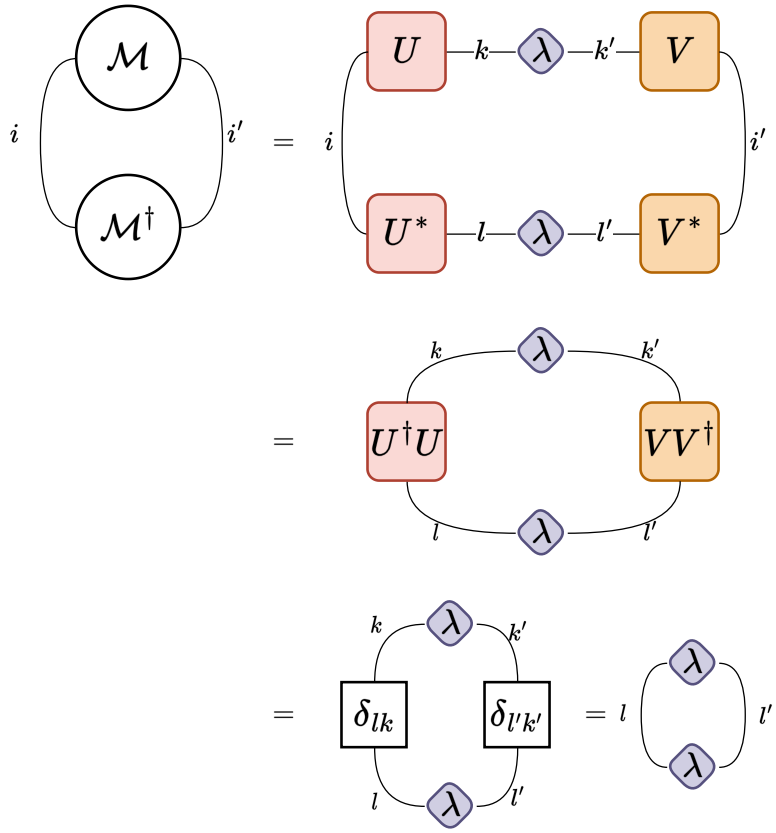

Figure 3.13: The unitarity of tensors in the SVD allows for efficient norm calculation.

QR decomposition

A general QR decomposition of an $n \times m$ tensor T results in a product

$$T = QR \tag{3.15}$$

$$= \begin{bmatrix} Q_1, & Q_2 \end{bmatrix} \begin{bmatrix} R_1 \\ 0 \end{bmatrix} \tag{3.16}$$

$$= Q_1 R_1, \tag{3.17}$$

where Q is an $n \times n$ unitary matrix, while R is an $n \times m$ matrix. The matrix Q_1 is an $n \times m$ matrix, while an upper triangular $m \times m$ matrix, R_1, is the relevant part of R. This decomposition is a computationally less-intensive alternative to SVD. It comes at the expense of the left tensor being only left-unitary, and truncation is limited to that of exact zeros in the decomposition.

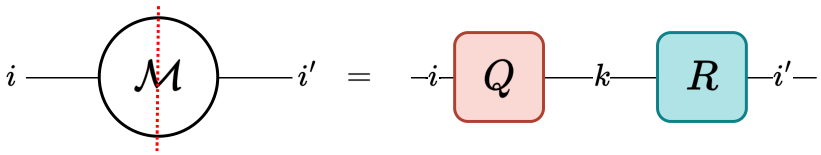

Figure 3.14: In the QR decomposition of an $n \times m$ tensor, the Q tensor is an $n \times n$ unitary matrix, while the R tensor is an $n \times m$ matrix formed by an upper triangular matrix and a 0 rectangular matrix or vector.

3.2 Loop-free Tensor Networks

The absence of loops in a tensor network enables useful properties. To begin with, a finite update sequence is clearly defined, from which follows isometrization of the full network, be it from gauge freedom or sequential decompositions like SVD or QR. Given the absence of cycles, adjacent nodes induce a bipartition of the graph. As we discuss in section 3.2.1, this bipartition of the TN is related to the Schmidt decomposition, and the dimension of the virtual link bounds the Schmidt rank [29].

3.2.1 Schmidt decomposition and entanglement

Consider a bipartite pure quantum state for subsystems A and B, that is, a quantum state $|\Psi\rangle$ that exists in $\mathcal{H}_A \otimes \mathcal{H}_B$. We can perform the Schmidt decomposition:

$$|\Psi\rangle = \sum_a \sum_b \mathcal{T}_{ab} |a\rangle |b\rangle \tag{3.18}$$

$$= \sum_r \lambda_r |\psi_r\rangle |\phi_r\rangle \tag{3.19}$$

$$= \sum_r \sum_a \sum_b \psi_{ra} \lambda_r \phi_{rb} |a\rangle |b\rangle \tag{3.20}$$

where $\lambda_r \geq 0 \ \forall r$. For bipartite separable states, $|\Psi\rangle_{AB} = |\psi\rangle_A |\phi\rangle_B$, the Schmidt decomposition has only one non-zero λ_r. This results in a criterion for non-separable states: entangled bipartite states necessarily have more than one non-zero λ_r. The total number of positive λ_r is known as the *Schmidt rank*.

For pure states in bipartite systems, the Von Neumann entropy gives us a measure

of entanglement, which simplifies to

$$\mathcal{E}\{|\psi\rangle\} = -\sum_r \lambda_r^2 \log \lambda_r^2, \tag{3.21}$$

where $0 \leq \lambda_a \leq 1$ are the Schmidt coefficients.

Notice how the Schmidt decomposition is analogous to the SVD one. This fact is the backbone of tensor network states as an approximate description of entangled quantum many-body systems. In practice, a selected number of sites, or sites and virtual links, are grouped into a bipartite system, whose singular values are discarded with basis on e. g. a defined threshold, thereby (momentarily) reducing the norm of the resulting system, and the bond dimension of a virtual link.

3.2.2 Matrix Product States

With the basic operations we described for tensor networks, we can cast a general tensor state into a bipartite system (Fig. 3.9). In particular, fusing links together until only two links remain, we effectively have a map on $\mathcal{H}_1 \otimes \mathcal{H}_{2|N} \cong \mathcal{H}_1 \otimes (\mathcal{H}_2 \otimes \cdots \otimes \mathcal{H}_N)$. We then decompose the bipartite system into singular values.

Contracting s to the right, we end up with two tensors, one with $d \times d$ components, and the other with $d \times d^{N-1}$. We can decompose the second tensor, now fusing its second link with the first, and the links into a separate link. After contraction of the singular values matrix to the right, we are left with two new tensors, Fig. 3.15. We can iterate this process until we have run out of sites. This decomposition is foundational to the hypothesis of tensor network solutions. Contraction of all these tensors together recovers the original state.

A Matrix Product State (MPS) is a tensor network ansatz which is structurally identical to the output decomposition described above. In practice, however, the initial, full state is only accessible for a small number of states. Instead, the MPS configuration is initialized with a maximum *bond dimension*, $\chi_{\max}$ the dimension of the virtual links between the tensors in the decomposition, Fig. 3.16.

