Tensor Networks and Geometry for the Modelling of Disordered Quantum Many-Body Systems

by

Andrew M Goldsborough

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy

Physics

July 2015

Contents

List of Figures

Acknowledgments

Scientific research is never performed in a bubble, we rely so much on those around us for support, inspiration and distraction, without which getting to the point of submitting a PhD thesis would be impossible. I have a lot of people to thank and I have no doubt that I have missed a great many more.

First and foremost I would like to thank my supervisor Prof. Rudolf A. Römer for giving me the opportunity to live a life of science. He somehow managed to balance both guidance and freedom to pursue my own research interests, providing a wealth of knowledge on both physics and general life that will no doubt be influential in my future decisions. On the technical side I would like to thank the Engineering and Physical Science Research Council (EPSRC) for financial support (EP/J003476/1) and the MidPlus Regional HPC Centre (EP/K000128/1) for computing resources that were used to perform my research.

On a more personal level I would like to thank my fellow students in PS001 and members of the Warwick theory group notably: Matthew Bates, Dr. Jack Heal, Anja Humpert, Dr. Daniel Pearce, Dr. Sebastian Pinski and Alex Rautu to name but a few, for creating such an enjoyable and friendly working atmosphere, always being there to talk science or anything but. I would like to thank the coauthors of the graph theory work. The path length problem was originally something fun to try before going to the pub on a Friday afternoon, but turned out to be a fantastic learning experience. I would also like to thank the great many scientists from around the world whom I have met and become friends with at conferences and workshops. In many ways it is the people who make science such a unique and enjoyable enterprise.

Away from academia I would like to thank my group of school friends, collectively known as the *Knights of Ni* (KON) for all of the adventures over the years. My friends from the *Michael Stoker Building* (MSB) at Clare Hall, Cambridge for the good food and the good times. Dr. Mark Pinder for friendship and school bus discussions on science and philosophy that instilled a passion and thirst for knowledge that is yet to be diminished. My friend and flatmate Dr. David Dossett for friendship and guidance in some of the most stressful of times. My parents Alan and Janet, and brother Mark for unending support at every point in my life and making me the person that I am.

Finally, a special thanks go to my girlfriend Steph who has been so loving, kind and understanding, making me feel truly lucky to have her by my side.

Declarations

I declare that the content of this thesis is original work except where referenced within the text and has not been submitted as part of any other degree or qualification. Chapters 1, 2 and 4 provide a background to the field, the information for which was gathered from the texts cited throughout. Chapters 3, 5, 6 and 7 are respectively based on the following papers:

- A. M. Goldsborough and R. A. Römer. Using entanglement to discern phases in the disordered one-dimensional Bose-Hubbard model. *arXiv:1503.02973 [cond-mat.dis-nn]*, March 2015, (submitted to EPL).

- A. M. Goldsborough and R. A. Römer. Self-assembling tensor networks and holography in disordered spin chains. *Phys. Rev. B*, 89:214203, June 2014.

- A. M. Goldsborough, S. A. Rautu, and R. A. Römer. Leaf-to-leaf distances and their moments in finite and infinite m-ary tree graphs. *Phys. Rev. E*, 91:042133, April 2015.

- A. M. Goldsborough, J. M. Fellows, S. A. Rautu, M. Bates, G. Rowlands, and R. A. Römer. Leaf-to-leaf distances in ordered Catalan tree graphs. *arXiv:1502.07893 [math-ph]*, February 2015, (in preparation).

The entirety of the work was conducted under the supervision of Professor R. A. Römer and the content of the chapters highlighted above was performed in collaboration with the indicated authors. The thesis was typeset with $\LaTeX\,2_\varepsilon$ and the warwickthesis style using the Vim text editor. Graphs were drawn using XMGrace and Mathematica, diagrams with Inkscape.

Abstract

This thesis explores the use of tensor networks in the study of disordered quantum-many body systems and the connection between disorder in the Hamiltonian and tensor network geometry.

Tensor networks provide a powerful and elegant approach to quantum many-body simulation. The simplest example is the density matrix renormalisation group (DMRG), which is based on the variational update of a matrix product state (MPS). It has proved to be the most accurate approach for the numerical study of strongly correlated one dimensional systems. We use DMRG to study the one dimensional disordered Bose-Hubbard model at fillings $N/L = 1/2$, 1 and 2 and show that the whole phase diagram for each can be successfully obtained by analysing entanglement properties alone. We find that the average entanglement is insufficient to accurately locate all of the phases, however using the standard error on the mean we are able to construct a phase diagram that is consistent with previous studies.

It has recently been shown that there is a connection between the geometry of tensor networks and the entanglement and correlation properties that it can encode, which is a generalisation of the so called *area law* for entanglement entropy. This suggests that whilst gapped quantum systems can be accurately modeled using an MPS, a tensor network with a holographic geometry is natural to capture the logarithmic entanglement scaling and power law decaying correlation functions of critical systems. We create an algorithm for the disordered Heisenberg Hamiltonian that *self assembles* a tensor network based on the disorder in the couplings. The geometry created is that of a disordered tree tensor network (TTN) that when averaged has the holographic properties characteristic of critical systems.

We continue the analysis of holographic tensor network geometry by considering the average length of leaf-to-leaf paths in various tree graphs, which is related to two-point correlation functions in tensor networks. For regular, complete trees we analytically calculate the average path length and all statistical moments, and generalise it for any splitting number. We then turn to the Catalan trees, which is the set of unique binary trees with n vertices, as it has a similar geometry to the disordered TTNs. We calculate the average depth of a leaf and show that it is equal to the average path length. We compare these analytic results with the structures found in the TTN and randomly constructed trees to show that the renormalisation involved in the TTN algorithm is crucial in the selection of the tree structure.

Abbreviations

AdS - *Anti-de Sitter space*

AdS/CFT - *Anti-de Sitter/conformal field theory correspondence*

AFM - *Antiferromagnetic*

APBC - *Anti-periodic boundary condition*

CFT - *Conformal field theory*

DMRG - *Density matrix renormalisation group*

FM - *Ferromagnetic*

KT - *Kosterlitz-Thouless*

MBL - *Many-body localisation*

MDH - *Ma, Dasgupta and Hu*

MERA - *Multi-scale entanglement renormalisation ansatz*

MPO - *Matrix product operator*

MPS - *Matrix product state*

NRG - *Numerical renormalisation group*

OBC - *Open boundary condition*

PBC - *Periodic boundary condition*

PEPS - *Projected entangled pair state*

QMC - *Quantum Monte Carlo*

RG - *Renormalisation group*

SDRG - *Strong disorder renormalisation group*

SVD - *Singular value decomposition*

tSDRG - *Tree tensor network strong disorder renormalisation group*

TTN - *Tree tensor network*

Chapter 1

Introduction

At its most basic level a quantum many-body system can be defined by a Hilbert space and a Hamiltonian which can describe the evolution of the state. The dimension of the Hilbert space grows exponentially with the number of particles in the system. Calculations for systems containing a small handful of particles can be simple, but the exponential scaling means that exact calculations quickly become unfeasible. As a simple example take a spin-1/2 system, where each particle can take one of two states. Because of the tensor product form of the Hilbert space, the dimension scales as 2^n where n is the number of particles [1]. For one mole of such particles, the dimension and therefore the number of states is $\sim 2^{10^{23}}$. Thus the number of possible states in just one mole of these simplified particles dwarfs the total number of particles in the universe [2], which is estimated to be $\sim 10^{80}$. This highlights part of the problem in modelling quantum matter; any attempt to perform a calculation for a reasonably large number of particles fails spectacularly. The exponential growth of Hilbert space has led to some scholars questioning not just the impracticiality of such a large space, but even if such a construction is physical [3, 4].

As exact calculations for quantum many-body systems are not possible, it is necessary to use methods that reduce computational complexity whilst still achieving high accuracy. There have been several numerical algorithms that have enabled the simulation of materials that would have otherwise been impossible. For large scale electronic sturctures, like those found in condensed matter physics and quantum chemistry, dynamical mean field theory (DMFT) and density functional theory (DFT) have been very successful, but are limited to systems without significant entanglement. DMFT maps a many body problem onto a single particle, interacting with a field representing the average of all other particles [5]. DFT asserts that the properties of physical system can be determined from the electron density of the ground state, which can be determined from an effective non-interacting system [6]. Wilson's numerical renormalisation group (NRG) [7], as will be discussed in chapter 2, trucates the Hilbert space using the low energy eigenvectors of the Hamiltonian. It is used primarily on impurity models, and is not at all successful as a general approach to quantum many-body problems. Quantum Monte Carlo (QMC) is perhaps the most well known method, this maps a quantum problem in d spatial dimensions to that of a $d+1$ dimensional classical partition function [8]. One of the major issues with QMC is the *sign problem*, which prevents the method from being successful for many frustrated and fermionic systems. In these cases the mapping to the classical system introduces negative Boltzmann weights that make the statistical error quickly become larger than the property being calculated [8]. A new method created in 1992 by White, called the *density matrix renormalisation group*, revolutionised the simulation of one dimensional systems. It truncates the Hilbert space in accordance with the eigenstates corresponding to the largest χ eigenvalues of the ground state density matrix. Because of the use of the density matrix, rather than just the eigenvectors of the Hamiltonian as in NRG, DMRG encodes some of the entanglement of the ground state allowing accurate simulation of many quantum systems. Unlike QMC, DMRG does not suffer from the sign problem and is considered the most accurate numerical method for one dimensional systems [9].