3.2.3 Tree Tensor Networks

MPS are not the only loop-free TN decompositions. Consider again the N-sites state general tensor. For simplicity, assume the number of sites N to be a power of

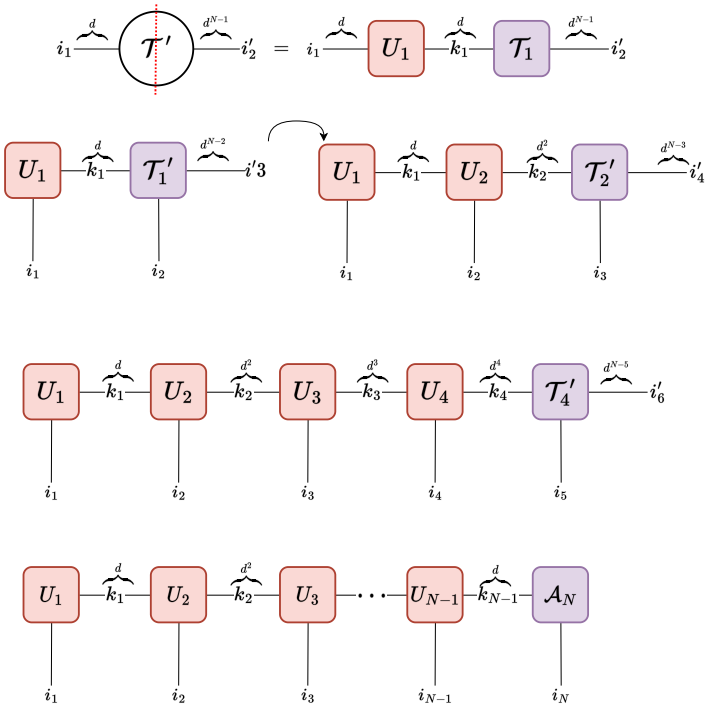

Figure 3.15: A derivation for illustrative purposes of the matrix product state decomposition starting from a full state.

2, $N = 2^m$. Split the tensor in half by fusing $N/2$ links to the left and the other $N/2$ to the right. After SVD and contraction of the singular values to either side, the outcome is two tensors with 2^{m-1} free links each and a virtual link that connects them. Now, fuse half of the free links at one tensor to the virtual link, and fuse its remaining links together to break it into two tensors with four free links each. One of the tensors has two virtual links. Fuse them together, and the physical links into a separate one, then decompose this tensor to generate another tensor with only one virtual link. Iteration of the last two steps eventually leads to a binary tree structure.

The decomposition structure depicted in Fig. 3.17 serves as reference to motivate the ansätze known as (binary) *Tree Tensor Networks* (TTNs), which are generated with a controlled bond dimension.

Given the path sizes between nodes, TTNs can encode a power-law decaying correlation [44], as opposed to MPS, which encode exponentially decaying correlations [21].

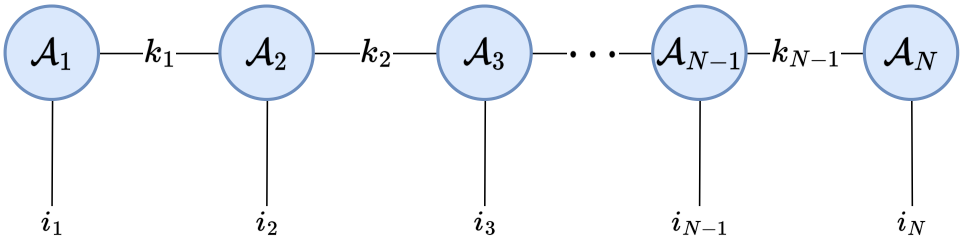

Figure 3.16: Generic MPS decomposition. The virtual links k_n have a dimension at most $\chi_{\max}$.

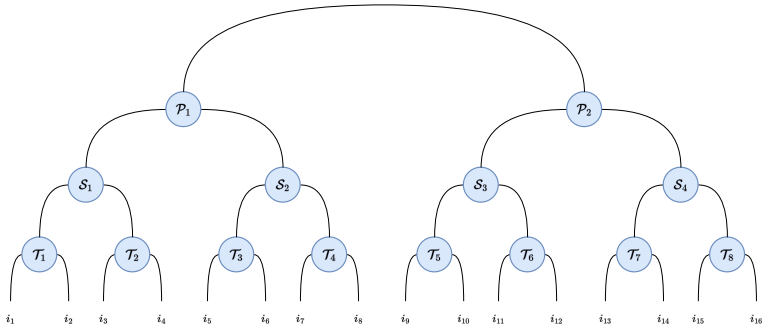

Figure 3.17: A binary TTN for 16 sites.

3.2.4 Enconding meaningful states

Having established the TTN and MPS decompositions, we naturally wish to address how meaningful states are encoded in these TNs. We can privilege a specific configuration of the basis we are using and find an almost trivial description.

$$|\psi\rangle = |0, 0, \ldots, 0, 0\rangle \tag{3.22}$$

$$\implies \mathcal{T}_{i_1,i_2,\ldots,i_N} = \begin{cases} 1, & \text{if } i_k = 0 \ \forall k \\ 0, & \text{any other case.} \end{cases} \tag{3.23}$$

In an MPS description, this is equivalent to

$$\mathcal{T}_{i_1,i_2,\ldots,i_N} = \sum_{j_1,j_2,\ldots,j_{N-1}} A^{[1]\ j_1}_{i_1} A^{[2]\ j_2}_{j_1,i_2} \ldots A^{[N]}_{j_{N-1},i_N} \tag{3.24}$$

$$\mathcal{T}_{0,0,\ldots,0} = \sum_{j_1,j_2,\ldots,j_{N-1}} A^{[1]\ j_1}_{0} A^{[2]\ j_2}_{j_1,0} \ldots A^{[N]}_{j_{N-1},0} = 1. \tag{3.25}$$

We set each $A^{[k]\ j_k}_{j_{k-1},i_k} = 0$ whenever $i_k \neq 0$. Then, for simplicity, we impose a similar condition over the j_k, so that

$$A^{[k]\ j_k}_{j_{k-1},i_k} = \begin{cases} 1, & \text{if } j_k = j_{k-1} = i_k = 0 \\ 0, & \text{otherwise.} \end{cases} \tag{3.26}$$

And, therefore,

$$\sum_{j_1,j_2,\ldots,j_{N-1}} A^{[1]\ j_1}_{0} A^{[2]\ j_2}_{j_1,0} \ldots A^{[N]}_{j_{N-1},0} = A^{[1]\ 0}_{0} A^{[2]\ 0}_{0,0} \ldots A^{[N]}_{0,0} = 1, \tag{3.27}$$

we effectively encode $|\psi\rangle = |0, 0, \ldots, 0\rangle$ in our MPS.

In this fashion, we could find a description for generic elements of the Fock space basis, which we can then isometrize for efficient manipulations. We could, for example, model a quantum circuit with gates that operate over a small number of qubits at a time [45].

Nevertheless, not every state can be described accurately by means of the MPS and TTN ansätze, or any tensor network decomposition for that matter. The state fidelity is constrained by the bond dimension, resulting in only approximate descriptions of states.

Eigenstates of Hamiltonian operators cannot always be reached efficiently from a Fock basis state. In practice, informed or random initializations for TNs are often useful to reach meaningful states through variational approaches. An outstanding algorithm for ground-state search is that of imaginary time evolution, the implementation of which for MPS we discuss in section 3.3.

3.3 Time-evolving block decimation

Time-Evolving Block Decimation (TEBD), as the name implies, can be used to study the time evolution of a quantum state under a given Hamiltonian. We can

split a nearest-neighbour Hamiltonian $\hat{H} = \sum_l \hat{h}_{l,l+1}$ into even and odd site terms:

$$\hat{H} = \sum_{l \text{ odd}} \hat{h}_{l,l+1} + \sum_{l \text{ even}} \hat{h}_{l,l+1} \tag{3.28}$$

$$= \sum_{l \text{ odd}} \hat{F}_l + \sum_{l \text{ even}} \hat{G}_l = \hat{F} + \hat{G} \tag{3.29}$$

$$[F_l, F_{l'}] = [G_l, G_{l'}] = 0 \ \forall l, l' \tag{3.30}$$

where the last equation follows from the definition of the tensor product.

Given the commutation of the $\hat{F}_l$ terms, we have

$$e^{-\mathrm{i}t\hat{F}} = e^{-it\sum \hat{F}_l} = \prod_{l \text{ odd}} e^{-it\hat{F}_l}. \tag{3.31}$$

Likewise,

$$e^{-it\hat{G}} = \prod_{l \text{ even}} e^{-it\hat{G}_l}. \tag{3.32}$$

The operators $\hat{F}_l$ and $\hat{G}_l$ however, do not commute, and thus the application of $e^{it(\hat{F}+\hat{G})}$ must be approximated. Taking finitely small timesteps δt, an approximation, e. g. Suzuki-Trotter to the second order, enables the layered application of these unitary transformations.

When the Hamiltonian is time-independent, and in between energy measurements, we can spare a third part of steps by applying together adjacent $e^{-i\frac{\delta t}{2}\hat{G}}$.

$$e^{-i\delta t\hat{H}} |\psi\rangle \approx e^{-i\frac{\delta t}{2}\hat{G}} e^{-i\delta t\hat{F}} e^{-i\frac{\delta t}{2}\hat{G}} |\psi\rangle \tag{3.33}$$

$$e^{-i\delta t\hat{H}} e^{-i\delta t\hat{H}} |\psi\rangle = e^{-i\frac{\delta t}{2}\hat{G}} e^{-i\delta t\hat{F}} e^{-i\delta t\hat{G}} e^{-i\delta t\hat{F}} e^{-i\frac{\delta t}{2}\hat{G}} |\psi\rangle . \tag{3.34}$$

After each contraction, we recover the two site matrices by means of SVD. Notice that this decomposition would increase the dimension of the intermediate link, requiring truncation to keep the bond dimension, and subsequent renormalization of the resulting state.

As usual, considering imaginary time $\tau = \mathrm{i}t$, we obtain a ground state search procedure, assuming $|\psi\rangle$ to be initialized in a state which has a non-zero projection to the ground state.

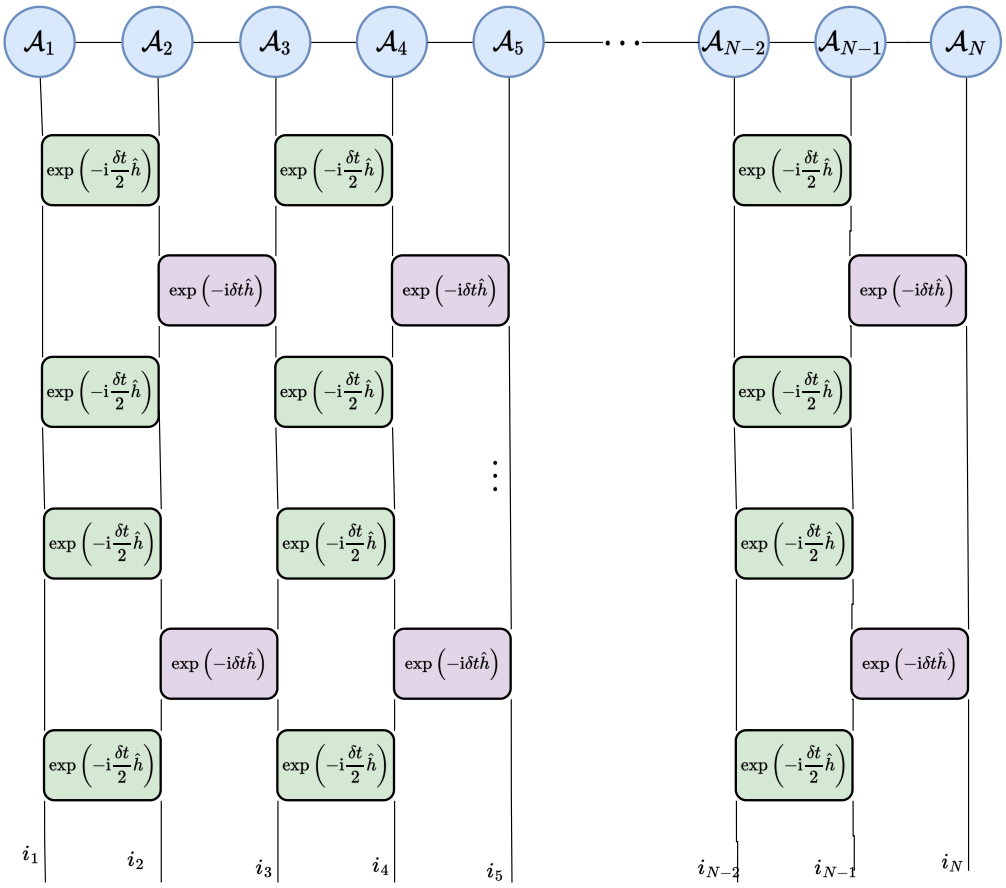

Figure 3.18: Application of two full time-steps of second-order evolution operator in TEBD. We depict odd (even) layers in green (purple).

4

Symmetric Tensor Networks

In chapter 3, we established the structure and methods for general tensor networks. We now take a step back to exploit our knowledge of representation theory from chapter 2, which we saw to be deeply related to the notion of a tensor.

An in-depth review of Abelian symmetric TNs can be found in [29]. Here, we focus on non-Abelian symmetric TNs, which exploit the structure provided by Clebsch-Gordan coefficients. Thus, invariant tensors so defined, allow us to be mostly concerned with the manipulation of a reduced tensor, while much of the dimensionality is covered by the structure tensors.

The present chapter mirrors the discussion of chapter 3, promoting tensors to symmetry-preserving objects. In section 4.1 we review how symmetric tree tensors emerge from the Clebsch-Gordan decomposition and Schur's lemma, and discuss how tree-tensors are building blocks for higher-rank tensors. Section 4.2 highlights the key differences between non-symmetric tensors and symmetric ones in low-level operations and decompositions. In section 4.3, we show MPS and TTN to be intuitive constructs with tree-tensor building blocks.

The discussions in this chapter largely borrow from [38], with methods from [39], as combined in [42]. Ref. [43] offers a practical guide on the implementation of these methods with extensive examples with SU(2).

4.1 Symmetric nodes

We begin our discussion by studying the invariant-tensor structure for different numbers of links. Since we are interested in the group action properties, the direction of links in tensors becomes relevant, and thus we make it explicit.

A *rank*-0 *symmetric tensor* is defined as a scalar. As such, it is by construction trivially invariant.

A *rank*-1 *symmetric tensor* is a vector. For it to be SU(2)-invariant, it can only exist in the trivial representation, $\mathbb{0}$. A rank-1 invariant tensor's dimensionality arises solely from the link degeneracy, Fig. 4.1.

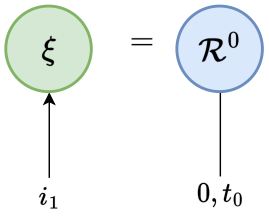

Figure 4.1: A one link invariant tensor can only have the trivial irrep.

Consider now an invariant rank-$(1,1)$ map $\hat{\Phi} : \mathbb{V}^n \to \mathbb{V}^n$. Of course, Schur's lemma follows, resulting in the decomposition

$$\hat{\Phi} = \bigoplus_J R^{[J]} \otimes \mathbb{I}_J, \tag{4.1}$$

where $\mathbb{I}_J$ is the identity map in the irrep J space, and $R^{[J]}$ is a diagonal matrix of dimension d_J, the degeneracy of irrep J. Eq. 4.1 describes the archetypical *rank*-2 *symmetric tensor*. We depict the decomposition of Φ by Fig. 4.2.