Disordered quantum many-body systems are of great interest but are particularly problematic. It is natural to want to study systems with disorder as perfect clean materials are very rare in nature. Furthermore the introduction of disorder can completely change the observed phase of the material. The study of disordered systems became a major part of the study of the electronic properties of condensed matter with the analysis of localisation by Anderson [10]. For non-interacting electronic systems in a three dimensional disordered lattice it was found that a critical value of the on site disorder marked the transition between extended and localised states. The interplay between interactions and disorder is much less well understood and *many-body localisation* is currently a hot topic of research.

In this thesis we will analyse the theory of *tensor networks*, and apply them with a focus on disordered systems. In chapter 2 we introduce the basic concepts of a tensor network in the context of DMRG. We provide all information necessary to understand the fundamental operations that make up a tensor network algorithm and use them to construct the *matrix product state* (MPS), which is the simplest example of a tensor network. We show how to perform a variational update to the MPS, which is the core of all modern DMRG algorithms. Using this knowledge of MPS DMRG we analyse the disordered Bose-Hubbard model in chapter 3. As entanglement is at the heart of DMRG and is calculated at every point in the simulation of the system, we illustrate the efficacy of entanglement as a tool for deciphering the phase of the ground state wavefunction. The content of this chapter is based on ref. [11], which is currently under review. In chapter 4 we discuss more generally the theory of tensor networks, why they work for certain cases and why they don't for others and how tensor networks go beyond DMRG to simulate more complex systems. Chapter 5 creates a tensor network algorithm where the structure of the network is determined by the disorder in the Hamiltonian. In this way each disorder realisation has a custom *tree tensor network* (TTN) and when disorder averaged we show that the system has *holographic* properties that govern the entanglement and correlation scaling. The chapter is based on [12], which was published in Physical Review B. The final two chapters (6 and 7) analyse how the geometry of tree tensor networks affects the asymptotic form of the averaged two-point correlation function. Chapter 6 looks at a complete tree and finds the average leaf-to-leaf path length when the leaves of the tree are effective lattice points. This chapter is based on [13], which is currently under review. Chapter 7 extends the ideas of the previous chapter to sets of full binary trees that are more similar to those generated by tSDRG in chapter 5. This work is in [14], which is currently being prepared for submission.

Chapter 2

Matrix Product States and the Density Matrix Renormalisation Group

2.1 White's Density Matrix Renormalisation Group

The field of tensor networks has its roots in Wilson's NRG [7] and the DMRG method devised by White [15]. As mentioned in chapter 1, for an interacting one dimensional system of L sites, the Hilbert space of the system is the product of the Hilbert spaces of the sites

$$\mathcal{H}^L = \mathcal{H} \otimes \ldots \otimes \mathcal{H}. \tag{2.1.1}$$

If each site can take one of n states, the Hilbert space of the system scales as n^L. This exponential increase means that any computation quickly becomes intractable. NRG and DMRG are numerical methods that reduce the size of this Hilbert space to a point that computations are possible, but still retains the necessary information regarding the low energy eigenstates. As a motivation to the theory and application of tensor networks we will briefly sketch out these two algorithms.

2.1.1 Numerical Renormalisation Group

The NRG algorithm famously worked well for the Kondo impurity model, which maps a spherically symmetric Kondo model onto a lattice Hamiltonian with the impurity as the first site and the conduction electrons as the rest of the lattice [16]. NRG is a purely numerical method and works by tracing out the higher energy scales of the system but keeping the low energy states. The procedure is as follows

[16]:

1. Take L sites at the left hand side of the system to form a block, where L is small enough so that the Hamiltonian H_L can be exactly diagonalised.

2. Diagonalise H_L and keep only the m eigenvectors corresponding to the lowest eigenvalues as the higher eigenvectors do not contribute significantly to the ground state of the system.

3. Form a matrix O of the m eigenvectors

$$O = \begin{pmatrix} \vdots & \vdots & & \vdots \\ V_1 & V_2 & \dots & V_m \\ \vdots & \vdots & & \vdots \end{pmatrix} \tag{2.1.2}$$

   where V_1 is the ground state eigenvector and so on.

4. Use O to change the basis and truncate the size of the Hamiltonian

$$\bar{H}_L = O^\dagger H_L O \tag{2.1.3}$$

   and all other operators

$$\bar{A}_L = O^\dagger A_L O \tag{2.1.4}$$

   These will now be matrices of dimension $m \times m$.

5. Add a site onto the block to form H_{L+1} and repeat the procedure from step 2 until the full length of the chain is reached.

This procedure works well because each successive site is less well coupled to the impurity and thus the energy scale is lower. Most lattice problems (for example the Heisenberg model) do not have these properties, and so NRG is not appropriate. It turns out that the solution to getting a successful procedure for a one dimensional lattice system lies in the density matrix projection. Instead of using the lowest m eigenstates to truncate the Hamiltonian matrix at each step, DMRG uses the lowest m eigenstates of the ground state *density matrix* to perform the truncation. Using the density matrix does not necessarily choose the lowest energy eigenstates locally, but rather the states that are most *highly coupled* to the ground state [17].

2.1.2 Infinite System DMRG Procedure

The DMRG procedure is usually split into the *infinite* and *finite* algorithms. The infinite algorithm is used to grow a *superblock* up to a certain size, whereas the finite algorithm acts upon the superblock to accurately find the ground state of the system. The infinite algorithm is [16, 18]:

1. Start with a superblock of size $L = 4$ that is made up of a left (system) block of size $l = 1$; H_l, two single sites and a right (environment) block of size $l = 1$; H_l^R.

2. Form H_{l+1} and H_{l+1}^R within the superblock.

3. Form the superblock from H_{l+1} and H_{l+1}^R.

4. Diagonalise the superblock and find the ground state eigenvector $|\psi\rangle$.

5. Create the density matrix

$$\rho_{ii'} = \sum_j \psi_{ij}^* \psi_{i'j}. \tag{2.1.5}$$

6. Diagonalise the density matrix and keep the eigenvectors V corresponding to the lowest m eigenvalues.

7. Create matrix O that will be used to truncate the operators:

$$O = \begin{pmatrix} \vdots & \vdots & & \vdots \\ V_1 & V_2 & \dots & V_m \\ \vdots & \vdots & & \vdots \end{pmatrix} \tag{2.1.6}$$

8. Use O to change the basis and truncate the size of the Hamiltonian:

$$\bar{H}_{l+1} = O^\dagger H_{l+1} O \tag{2.1.7}$$

   and all of the operators:

$$\bar{A}_{l+1} = O^\dagger A_{l+1} O \tag{2.1.8}$$

   These will now be matrices of dimension $m \times m$.

9. Create a new superblock of size $L + 2$ from $\bar{H}_{l+1}$, two sites and $\bar{H}_{l+1}^R$.

10. Repeat from step 3 to grow the size of the superblock by two sites each iteration. The procedure is shown graphically in figure 2.1.

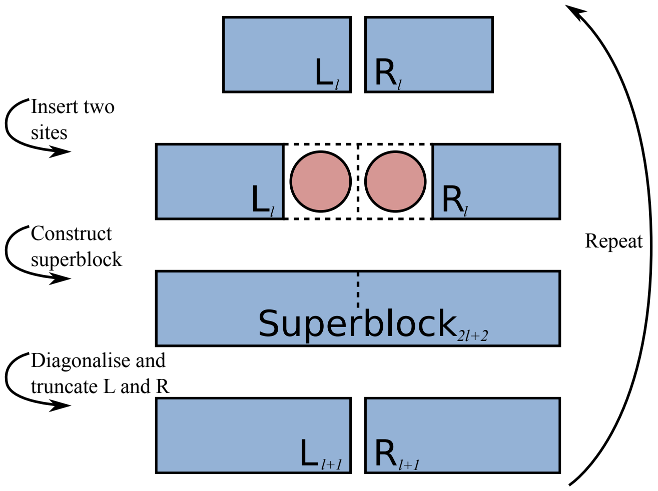

Figure 2.1: Pictorial representation of the infinite DMRG process. Starting with the left (system) and right (environment) blocks, each representing l sites, two sites are added to create a superblock. This is diagonalised and the ground state eigenvector used to make a density matrix, the eigenvectors corresponding to the lowest m eigenvalues of which are used to truncate the Hamiltonians for the new L and R blocks representing $l+1$ sites. The whole process is then repeated.