As we saw in 2.2.3, SU(2)-invariant operators are decomposed as

$$\mathcal{T} = \bigoplus_{[j,j',J]} \mathcal{R}^{F[J]}_{j,j'} \otimes \mathcal{C}^{F[J]}_{j,j'}. \tag{4.2}$$

This decomposition, illustrated in Fig. 4.3 is the *tree-tensor*, the invariant tensor structure that serves as a building block for tensor networks with global symmetries. It is the quintessential *rank*-3 *symmetric tensor*.

We may obtain a *rank*-4 *symmetric tensor* by means of contracting two tree-

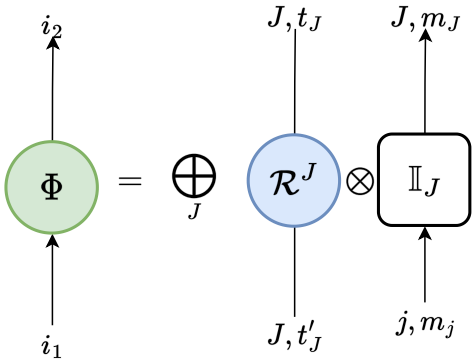

Figure 4.2: Schur decomposition of an invariant rank-2 tensor. Φ is a direct sum over the different sectors J, and decomposes into the reduced tensor $\mathcal{R}$

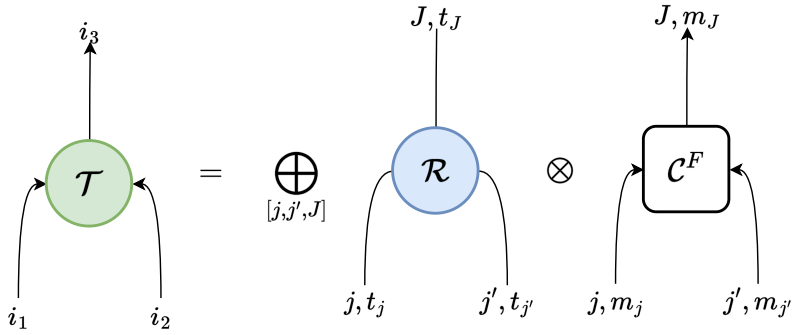

Figure 4.3: A tree-node is the building block decomposition of symmetric tensor networks.

tensors together, as in Fig. 4.4. This is not the only possible underlying structure. We can contract two tree tensors in a different way, and end up with the same irrep spaces for the links.

Tensors with an even higher number of links possess many more different tree-tensor decompositions for the structural part. The ansätze and algorithms we focus on, almost exclusively rely on tree-tensor decompositions. A comprehensive treatment of how different multi-link decompositions relate can be found in [38].

4.2 Manipulations of symmetric tensors

The fundamental operations of regular tensor networks have an analogue in non-Abelian symmetric TNs, with the distinction that they are performed sector-wise and over the reduced tensors. This execution enables an overall improved perfor-

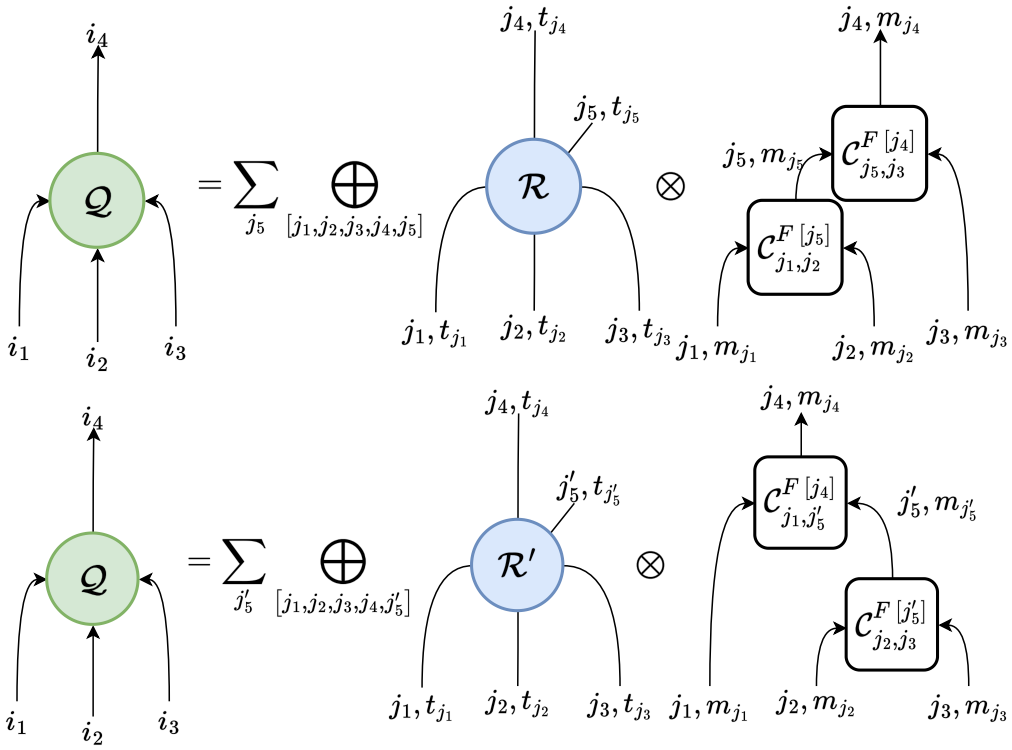

Figure 4.4: A rank-4 tensor requires an intermediate structural link which is not fixed by the tensor's external links.

mance in runtime, and renders the operations highly parallelisable.

4.2.1 Operations

Contraction

Contraction requires a duality relation between the spaces of target links. For regular TNs, this can be overlooked: the dimension suffices as a reference. However, in symmetric tensor networks, the directions, as well as the sectors contained, must be compatible, so that symmetry transformations are propagated properly.

The incoming link in $\hat{\Phi}$ from Eq. 4.1 and the outgoing link in $\mathcal{T}$ from Eq. 4.2 live in dual spaces to one another. This means we can in fact contract them together, as in Fig. 4.5. Furthermore, given the special structure of both tensors, the resulting tensor remains symmetric. This operation is performed sector-wise and its significant part is the contraction of links in the reduced tensors, since structure tensors

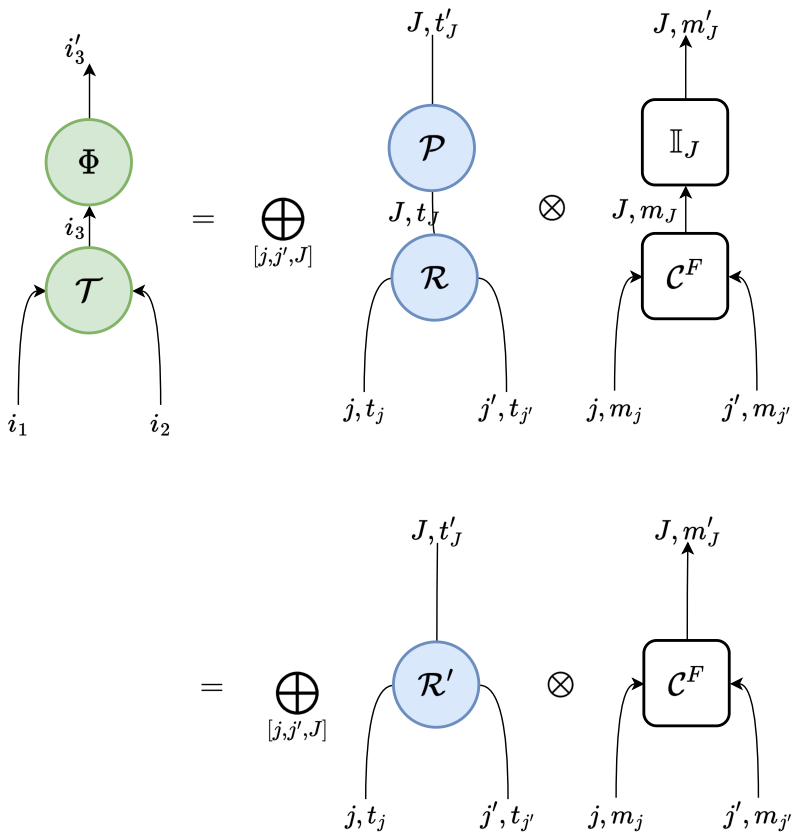

Figure 4.5: Another tree-tensor results from contracting a tree-tensor and an invariant rank-2 tensor.

are kept in the form discussed for rank-4 tensors and higher.

Fuse

The fusion of links in symmetric tensors differs from that of regular TNs. Given the transformations that act on the spaces associated to each tensor index, we must understand how the resulting link space decomposes to keep the symmetric tensor properties. In practice, this means precisely applying the tensor Υ^F from section 2.2.2,

$$\Upsilon^F = \bigoplus_{[j,j',J]} \mathcal{X}^{F[J]}_{j,j'} \otimes \mathcal{C}^{F[J]}_{j,j'}. \tag{4.3}$$

In the symmetric TN notation, Υ^F is depicted by Fig. 4.6. The target links are contracted together with their corresponding links in the fuse tensor. For the same

reason as Fig. 4.5, the resulting tensor remains invariant.

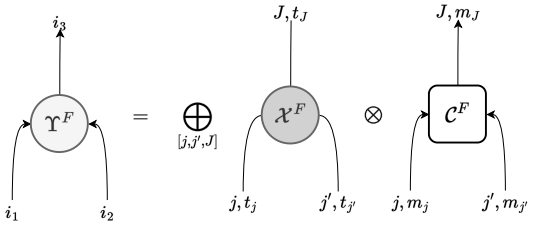

Figure 4.6: A fuse tensor is a tree-tensor that maps components between the bases of degenerate irreducible representations.

Split

Splitting a link in a symmetric node requires reviewing even further conditions. First of all, it must be possible to map the decomposition of link sectors to the tensor product of two sets of sectors. For SU(2) this is always possible. For any irrep J,

$$\mathbb{0} \otimes \mathrm{J} = \mathrm{J}, \tag{4.4}$$

and thus a link i_A can always be split into two links i_B and i_C, one of them identical sector-wise to i_A, and the other with the trivial sector. This is particularly useful to introduce a *dummy link*.

In contrast, target sectors must be specified when we desire splits other than the trivial one; it must of course be possible to find a bi-partite factorization that contains such sectors. In practice, we usually split links that are originally the outcome of a known fusion Υ^F. We can therefore conjugate this fuse node to split the target link.

Permutation

The permutation of links is performed in much the same fashion as with regular TNs, except coupling sector-wise, so that

$$\mathcal{T}^{\sigma}{}_{p(a,b,c)} = \bigoplus_{[p(a,b,c)]} \mathcal{R}^{\sigma}{}_{p(j_a,t_{j_a};j_b,t_{j_b};j_c,t_{j_c})} \otimes \mathcal{C}^{\sigma}{}_{p(j_a,m_{j_a};j_b,m_{j_b};j_c,m_{j_c})}. \tag{4.5}$$

4.2.2 Decompositions

The decompositions that were valid for regular TNs remain possible, with the advantage that they are now performed on the reduced tensors along the coupling sectors.

Singular Value Decomposition

Starting from a rank-2 tensor,

$$\mathcal{M} = \bigoplus_J \mathcal{R}_J \otimes \mathbb{I}_J \tag{4.6}$$

$$= \bigoplus_J \left(U^{[J]} \otimes \mathbb{I}_J \right) \left(\lambda^{[J]} \otimes \mathbb{I}_J \right) \left(V^{[J]} \otimes \mathbb{I}_J \right), \tag{4.7}$$

the decomposition is mapped to the reduced tensor, and all the properties from regular SVD follow block-wise.

4.3 Special Symmetric tensor network states

We can iterate this procedure with as many sites are required, and generate a non-Abelian symmetric MPS or bTTN. Contracting the top layer links together while preserving the symmetry requires them to be dual. For SU(2), such fusion is straightforward, as its irreps are self-dual. We can "revert" an outcoming link by focusing on the Clebsch-Gordan coefficients that map the irrep fusion to the singlet. Likewise, incoming links of the same irrep can be obtained by splitting the singlet irrep:

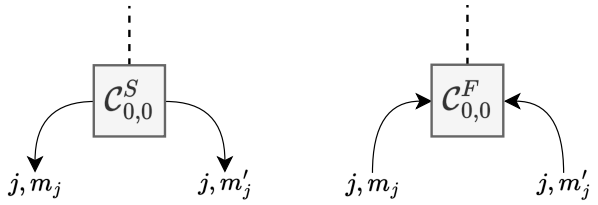

Figure 4.7: Reversal nodes map a link to its dual space pivoting on the trivial irrep.

Invariance for rank-one tensors is only possible when the structure tensor is the trivial one, that is, the symmetry sector is the trivial irrep. To study a symmetry

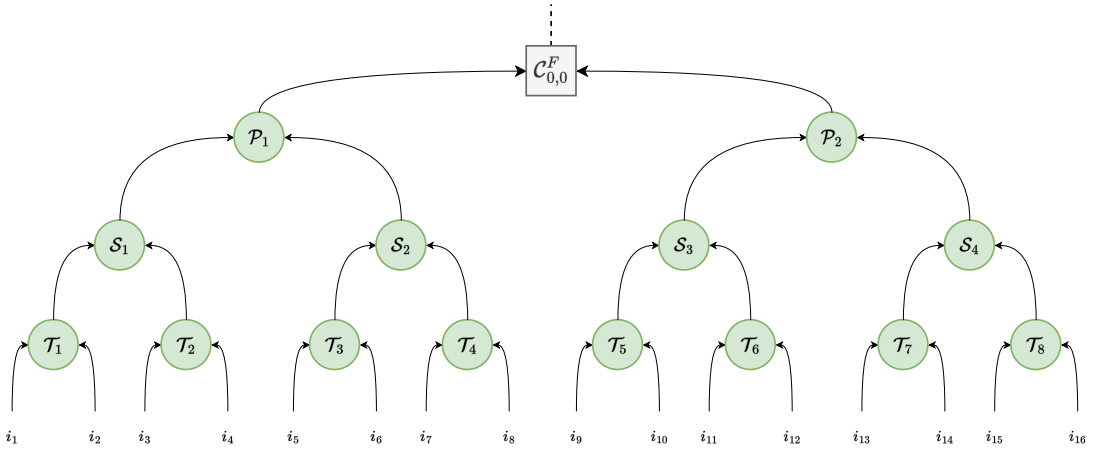

Figure 4.8: A binary TTN is formed by many tree-tensors. A virtual reversal node connects the top layer of a symmetric bTTN. Contraction of this node to either side sets the direction of the renormalization.