2.1.3 Finite System DMRG Procedure

The finite system DMRG algorithm uses the same basic principles that are used in the infinite algorithm, however the system is kept at a set size L and the aim is to calculate the energy of the desired state to a greater accuracy. The finite system algorithm is usually used after the length of the chain has grown to a desired size using the infinite algorithm. In the finite case the superblock is again constructed from a left (system) block, length l, two states and a right (environment) block, size l'. Now the system and environment blocks are no longer kept the same size so that the location on the chain that the two sites are inserted moves along the chain. The algorithm is performed as follows:

1. Perform the infinite algorithm until the chain length is the desired size L storing the $\bar{H}_l$, $\bar{H}^R_{l'}$ and all of the operators A_l that are required to construct the superblock at each step.

2. Now the system size is set to $L = l + 2 + l'$, begin growing the left block at the expense of the right block. Start with $l = \frac{L}{2}$ and $l' = \frac{L}{2} - 2$.

3. Perform steps 3-8 of the infinite algorithm.

4. Form a superblock from $\bar{H}_{l+1}$, two sites and $\bar{H}_{l'-1}$.

5. Proceed by performing steps 3 and 4 until the right hand edge of the system is reached, where $l' = 1$. This is the *right sweep.*

6. Now grow the right block at the expense of the left block. Perform steps 3 and 4 with the roles of l and l' reversed until the left hand edge of the system is reached, where $l = 1$. This is the *left sweep.*

7. Again, reverse the roles of l and l' and perform another left sweep.

8. Repeat, sweeping back and forth across the system until convergence is hit or a specified number of sweeps is reached.

DMRG proved to be a highly accurate and versatile numerical method with applications too numerous to list here, but comprehensive review articles provide many references [9, 19]. It was discovered by Östland and Rommer [20] that DMRG can be interpreted as an MPS. The MPS allows a greater understanding of the structure of information in the DMRG algorithm, giving reason for its successes and failures (see chapter 4). This has lead to further developments such as time evolution [21, 22, 23], systems in the thermodynamic limit [24] and efficient periodic boundary conditions (PBCs) [25, 26, 27]. The theory that has been developed in the context of MPS DMRG opens the door to more complex algorithms acting on networks of tensors with structures designed to suit the problem at hand. All of these *tensor network* algorithms have a lot in common with MPS DMRG, so the rest of this chapter will be devoted to a pedagogical overview of MPS.

2.2 Introduction to Matrix Product States

Consider a quantum system with basis states $|\uparrow\rangle$ and $|\downarrow\rangle$. Simple states can be formed as a product of these [28], for example

$$|\Psi\rangle = |\uparrow\rangle \otimes |\uparrow\rangle \,. \tag{2.2.1}$$

In general these states can be written as a product of the two sites

$$|\Psi\rangle = (a\,|\uparrow\rangle + b\,|\downarrow\rangle) \otimes (c\,|\uparrow\rangle + d\,|\downarrow\rangle) \,. \tag{2.2.2}$$

Using summation notation this can be an outer product of two element vectors

$$\begin{aligned} |\Psi\rangle &= \sum_{\sigma_1,\sigma_2} V_{\sigma_1} |\sigma_1\rangle \otimes V_{\sigma_2} |\sigma_2\rangle \\ &= \sum_{\sigma_1,\sigma_2} C_{\sigma_1,\sigma_2} |\sigma_1\rangle \otimes |\sigma_2\rangle , \end{aligned} \tag{2.2.3}$$

where σ_i can be $\uparrow$ or $\downarrow$ and C_{σ_1,σ_2} is a two component tensor with elements

$$C_{\sigma_1,\sigma_2} = \begin{pmatrix} ac & ad \\ bc & bd \end{pmatrix} . \tag{2.2.4}$$

Product states such as these have the bases independent of each other and the expectation values factorise. On the other hand, entangled states such as

$$|\Psi\rangle = \frac{1}{\sqrt{2}} \left(|\uparrow\rangle \otimes |\downarrow\rangle - |\downarrow\rangle \otimes |\uparrow\rangle \right) \tag{2.2.5}$$

do not have factorising expectation values. The maximally entangled state (2.2.5) would require a tensor of the form

	$\sigma_2 =\downarrow$	$\sigma_2 =\uparrow$
$\sigma_1 =\downarrow$	0	$-1/\sqrt{2}$
$\sigma_1 =\uparrow$	$1/\sqrt{2}$	0

which cannot be described by eq. (2.2.4). If instead of being described by vectors, let V_{σ_1} and V_{σ_2} be matrices [29], for example

$$M_{\sigma_1,i} = \frac{1}{\sqrt[4]{2}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad , \quad M_{i,\sigma_2} = \frac{1}{\sqrt[4]{2}} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \tag{2.2.6}$$

These matrices are then combined using a standard matrix product over the i index, which gives the entangled state

$$\sum_i M_{\sigma_1,i} M_{i,\sigma_2} = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} , \tag{2.2.7}$$

as desired. The i index introduces entanglement between the two states and can be thought of as a form of bond.

Equation (2.2.2) can be generalised to a lattice of L states with open boundary conditions (OBCs). Here, a product state is given by

$$|\Psi\rangle = \prod_{i=1}^{L} \left[a_i(\uparrow) \left|\uparrow\right\rangle_i + a_i(\downarrow) \left|\downarrow\right\rangle_i \right], \tag{2.2.8}$$

where there are two coefficients, $a_i(\uparrow)$ and $a_i(\downarrow)$, for each site [28]. Equation (2.2.3) is generalised to include an L index tensor $C_{\sigma_1 \ldots \sigma_L}$ and the wavefunction becomes

$$|\Psi\rangle = \sum_{\sigma_1,\ldots,\sigma_L} C_{\sigma_1 \ldots \sigma_L} |\sigma_1\rangle \otimes \cdots \otimes |\sigma_L\rangle . \tag{2.2.9}$$

As with the two particle case, the L index tensor can be split into a series of local tensors with connections to their neighbours that allow the inclusion of entanglement. When away from the boundaries each site has two neighbours thus the tensors at each site have three indices; one for the site basis and one for each neighbour. In full the wavefunction takes the form

$$|\Psi\rangle = \sum_{\sigma_1,\ldots,\sigma_L} \sum_{a_1,\ldots,a_{L-1}} M_{\sigma_1,a_1} M_{\sigma_2,a_1 a_2} \ldots M_{\sigma_{L-1},a_{L-2}a_{L-1}} M_{\sigma_L,a_{L-1}} |\sigma_1, \ldots, \sigma_L\rangle , \tag{2.2.10}$$

where $|\sigma_1 \ldots \sigma_L\rangle$ is a short hand for $|\sigma_1\rangle \otimes \cdots \otimes |\sigma_L\rangle$. For the remainder of the thesis the $\otimes$ will be omitted for simplicity but is implied in the product of basis vectors. The σ_i indices label the spins of the basis and are known as the *physical indices*, whereas the a_i are the *bond*, *virtual* or *auxiliary indices*. To draw a distinction between the two index types it is convention to have the physical σ_i as upper indices, thus giving the standard form of an MPS

$$|\Psi\rangle = \sum_{\sigma_1,\ldots,\sigma_L} \sum_{a_1,\ldots,a_{L-1}} M^{\sigma_1}_{a_1} M^{\sigma_2}_{a_1 a_2} \ldots M^{\sigma_L}_{a_{L-2}a_{L-1}} M^{\sigma_L}_{a_{L-1}} |\sigma_1, \ldots, \sigma_L\rangle . \tag{2.2.11}$$

Taking the hermitian conjugate gives the bra state

$$\langle\Psi| = \sum_{\sigma_1,\ldots,\sigma_L} \sum_{a_1,\ldots,a_{L-1}} M^{*\sigma_1}_{a_1} M^{*\sigma_2}_{a_1 a_2} \ldots M^{*\sigma_{L-1}}_{a_{L-2}a_{L-1}} M^{*\sigma_L}_{a_{L-1}} \langle\sigma_1, \ldots, \sigma_L| . \tag{2.2.12}$$

The MPS is a valuable tool in the computation of quantum many body states as, by controlling the size of the matrices one can introduce enough entanglement to model a local Hamiltonian but keep the Hilbert space small enough such that calculations are tractable. This is know as setting the *bond dimension*, χ, and within a DMRG algorithm is equivalent to m, the number of eigenvectors to keep when diagonalising

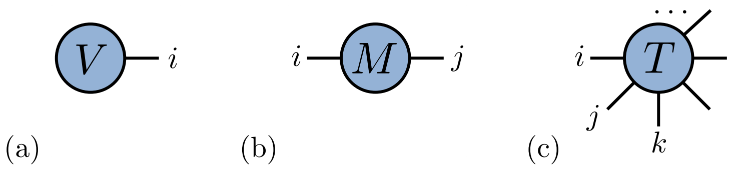

Figure 2.2: Diagrammatic representations of (a) a vector, (b) a matrix, and (c) a general tensor. The shape represents the object and the lines or *legs* the indices.

the density matrix (see 2.1.2).