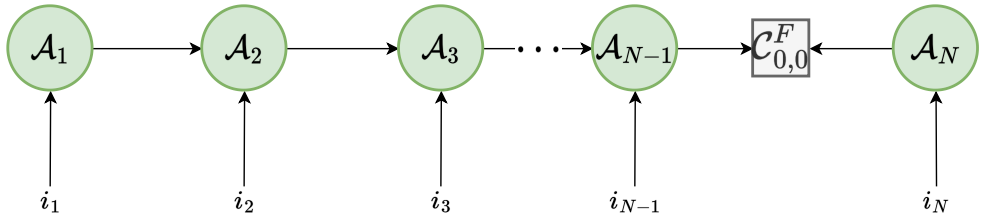

Figure 4.9: An MPS is formed by many tree-tensors. A virtual reversal node connects the last two tensors of a symmetric MPS.

sector other than $\mathbb{0}$, we attach an outgoing *selector link* which contains the target sector. Conceptually, this is equivalent to imposing a structure compatible with Schur's lemma for the tensor at hand, so that the symmetry is no longer limited to trivial transformations. The resulting tensor is therefore covariant.

4.4 Application: Bilinear biquadratic spin-$\mathbb{1}$ chain

Consider the bilinear biquadratic model [33, 34], a two-site interaction given by

$$\hat{h}_{i,i+1} = \cos\theta\,(\mathrm{S}_i \cdot \mathrm{S}_{i+1}) + \sin\theta\,(\mathrm{S}_i \cdot \mathrm{S}_{i+1})^2 \tag{4.8}$$

where S_i is the spin operators vector, thus $\mathrm{S}_i \cdot \mathrm{S}_{i+1} = S^x_i \otimes S^x_{i+1} + S^y_i \otimes S^y_{i+1} + S^z_i \otimes S^z_{i+1}$. Notice that since for irrep $\frac{\mathbb{1}}{2}$, $(S^a)^2 = \mathbb{I}_{\frac{\mathbb{1}}{2}}$, $\forall a$, this model necessarily operates over sites of higher dimensional irreps. This model exhibits pointwise as well as global SU(2)

symmetry, and is therefore a candidate for non-Abelian symmetric TN modelling.

4.4.1 Sectors of the nearest-neighbour Hamiltonian

Starting from the spin-$\mathbb{1}$ representation of the SU(2) generators,

$$S^x_{\mathbb{1}} = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \qquad S^y_{\mathbb{1}} = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & -\mathrm{i} & 0 \\ \mathrm{i} & 0 & -\mathrm{i} \\ 0 & \mathrm{i} & 0 \end{pmatrix}, \qquad S^z_{\mathbb{1}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}, \tag{4.9}$$

the two-site interaction of the BLBQ model requires two terms,

$$\mathrm{S}_{\mathbb{1}} \cdot \mathrm{S}_{\mathbb{1}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{4.10}$$

and the same term squared,

$$(\mathrm{S}_{\mathbb{1}} \cdot \mathrm{S}_{\mathbb{1}})^2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 2 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.11}$$

Each term exhibits SU(2) invariance, and therefore, linear combinations of them are also SU(2)-invariant. We can use the Casimir operator basis to diagonalise both

terms,

$$\mathrm{S}_{\mathbb{1}} \cdot \mathrm{S}_{\mathbb{1}} = (-2 \otimes \mathbb{I}_0) \oplus (-1 \otimes \mathbb{I}_1) \oplus (1 \otimes \mathbb{I}_2) \tag{4.12}$$

$$(\mathrm{S}_{\mathbb{1}} \cdot \mathrm{S}_{\mathbb{1}})^2 = (4 \otimes \mathbb{I}_0) \oplus (1 \otimes \mathbb{I}_1) \oplus (1 \otimes \mathbb{I}_2), \tag{4.13}$$

broken into symmetry sectors as expected from Schur's lemma.

Thus, the BLBQ two-site interaction is a non-Abelian symmetric tensor operator of the form

$$\hat{h}(\theta) = \cos\theta \left(\mathrm{S}'_{\mathbb{1}} \cdot \mathrm{S}'_{\mathbb{1}}\right) + \sin\theta \left(\mathrm{S}'_{\mathbb{1}} \cdot \mathrm{S}'_{\mathbb{1}}\right)^2 \tag{4.14}$$

$$= (4\sin\theta - 2\cos\theta) \otimes \mathbb{I}_{\mathbb{V}_0} \oplus \tag{4.15}$$

$$\oplus\, (\sin\theta - \cos\theta) \otimes \mathbb{I}_{\mathbb{V}_1} \oplus \tag{4.16}$$

$$\oplus\, (\sin\theta + \cos\theta) \otimes \mathbb{I}_{\mathbb{V}_2}. \tag{4.17}$$

With this expression for the nearest-neighbours interaction, we are now ready to tackle the spin-$\mathbb{1}$ BLBQ model with a symmetric tensor network.

4.4.2 TEBD ground states

For a small number of sites, we take as reference the spin-0 sector lowest energy, which is not necessarily the ground state. The exact diagonalization energy curve for the full range of θ is depicted in 4.10. Already at $N = 6$ sites we notice pronounced phase transitions in the model [34]. We now proceed to measure the energy relative error,

$$\mathrm{err}(\theta_i) = \left| \frac{E_{\mathrm{ED}}(\theta_i) - E_{\mathrm{TEBD}}(\theta_i)}{E_{\mathrm{ED}}(\theta_i)} \right|, \tag{4.18}$$

where E_{ED} and E_{TEBD} are the sector minimum energies as predicted by ED and TEBD, respectively. Notice how those points harder to simulate are close to phase transitions, Fig. 4.11.

In contrast, for the highest possible sector in each site, we encounter well-converged results, where most of the error values are numerical zeros. For non-degenerate sectors in the physical links, as is the case in our model, the highest possible spin sector is not degenerate, and its reduced tensor amounts to a single entry. To illustrate

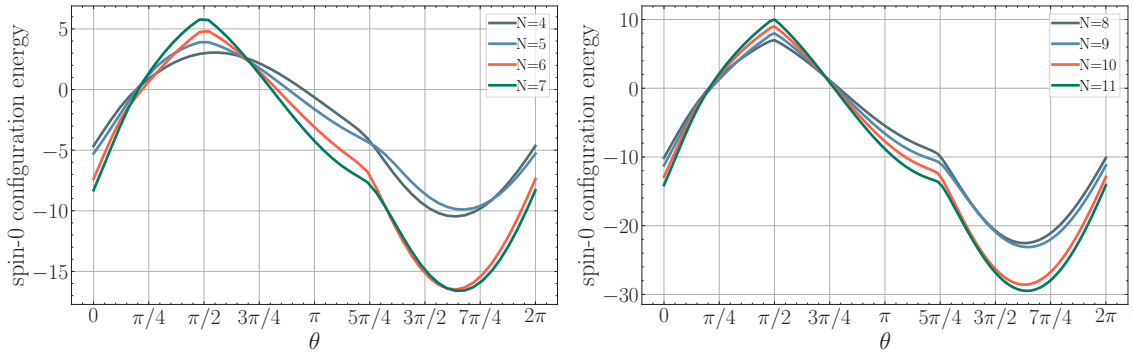

Figure 4.10: Spin-0 minimum energy in BLBQ model for different lattice sizes N. In the exact diagonalization for a small number of sites, N (left), different phase transitions are already present. In the total spin 0 ground state TEBD estimation for larger numbers of sites (right), the transitions become more pronounced.