2.3 Diagrammatic Notation

One of the most elegant features of the tensor network formalism is the diagrammatic representation of the mathematics, which makes visualising the algorithms very easy. A tensor is drawn as a shape (here circles) where each index is represented by a line or *leg*. In this manner a vector, which is a one index tensor, is a circle with one line coming from it. Similarly a matrix is a circle with two legs and a general tensor with k indices is a circle with k legs. Examples of these are given in fig. 2.2.

Multiplication of tensors is generalised by the notion of tensor contraction. The simplest example is a vector inner product that takes two vectors to give a scalar

$$S = \sum_i V_i V'_i. \tag{2.3.1}$$

In the tensor network diagrammatic convention the contraction is drawn by joining the lines that represent the summed over index, resulting in a object with no legs as shown in fig. 2.3(a). Multiplication of a vector with a matrix is

$$V'_j = \sum_i V_i M_{ij}, \tag{2.3.2}$$

drawn in fig. 2.3(b). General contraction is the summation over the repeated index of any two tensors, for example

$$C_{ij\ldots i'j'\ldots} = \sum_k A_{ij\ldots k\ldots} B_{i'j'\ldots k\ldots}, \tag{2.3.3}$$

the diagram of which is fig. 2.3(c). The standard form of the MPS (2.2.11) is therefore a set of L tensors where each is contracted by the bond indices to its neighbours.

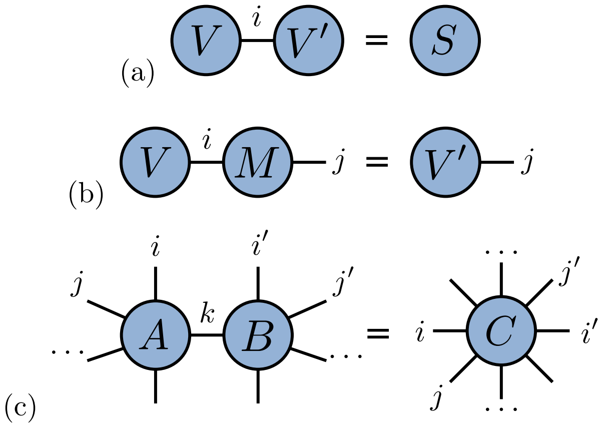

Figure 2.3: Diagrammatic representations of contractions. (a) Inner product of a vector with another vector (eq. 2.3.1). (b) Vector multiplying a matrix (eq. 2.3.2). (c) General tensor contraction (eq. 2.3.3).

The standard diagram for an MPS is a chain of circles connected horizontally by the bond indices with the physical indices drawn vertically. The MPS of eq. (2.2.11) is given in fig. 2.4(a). The bra state MPS (2.2.12) will be drawn in a similar way to the ket but with the physical indices pointing in the opposite direction as shown in fig. 2.4(b).

There is no set convention as to whether the physical indices for the wavefunction should point up or down and the literature contains examples of both (up: [29, 30, 31, 32, 33], down: [2, 34, 35, 36, 37, 38, 39, 40, 41]). Throughout this document a ket state will have physical indices that point down to be consistent with the conventions of the literature on holographic tensor network states [12, 34, 42, 43, 44, 45] that will become important in latter parts of the thesis.

2.4 Fundamental Details

Tensor networks require operations that manipulation the tensors, the most fundamental of which are *reshaping* and contraction. It is using these actions that all of the more complicated tensor network algorithms are built.

2.4.1 Tensor Reshaping

The reshaping of the tensors involves *permutation*, *fusing* and *splitting* of indices. Permutation is the reordering of the indices of a tensor, generalising the notion of a

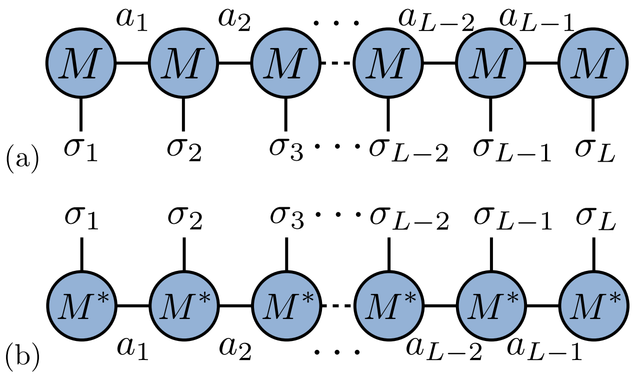

Figure 2.4: Diagrammatic representation of an MPS (a) ket given by eq. (2.2.11) and (b) bra given by eq. (2.2.12). The circles represent the M tensors and the lines are the tensor indices. The horizontal lines represent the bond indices, the vertical lines the physical indices.

transpose operation on a matrix. For example $A^{a_1 a_2 a_3}$ can be permuted to $A^{a_2 a_1 a_3}$ as shown by fig. 2.5(a). Permutation and limited reshaping functions for tensors are common and exist in packages such as MATLAB and NumPy, but general fusion and splitting functions are not.

Fusion involves combining two or more indices to create a composite index that spans all combinations of the original indices, the following is a description of how to create such a function and is shown diagrammatically in fig. 2.5(b). Begin by labelling the indices of the tensor by the desired final order with the legs that are to be fused given a positive number, the indices to be left given a negative number. The indices are then permuted to be in the order of the labels, with the convention that the diagrams are labelled from 9 o'clock anticlockwise as in [37]. Note that now the two indices that are to be fused are next to each other. To fuse these indices it is necessary to calculate the dimension of the resulting composite index, which is product of the component indices. The next step is to reshape the tensor into the desired form. This is performed by flattening the tensor into a vector then splitting it up into indices of the desired dimension (in MATLAB, for example, this is done by the *reshape* function) where the final index is now the composite index. The final step is to permute the reshaped tensor such that the indices are in the desired order. Here we note that the distinction between indices to be fused and not is no longer needed so the negative indices are made positive to allow the permutation to be made in the correct order.

The split function is in essence the inverse of the fusion function; it splits an index into smaller component indices. Begin this function by defining the index that

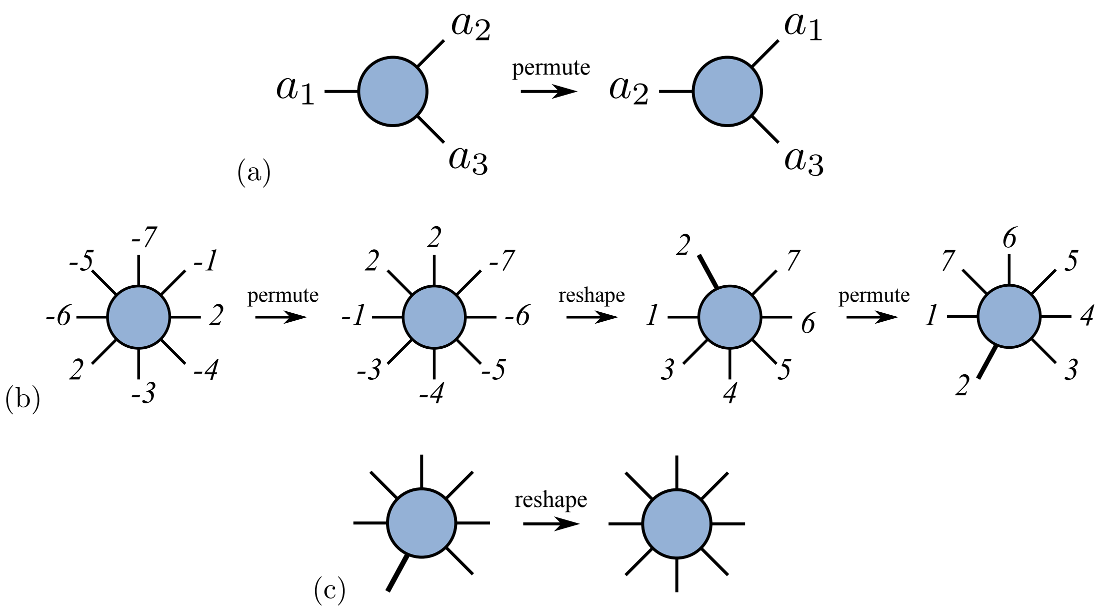

Figure 2.5: Example of (a) permutation of indices, (b) the tensor fusion function and (c) the tensor split function as described in the text.

will be split and the sizes of the desired component indices after splitting. Create a list of the index dimensions for the tensor and insert the split dimensions in place of the dimension of the index to be split then reshape the tensor according to this list of index sizes. This is diagrammatically shown in fig. 2.5(c). It is important that the fuse and split have the same indexing convention such that splitting after a fusion recovers the same tensor. Note that MATLAB like Fortran is *column-major*, whereas NumPy like C and Mathematica is *row-major*.