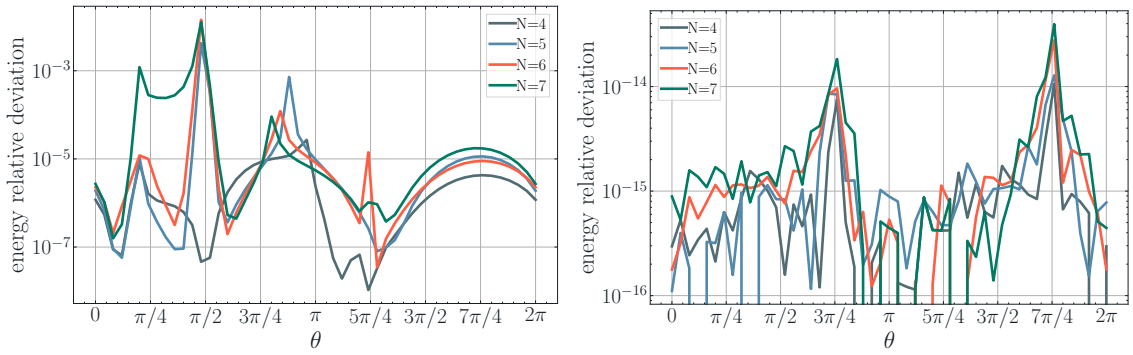

Figure 4.11: Error in TEBD ground energy for total spin 0 configuration in the BLBQ model (left). Phase transitions are harder to predict accurately. To the right, error in energy prediction for the highest possible total spin for a small number of sites. The values are numerical zeros. The highest possible total spin configuration is usually non-degenerate, making it an easy simulation target.

this, consider an N-sites, spin-$\mathbb{1}$ MPS. In the virtual links, always keep the highest possible sector. This is non-degenerate at each site n,

$$\mathcal{R}^{[v_n,t_{v_n}]}_{v_{n-1},t_{v_{n-1}};j_n,t_{j_n}}, \tag{4.19}$$

$$|t_{v_{n-1}}| = |t_{j_n}| = |t_{v_n}| = 1, \tag{4.20}$$

where v_l is the highest sector in the outcoming virtual link of site l's matrix. This chain eventually leads to a selector link which cannot have more than one possible t_s for the maximum s.

We take as reference the highest error θ values from Fig. 4.11 to study the conver-

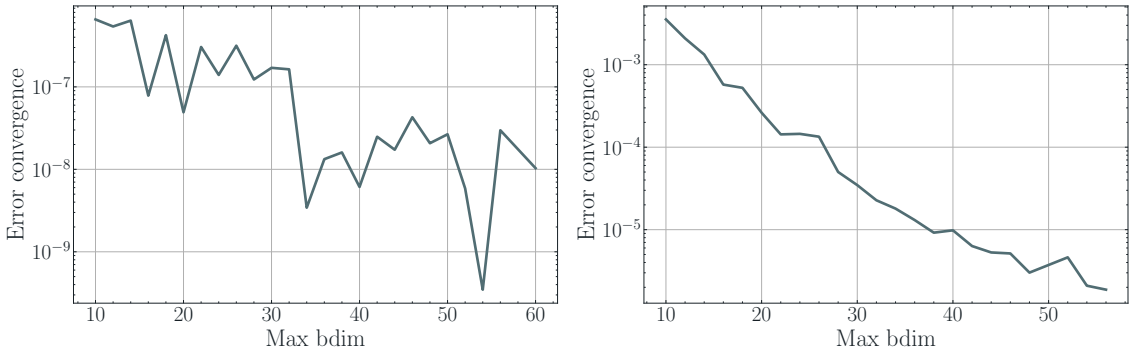

Figure 4.12: Measured energy convergence study for critical point $\theta = \pi/2$ (left) and $\theta = \pi/4$ (right).

gence in a 50-site MPS for the BLBQ model, Fig. 4.12. At each step t, we increase the bond dimension ξ_t and compare the energy predictions as follows,

$$\mathrm{conv}_{\chi_t} = \left| \frac{E^0_{\chi_t} - E^0_{\chi_{t-1}}}{E^0_{\chi_{t-1}}} \right|. \tag{4.21}$$

We observe the transition between the ferromagnetic phase and the critical phase, $\theta = \pi/2$ to remain resource-intensive.

5

Conclusions and Outlook

Our contribution is a big step towards the use of non-Abelian tensors by client modules such as the Matrix Product Operator. Many internal dependencies of the NASTY library were adapted, so as to comply with an easier-to-maintain, deploy and upgrade module. The global types, general tensors, and error-handling dependencies are now fulfilled by equivalent independent submodules. The advantage of this update lies in the fact that such submodules are consumed by applications similar to NASTY, and become more reliable development-wise: upgrades and bug-fixes at the lower level are easier to propagate. We developed robust and versatile interfaces aimed towards non-Abelian TTN simulations.

We showed the computational advantages of tensor networks (TNs), and their relevance to quantum many-body (QMB) problems. We discussed the conceptual challenges towards building a symmetric TN, and established a collection of unit and integration tests in the pursuit of reliable TN software development. We expanded the functionality of the Quantum Green TEA library, enabling the explicit encoding of non-Abelian symmetries for their subsequent exploitation in the simulation of QMB systems. Given the prevalence of such symmetries in systems of interest for the development of quantum technologies, and, even more so, models of theoretical relevance such as the bi-linear bi-quadratic (BLBQ) spin-1 model, we expect this outcome to be of considerable use. Exploiting Schur's lemma and, more generally, the Wigner-Eckart theorem, the construct of tree-tensor blocks for symmetric

TNs proved convenient for manipulating binary tree tensor network (TTN) ansätze. This compatibility has- resulted in an almost seamless transition from non-Abelian symmetric matrix product states (MPS) to TTNs of the same nature.

We demonstrated the accuracy and effectiveness of the TEBD algorithm by comparing it with the results of exact diagonalization (ED) for the Heisenberg nearest neighbours model, as well as the BLBQ model for bosonic interactions. Given the periodic nature of the θ parameter in the BLBQ model, we were able to establish a maximum deviation between the ED prediction and that of TEBD, with a 10^{-2} relative error for the critical point $\theta = \pi/4$. With this reference, we scaled the model up to 50 sites and tested the estimation convergence with an increasing bond dimension.

A first potential improvement for the future is the encoding of multiple non-Abelian symmetries, as much of the structure is already compatible with this. Although a separate Quantum TEA implementation exists for Abelian symmetries, a multiple symmetries version of NASTY may be easily adapted to deal with combined symmetries, e. g. U(1)×SU(2), where at least one of the symmetries is Abelian and at least one of them is non-Abelian.

Extending the applications of the current implementation, we could perform efficient measurement of symmetric correlators by encoding them as rank-4 tensors and performing an SVD decomposition, avoiding the computational costs of converting a non-Abelian symmetric tensor network state into a dense tensor network.

References

[1] I. M. Georgescu, S. Ashhab, and F. Nori, "Quantum simulation," *Rev. Mod. Phys.*, vol. 86, pp. 153–185, Mar 2014.

[2] "Quantum computational networks," *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, vol. 425, pp. 73–90, Sept. 1989.

[3] J.-S. Lee, Y. Chung, J. Kim, and S. Lee, "A Practical Method of Constructing Quantum Combinational Logic Circuits," 1999.

[4] R. Jozsa and N. Linden, "On the role of entanglement in quantum-computational speed-up," *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 459, pp. 2011–2032, Aug. 2003.

[5] R. P. Feynman, "Quantum mechanical computers," *Foundations of Physics*, vol. 16, pp. 507–531, Jun 1986.

[6] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484–1509, 1997.