2.4.2 Tensor Contraction

The most commonly performed action in a tensor network algorithm is the tensor contraction, so it is vital that it is done fast. As mentioned in section 2.3, the concept of a tensor contraction is a generalisation of a matrix multiplication to objects of arbitrary dimension. Matrix multiplication algorithms are very common and there exist very highly optimised functions in linear algebra packages such as BLAS and LAPACK, which form the basis of the tensor contraction function. To use the matrix multiplication functions, the tensors must first be reshaped into matrices, multiplied and then converted back into tensors.

This description of a tensor contraction follows the example in [37]. The fact that there can be more than two indices to the tensor means that there are

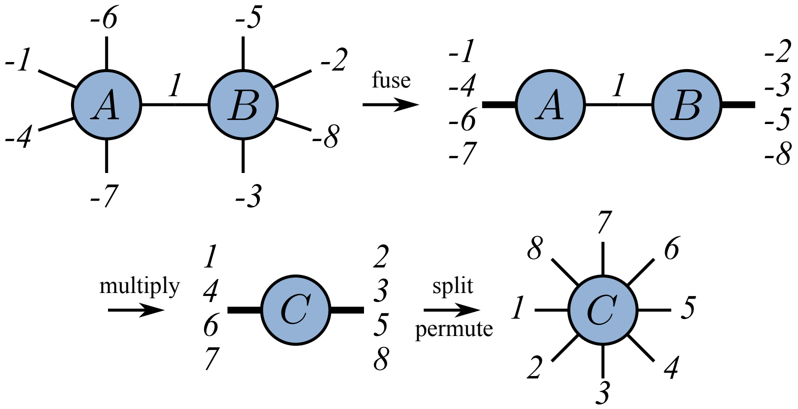

Figure 2.6: Diagrammatic form of the tensor contraction function as described in the text. Note that the minus signs are dropped after the multiplication.

often many different ways that two tensors can be contracted depending on which index is being summed over. Therefore in a similar manner to the fusion function, the contraction function requires instruction as to which indices to contract and the desired order of the indices of the resulting tensor, where the indices to be contracted are positive and those to be left are negative. Using the numbering, permute the tensors such that the indices to be contracted are the final indices on the left hand tensor, A, and the first indices on the right hand tensor, B. Fuse the indices that are not contracted on each tensor leaving two matrices, each with one index to be contracted. These matrices can then be efficiently multiplied resulting in a matrix, C, with one index being a composite of the un-contracted indices of A and the other the un-contracted indices of B. These two legs are then split and permuted into the desired form. Similar to the fusion example, after multiplication there are no indices to be contracted over therefore the negativity of the remaining indices can be dropped so that the permutation achieves the desired result. Figure 2.6 shows a diagrammatic example of the mechanics of the tensor contraction.

2.4.3 Time Estimates

The order in which a tensor network is contracted can have a great effect on the execution time of an algorithm, therefore it is useful to have a means of estimating this cost so that the optimal order of contraction can be found. The diagrammatic notation provides a simple method for this. Using *big O notation*, the *order* of an operation is given by the product of the dimensions of the indices involved. For

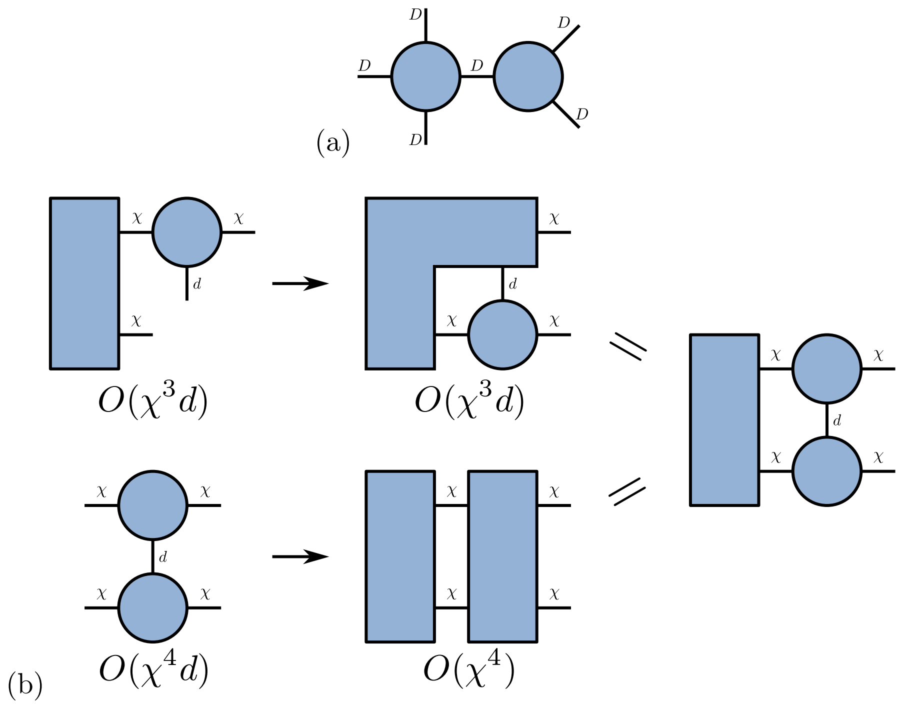

Figure 2.7: Contraction of (a) two tensors with computational costing $O(D^6)$ and (b) Contraction of three tensors, showing that the order of contraction effects the cost of an operation.

example contraction of a rank four and a rank three tensor as shown in fig. 2.7(a) where each index has D elements is of order D^6 or $O(D^6)$.

An simple example of where the order of contraction effects the overall cost of an operation is shown in fig. 2.7(b). This is the contraction of two MPS tensors with a left block, which is part of calculating an expectation value, the details of which are in section 2.5.4. The top order is $O(\chi^3 d)$, whereas the bottom is $O(\chi^4 d)$. With an MPS the bond dimension χ can be $\sim$ 100 - 1000 so the second order takes about *a thousand times longer*, clearly showing the advantage of choosing an optimal order of contraction.

2.4.4 Singular Value Decomposition

Another commonly used tool in the manipulation of tensor network states is the *singular value decomposition* (SVD) [30], which states that a rectangular matrix A of dimension $N_A \times N_B$ can always be written in the form

$$A = USV^{\dagger}. \tag{2.4.1}$$

U is an $N_A \times \min(N_A, N_B)$ matrix with orthonormal columns ($U^\dagger U = \mathbb{1}$), which is unitary if $N_A = N_B$ ($UU^\dagger = U^\dagger U = \mathbb{1}$). S is diagonal with non-negative entries, which are the *singular values* and are usually given in order of decreasing size. $V^\dagger$ is a $\min(N_A, N_B) \times N_B$ matrix with orthogonal rows ($(V^\dagger)(V^\dagger)^\dagger = \mathbb{1}$), which is also unitary if $N_A = N_B$.

For a system of more than two sites, with the bases split into two blocks A and B with dimensions N_A and N_B respectively, a state of the system $|\Psi\rangle$ can be written as [30]

$$|\Psi\rangle = \sum_{i,j} \Psi_{ij} \left|\sigma_i\right\rangle_A \left|\sigma_j\right\rangle_B , \tag{2.4.2}$$

where $\left|\sigma_i\right\rangle_A$ and $\left|\sigma_j\right\rangle_B$ are orthonormal bases of A and B respectively, and the coefficients are elements of matrix Ψ_{ij}. Performing an SVD of Ψ_{ij} in eq. (2.4.2) gives

$$\begin{aligned} |\Psi\rangle &= \sum_{i,j} \Psi_{ij} \left|\sigma_i\right\rangle_A \left|\sigma_j\right\rangle_B \\ &= \sum_{i,j,a} U_{ia} S_{aa} V^\dagger_{aj} \left|\sigma_i\right\rangle_A \left|\sigma_j\right\rangle_B , \end{aligned} \tag{2.4.3}$$

This is also known as a *Schmidt decomposition.*

The amount of entanglement between blocks A and B is encoded within the singular values, which is quantified by the *von Neumann* or *entanglement entropy*

$$S_{\mathrm{A|B}} = -\sum_{a=1} s_a^2 \log_2 s_a^2, \tag{2.4.4}$$

where s_a are the singular values from diagonal matrix S. When the wavefunction is normalised the set of singular values square and sum to 1. If only one of the s_a is non-zero, then the system is in a product state with $S_{A|B} = 0$. The other extreme is having all of the singular values equal, i.e. $s_a = 1/\sqrt{N}$ where N is the dimension of matrix S. This *maximally entangled* state therefore has entropy $S_{A|B} = \log_2 N$.

2.5 Matrix Product States

2.5.1 The Canonical Form

It is possible to take any state with tensor $C^{\sigma_1 \dots \sigma_L}$ and convert it into an MPS [30]. The tensor is first reshaped into a matrix $C'^{[1]}_{\sigma_1,(\sigma_2 \dots \sigma_L)}$, where the second index is a composite of the spin indices for sites 2 - L. The matrix $C'^{[1]}$ can then be split into two by performing an SVD

$$
\begin{aligned}
C^{\sigma_1\ldots\sigma_L} = C'^{[1]}_{\sigma_1,(\sigma_2\ldots\sigma_L)} &= \sum_{a_1} U^{[1]}_{\sigma_1,a_1} S^{[1]}_{a_1,a_1} V^{[1]\dagger}_{a_1,(\sigma_2\ldots\sigma_L)} \\
&= \sum_{a_1} A^{\sigma_1}_{a_1} C^{\sigma_2\ldots\sigma_L}_{a_1} \\
&= \sum_{a_1} A^{\sigma_1}_{a_1} C'^{[2]}_{(a_1\sigma_2),(\sigma_3\ldots\sigma_L)}, \qquad (2.5.1)
\end{aligned}
$$

where $U^{[1]}_{\sigma_1,a_1}$ has been reshaped into $A^{\sigma_1}_{a_1}$. $S^{[1]}$ has been contracted with $V^{[1]\dagger}$ and been reshaped to obtain a tensor that now represents all physical indices apart from σ_1. This can then be reshaped into a matrix $C'^{[2]}_{(a_1\sigma_2),(\sigma_3\ldots\sigma_L)}$ and split by SVD

$$
\begin{aligned}
C^{\sigma_1\ldots\sigma_L} &= \sum_{a_1} A^{\sigma_1}_{a_1} \sum_{a_2} U^{[2]}_{(a_1\sigma_2),a_2} S^{[2]}_{a_2,a_2} V^{[2]\dagger}_{a_2,(\sigma_3\ldots\sigma_L)} \\
&= \sum_{a_1,a_2} A^{\sigma_1}_{a_1} A^{\sigma_2}_{a_1a_2} C^{\sigma_3\ldots\sigma_L}_{a_2}, \qquad (2.5.2)
\end{aligned}
$$

where again $U^{[2]}_{(a_1\sigma_2),a_2}$ has been reshaped into $A^{\sigma_2}_{a_1a_2}$ and $C^{\sigma_3\ldots\sigma_L}_{a_2}$ has been made from the contraction and reshaping of $S^{[2]}$ and $U^{[2]}$. This process can be repeated for all L sites of the chain giving

$$
C_{\sigma_1\ldots\sigma_L} = \sum_{a_1,\ldots,a_{L-1}} A^{\sigma_1}_{a_1} A^{\sigma_2}_{a_1a_2} \ldots A^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} A^{\sigma_L}_{a_{L-1}}. \qquad (2.5.3)
$$

Inserting into eq. (2.2.9) gives

$$
|\Psi\rangle = \sum_{\sigma_1,\ldots,\sigma_L} \sum_{a_1,\ldots,a_{L-1}} A^{\sigma_1}_{a_1} A^{\sigma_2}_{a_1a_2} \ldots A^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} A^{\sigma_L}_{a_{L-1}} |\sigma_1,\ldots,\sigma_L\rangle, \qquad (2.5.4)
$$

which is the explicit *left-canonical form* of an MPS. The diagrammatic derivation is given by fig. 2.8. The fact that the A matrices were obtained from the U of an SVD means that it is *left-normalised*, i.e. it satisfies

$$
\begin{aligned}
\sum_{(a_{i-1},\sigma_i)} U^{\dagger}_{(a_{i-1}\sigma_i),a'_i} U_{(a_{i-1}\sigma_i),a_i} &= \delta_{a'_i,a_i}, \\
\sum_{(a_{i-1},\sigma_i)} U^{*}_{a'_i,(a_{i-1}\sigma_i)} U_{(a_{i-1}\sigma_i),a_i} &= \delta_{a'_i,a_i}, \\
\sum_{\sigma_i} \sum_{a_{i-1}} A^{*\sigma_i}_{a_{i-1}a'_i} A^{\sigma_i}_{a_{i-1}a_i} &= \delta_{a'_i,a_i}, \qquad (2.5.5)
\end{aligned}
$$

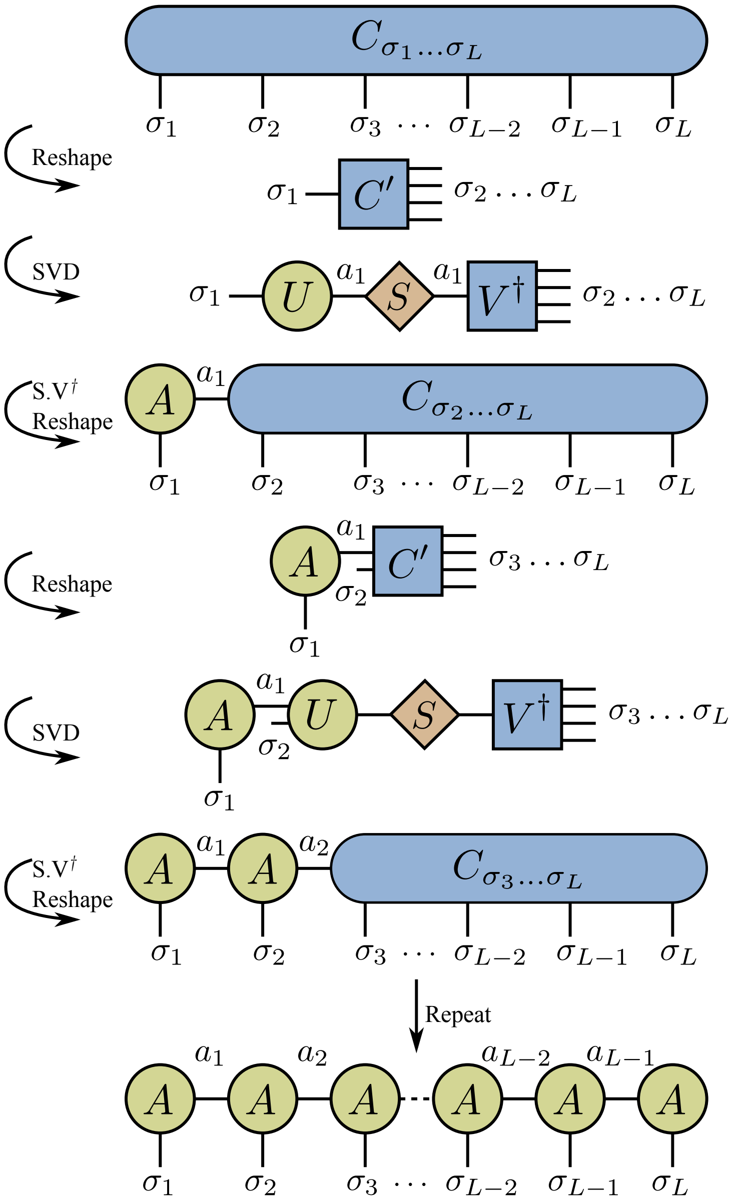
$C_{\sigma_1 \ldots \sigma_L}$
σ_1 σ_2 σ_3 $\cdots$ σ_{L-2} σ_{L-1} σ_L
Reshape
σ_1 C' $\sigma_2 \ldots \sigma_L$
SVD
σ_1 U a_1 S a_1 $V^\dagger$ $\sigma_2 \ldots \sigma_L$
S.V†
Reshape
A a_1 $C_{\sigma_2 \ldots \sigma_L}$
σ_1 σ_2 σ_3 $\cdots$ σ_{L-2} σ_{L-1} σ_L
Reshape
A a_1 C' $\sigma_3 \ldots \sigma_L$ σ_2 σ_1
SVD
A a_1 U S $V^\dagger$ $\sigma_3 \ldots \sigma_L$ σ_2 σ_1
S.V†
Reshape
A a_1 A a_2 $C_{\sigma_3 \ldots \sigma_L}$
σ_1 σ_2 σ_3 $\cdots$ σ_{L-2} σ_{L-1} σ_L
Repeat
a_1 a_2 a_{L-2} a_{L-1}
A A A A A A
σ_1 σ_2 σ_3 $\cdots$ σ_{L-2} σ_{L-1} σ_L

Figure 2.8: Diagrammatic representation of the derivation of the left canonical form of an MPS. The first line shows eq. (2.2.9); the initial general state of L sites. This is reshaped into a matrix to which an SVD is performed as in eq. (2.5.1). The diamond tensor is the matrix of singular values D. U is reshaped to A and D is contracted with $V^\dagger$ and reshaped to give the general tensor of $L-1$ states. This is repeated to give the left normalised MPS as desired.