[7] B. P. Lanyon, T. J. Weinhold, N. K. Langford, M. Barbieri, D. F. V. James, A. Gilchrist, and A. G. White, "Experimental Demonstration of a Compiled Version of Shor's Algorithm with Quantum Entanglement," *Phys. Rev. Lett.*, vol. 99, p. 250505, Dec 2007.

[8] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, (New York, NY, USA), p. 212–219, Association for Computing Machinery, 1996.

[9] E. Farhi, J. Goldstone, and S. Gutmann, "A Quantum Approximate Optimization Algorithm," 2014.

[10] Z. Wang, S. Hadfield, Z. Jiang, and E. G. Rieffel, "Quantum approximate optimization algorithm for maxcut: A fermionic view," *Phys. Rev. A*, vol. 97, p. 022304, Feb 2018.

[11] J. Preskill, "Quantum Computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018.

[12] W. Kohn, "Density functional theory," *Introductory Quantum Mechanics with MATLAB: For Atoms, Molecules, Clusters, and Nanocrystals*, 2019.

[13] W. M. C. Foulkes, L. Mitas, R. J. Needs, and G. Rajagopal, "Quantum Monte Carlo simulations of solids," *Rev. Mod. Phys.*, vol. 73, pp. 33–83, Jan 2001.

[14] E. Y. Loh, J. E. Gubernatis, R. T. Scalettar, S. R. White, D. J. Scalapino, and R. L. Sugar, "Sign problem in the numerical simulation of many-electron systems," *Phys. Rev. B*, vol. 41, pp. 9301–9307, May 1990.

[15] E. Biham, G. Brassard, D. Kenigsberg, and T. Mor, "Quantum computing without entanglement," *Theoretical Computer Science*, vol. 320, pp. 15–33, June 2004.

[16] J. J. Sakurai and J. Napolitano, *Modern Quantum Mechanics*. Cambridge University Press, 2 ed., 2017.

[17] J. I. Cirac, D. Pérez-García, N. Schuch, and F. Verstraete, "Matrix product states and projected entangled pair states: Concepts, symmetries, theorems," *Rev. Mod. Phys.*, vol. 93, p. 045003, Dec 2021.

[18] M. C. Bañuls, "Tensor Network Algorithms: a Route Map," 2022.

[19] K. G. Wilson, "The renormalization group: Critical phenomena and the Kondo problem," *Rev. Mod. Phys.*, vol. 47, pp. 773–840, Oct 1975.

[20] S. R. White, "Density matrix formulation for quantum renormalization groups," *Phys. Rev. Lett.*, vol. 69, pp. 2863–2866, Nov 1992.

[21] G. Vidal, J. I. Latorre, E. Rico, and A. Kitaev, "Entanglement in Quantum Critical Phenomena," *Phys. Rev. Lett.*, vol. 90, p. 227902, Jun 2003.

[22] U. Schollwöck, "The density-matrix renormalization group," *Rev. Mod. Phys.*, vol. 77, pp. 259–315, Apr 2005.

[23] F. Verstraete, V. Murg, and J. Cirac, "Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems," *Advances in Physics*, vol. 57, no. 2, pp. 143–224, 2008.

[24] X. Yuan, J. Sun, J. Liu, Q. Zhao, and Y. Zhou, "Quantum Simulation with Hybrid Tensor Networks," *Phys. Rev. Lett.*, vol. 127, p. 040501, Jul 2021.

[25] S. Montangero, E. Rico, and P. Silvi, "Loop-free tensor networks for high-energy physics," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 380, dec 2021.

[26] Z.-Y. Han, J. Wang, H. Fan, L. Wang, and P. Zhang, "Unsupervised Generative Modeling Using Matrix Product States," *Phys. Rev. X*, vol. 8, p. 031012, Jul 2018.

[27] J. C. Bridgeman and C. T. Chubb, "Hand-waving and interpretive dance: an introductory course on tensor networks," *Journal of Physics A: Mathematical and Theoretical*, vol. 50, p. 223001, may 2017.

[28] G. Vidal, "Efficient simulation of one-dimensional quantum many-body systems," *Phys. Rev. Lett.*, vol. 93, p. 040502, Jul 2004.

[29] P. Silvi, F. Tschirsich, M. Gerster, J. Jünemann, D. Jaschke, M. Rizzi, and S. Montangero, "The Tensor Networks Anthology: Simulation techniques for many-body quantum lattice systems," *SciPost Phys. Lect. Notes*, p. 8, 2019.

[30] J. Gray and S. Kourtis, "Hyper-optimized tensor network contraction," *Quantum*, vol. 5, p. 410, Mar. 2021.

[31] I. P. McCulloch, "From density-matrix renormalization group to matrix product states," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2007, p. P10014, oct 2007.

[32] U. Schollwöck, T. Jolicœur, and T. Garel, "Onset of incommensurability at the valence-bond-solid point in the S=1 quantum spin chain," *Phys. Rev. B*, vol. 53, pp. 3304–3311, Feb 1996.

[33] G. De Chiara, M. Lewenstein, and A. Sanpera, "Bilinear-biquadratic spin-1 chain undergoing quadratic Zeeman effect," *Phys. Rev. B*, vol. 84, p. 054451, Aug 2011.

[34] M. V. Rakov and M. Weyrauch, "Bilinear-biquadratic spin-1 model in the Haldane and dimerized phases," *Phys. Rev. B*, vol. 105, p. 024424, Jan 2022.

[35] B. Li, Z.-H. Yu, and S.-M. Fei, "Geometry of Quantum Computation with Qutrits," *Scientific Reports*, vol. 3, Sept. 2013.

[36] M. S. Blok, V. V. Ramasesh, T. Schuster, K. O'Brien, J. M. Kreikebaum, D. Dahlen, A. Morvan, B. Yoshida, N. Y. Yao, and I. Siddiqi, "Quantum Information Scrambling on a Superconducting Qutrit Processor," *Phys. Rev. X*, vol. 11, p. 021010, Apr 2021.

[37] N. D. Mermin and H. Wagner, "Absence of Ferromagnetism or Antiferromagnetism in One- or Two-Dimensional Isotropic Heisenberg Models," *Phys. Rev. Lett.*, vol. 17, pp. 1133–1136, Nov 1966.

[38] S. Singh and G. Vidal, "Tensor network states and algorithms in the presence of a global SU(2) symmetry," *Phys. Rev. B*, vol. 86, p. 195114, Nov 2012.

[39] A. Weichselbaum, "Non-Abelian symmetries in tensor networks: A quantum symmetry space approach," *Annals of Physics*, vol. 327, pp. 2972–3047, dec 2012.

[40] D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, "Matrix Product State Representations," 2006.

[41] Y.-Y. Shi, L.-M. Duan, and G. Vidal, "Classical simulation of quantum many-body systems with a tree tensor network," *Phys. Rev. A*, vol. 74, p. 022320, Aug 2006.

[42] F. Frie, "Algorithms for non-Abelian Symmetries in Tensor Networks," Master's thesis, 2016.

[43] P. Schmoll, S. Singh, M. Rizzi, and R. Orús, "A programming guide for tensor networks with global SU(2) symmetry," *Annals of Physics*, vol. 419, p. 168232, 2020.

[44] P. Silvi, V. Giovannetti, S. Montangero, M. Rizzi, J. I. Cirac, and R. Fazio, "Homogeneous binary trees as ground states of quantum critical hamiltonians," *Phys. Rev. A*, vol. 81, p. 062335, Jun 2010.

[45] M. Ballarin, "Quantum Computer simulation via Tensor Networks," Master's thesis, 2019.