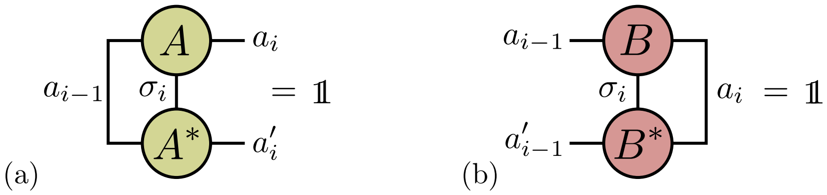

Figure 2.9: Diagrammatic representation of (a) left-normalisation given by eq. (2.5.5) and (b) right-normalisation given by eq. (2.5.9).

or diagrammatically in fig. 2.9(a).

Equation (2.5.4) is obtained by starting at site 1 and working towards site L. The start point of the derivation does not alter the state, thus an equivalent MPS can be derived starting from L and working towards 1,

$$\begin{aligned} C_{\sigma_1\ldots\sigma_L} = C'^{[L]}_{(\sigma_1\ldots\sigma_{L-1}),\sigma_L} &= \sum_{a_{L-1}} U^{[L]}_{(\sigma_1\ldots\sigma_{L-1}),a_{L-1}} S^{[L]}_{a_{L-1},a_{L-1}} V^{[L]\dagger}_{a_{L-1},\sigma_L} \\ &= \sum_{a_{L-1}} C'^{[L-1]}_{(\sigma_1\ldots\sigma_{L-2}),(\sigma_{L-1}a_{L-1})} B^{\sigma_L}_{a_{L-1}} \\ &= \sum_{a_{L-2},a_{L-1}} C'^{[L-2]}_{(\sigma_1\ldots\sigma_{L-3}),(\sigma_{L-2}a_{L-2})} B^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} B^{\sigma_L}_{a_{L-1}} \\ &= \ldots, \\ &= \sum_{a_1,\ldots,a_{L-1}} B^{\sigma_1}_{a_1} B^{\sigma_2}_{a_1a_2} \ldots B^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} B^{\sigma_L}_{a_{L-1}}, \end{aligned} \tag{2.5.6}$$

giving

$$|\Psi\rangle = \sum_{\sigma_1,\ldots,\sigma_L} \sum_{a_1,\ldots,a_{L-1}} B^{\sigma_1}_{a_1} B^{\sigma_2}_{a_1,a_2} \ldots B^{\sigma_{L-1}}_{a_{L-2},a_{L-1}} B^{\sigma_L}_{a_{L-1}} |\sigma_1,\ldots,\sigma_L\rangle\,. \tag{2.5.7}$$

This is the *right-canonical form* of an MPS. The B matrices are reshaped $V^\dagger$ matrices which satisfy

$$\begin{aligned} V^\dagger V &= \mathbb{1} \\ \Rightarrow \quad (V^\dagger)(V^\dagger)^\dagger &= \mathbb{1}, \end{aligned} \tag{2.5.8}$$

hence the B matrices are *right-normalised*

$$\sum_{\sigma_i}\sum_{a_i} B^{\sigma_i}_{a_{i-1}a_i} B^{*\sigma_i}_{a'_{i-1}a_i} = \delta_{a'_{i-1},a_{i-1}}. \tag{2.5.9}$$

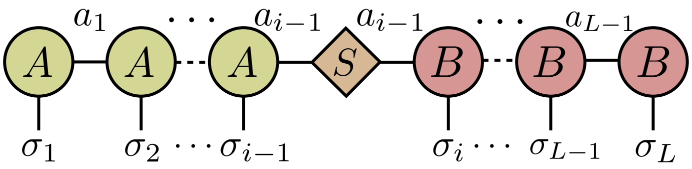

Figure 2.10: Diagrammatic representation of an MPS with *mixed normalisation*.

The diagrammatic form is given by fig. 2.9(b).

By starting the normalisation process as both ends it is possible to have a *mixed canonical* state, where the left side is left normalised and the right side is right normalised. At the point where the two normalisations meet there will be a leftover matrix of singular values as shown in fig. 2.10. This matrix can be contracted with either neighbouring tensor to make a tensor that does not have either right or left normalisation. These states will prove useful in calculating expectation values in the next section. These normalisations do not have any effect on expectation values; as such they are different *gauges*.

2.5.2 Overlaps

Orthonormality of the basis vectors and normalisation of the wavefunctions implies that we expect $\langle\Psi|\Psi\rangle = 1$, which is easy to show using the right normalisation of the A tensors [30]

$$\begin{aligned}
\langle\Psi|\Psi\rangle &= \sum_{\sigma_1,\dots,\sigma_L} \sum_{\substack{a_1,\dots,a_{L-1}\\ a'_1,\dots,a'_{L-1}}} A^{*\sigma_1}_{a'_1} A^{*\sigma_2}_{a'_1 a'_2} \dots A^{*\sigma_L}_{a'_{L-1}} \\
&\qquad\qquad \times A^{\sigma_1}_{a_1} A^{\sigma_2}_{a_1 a_2} \dots A^{\sigma_L}_{a_{L-1}} \langle \sigma_1,\dots,\sigma_L | \sigma_1,\dots,\sigma_L \rangle \\
&= \sum_{\sigma_2,\dots,\sigma_L} \sum_{\substack{a_1,\dots,a_{L-1}\\ a'_1,\dots,a'_{L-1}}} \left[\sum_{\sigma_1} A^{*\sigma_1}_{a'_1} A^{\sigma_1}_{a_1} \right] A^{*\sigma_2}_{a'_1 a'_2} \dots A^{*\sigma_L}_{a'_{L-1}} A^{\sigma_2}_{a_1 a_2} \dots A^{\sigma_L}_{a_{L-1}} \\
&= \sum_{\sigma_2,\dots,\sigma_L} \sum_{\substack{a_1,\dots,a_{L-1}\\ a'_1,\dots,a'_{L-1}}} \delta_{a'_1 a_1} A^{*\sigma_2}_{a'_1 a'_2} \dots A^{*\sigma_L}_{a'_{L-1}} A^{\sigma_2}_{a_1 a_2} \dots A^{\sigma_L}_{a_{L-1}},
\end{aligned} \tag{2.5.10}$$

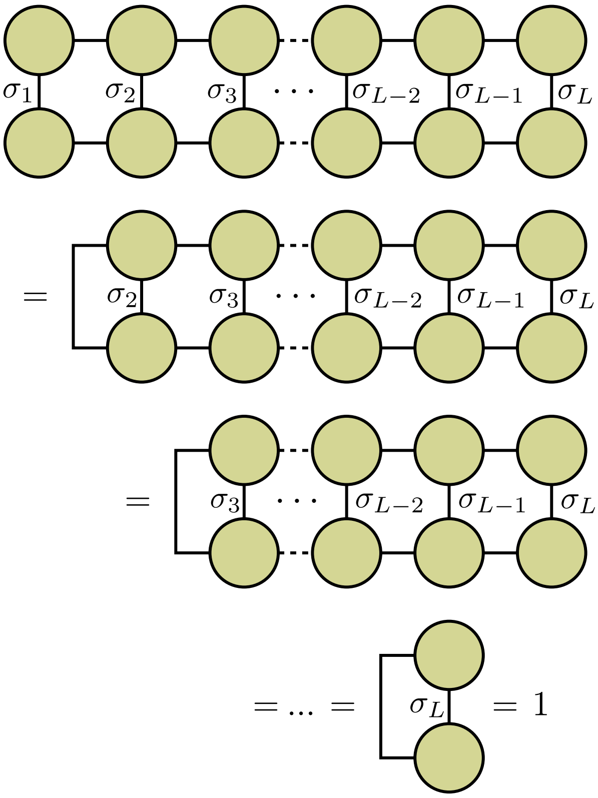

Figure 2.11: Diagrammatic derivation of the normalisation of a bra-ket as in eq. (2.5.11).

where we have used the conditions on A in eq. (2.5.5). Repeating until the end of the chain yields the result

$$\begin{aligned}
\langle\Psi|\Psi\rangle &= \sum_{\sigma_3,\ldots,\sigma_L} \sum_{\substack{a_2,\ldots,a_{L-1}\\ a'_2,\ldots,a'_{L-1}}} \left[\sum_{\sigma_2}\sum_{a_1} A^{*\sigma_2}_{a_1 a'_2} A^{\sigma_2}_{a_1 a_2}\right] A^{*\sigma_3}_{a'_2 a'_3} \ldots A^{*\sigma_L}_{a'_{L-1}} A^{\sigma_3}_{a_2 a_3} \ldots A^{\sigma_L}_{a_{L-1}} \\
&= \sum_{\sigma_3,\ldots,\sigma_L} \sum_{\substack{a_2,\ldots,a_{L-1}\\ a'_2,\ldots,a'_{L-1}}} \delta_{a'_2 a_2} A^{*\sigma_3}_{a'_2 a'_3} \ldots A^{*\sigma_L}_{a'_{L-1}} A^{\sigma_3}_{a_2 a_3} \ldots A^{\sigma_L}_{a_{L-1}} \\
&= \ldots, \\
&= \sum_{\sigma_L}\sum_{a_{L-1}} A^{*\sigma_L}_{a_{L-1}} A^{\sigma_L}_{a_{L-1}} \\
&= 1. \qquad (2.5.11)
\end{aligned}$$

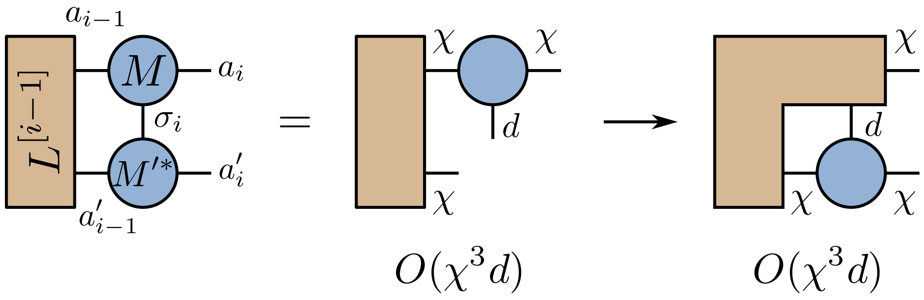

Figure 2.12: Construction of the left block L in derivation (2.5.12) tensor by tensor.

This derivation can be performed very neatly pictorially as in fig. 2.11. A similar derivation gives the same result for right-normalised B matrices. The overlap of two wavefunctions ($\langle\Phi|\Psi\rangle$) cannot make use of this convenient normalisation, so it is necessary to contract all of the terms. If each wavefunction is represented by an MPS with a bond dimension χ this can still be done efficiently

$$\begin{aligned}
\langle\Phi|\Psi\rangle &= \sum_{\sigma_1,\dots,\sigma_L} \sum_{\substack{a_1,\dots,a_{L-1}\\ a'_1,\dots,a'_{L-1}}} M'^{*\sigma_1}_{a'_1} M'^{*\sigma_2}_{a'_1 a'_2} \dots M'^{*\sigma_L}_{a'_{L-1}} \\
&\qquad\qquad \times M^{\sigma_1}_{a_1} M^{\sigma_2}_{a_1 a_2} \dots M^{\sigma_L}_{a_{L-1}} \langle\sigma_1,\dots,\sigma_L|\sigma_1,\dots,\sigma_L\rangle \\
&= \sum_{\sigma_2,\dots,\sigma_L} \sum_{\substack{a_1,\dots,a_{L-1}\\ a'_1,\dots,a'_{L-1}}} L^{[1]}_{a_1 a'_1} M'^{*\sigma_2}_{a'_1 a'_2} \dots M'^{*\sigma_L}_{a'_{L-1}} M^{\sigma_2}_{a_1 a_2} \dots M^{\sigma_L}_{a_{L-1}} \\
&= \dots \\
&= \sum_{\sigma_i,\dots,\sigma_L} \sum_{\substack{a_{i-1},\dots,a_{L-1}\\ a'_{i-1}\dots,a'_{L-1}}} L^{[L-1]}_{a_{i-1} a'_{i-1}} M'^{*\sigma_i}_{a'_{i-1} a'_i} \dots M'^{*\sigma_L}_{a'_{L-1}} M^{\sigma_i}_{a_{i-1} a_i} \dots M^{\sigma_L}_{a_{L-1}} \\
&= \dots,
\end{aligned} \tag{2.5.12}$$

where the M are the MPS tensors for $|\Psi\rangle$ and M'^* are the tensors for $\langle\Phi|$. The calculation is performed one tensor at a time form left to right, grouping the contracted tensors into a *block* L. Figure 2.12 shows the contraction of one site into the block, which is performed in $O(\chi^3 d)$ where d is the *basis dimension*, or the number of elements in σ_i. Hence the whole overlap is calculated in $O(L\chi^3 d)$.

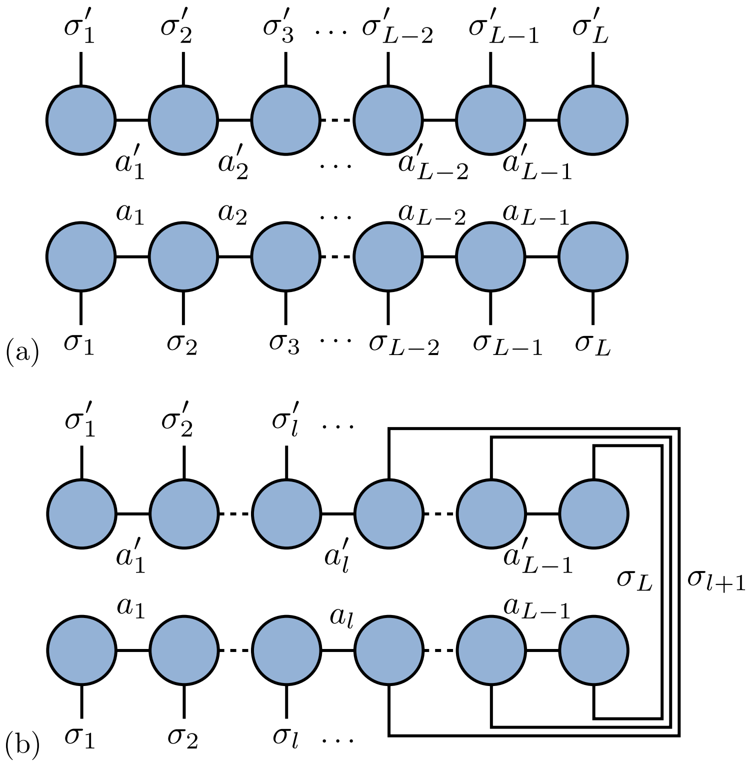
σ'_1 σ'_2 σ'_3 … σ'_{L-2} σ'_{L-1} σ'_L
a'_1 a'_2 … a'_{L-2} a'_{L-1}
a_1 a_2 … a_{L-2} a_{L-1}
(a)
σ_1 σ_2 σ_3 … σ_{L-2} σ_{L-1} σ_L
σ'_1 σ'_2 σ'_l …
a'_1 a'_l a'_{L-1}
σ_L σ_{l+1}
a_1 a_l a_{L-1}
(b)
σ_1 σ_2 σ_l …

Figure 2.13: (a) MPS density operator $\hat{\rho}$ of eq. (2.5.14) and (b) reduced density operator $\hat{\rho}_A$ of eq. (2.5.16), where sites $l+1$ to L have been traced over.

2.5.3 The Density Operator

The density operator for a wavefunction $|\Psi\rangle$ is defined as

$$\hat{\rho} = |\Psi\rangle \langle\Psi| . \tag{2.5.13}$$

For an MPS this takes the form [30]

$$\hat{\rho} = \sum_{\substack{\sigma_1,\ldots,\sigma_L \\ \sigma'_1,\ldots,\sigma'_L}} \sum_{\substack{a_1,\ldots,a_L \\ a'_1,\ldots,a'_L}} M^{\sigma_1}_{a_1} \ldots M^{\sigma_L}_{a_{L-1}} M^{*\sigma'_1}_{a'_1} \ldots M^{*\sigma'_L}_{a'_{L-1}} |\sigma_1,\ldots,\sigma_L\rangle \langle\sigma'_1,\ldots,\sigma'_L| , \tag{2.5.14}$$

or diagrammatically as in fig. 2.13(a). Like the singular values of an SVD, the density operator contains the entanglement information of the state. *Reduced density operators* are calculated by tracing over the degrees of freedom of the system. Most relevant here are where spatial regions of the system are traced over, for example let the left l sites of the chain be region A and the remaining sites be region B. The reduced density matrices are then

$$\hat{\rho}_A = \mathrm{Tr}_B \left|\Psi\right\rangle\left\langle\Psi\right| \quad , \quad \hat{\rho}_B = \mathrm{Tr}_A \left|\Psi\right\rangle\left\langle\Psi\right| . \tag{2.5.15}$$

$\hat{\rho}_A$ can be expressed in MPS form as

$$\begin{aligned}\hat{\rho}_A = \sum_{\substack{\sigma_1,\ldots,\sigma_L \\ \sigma'_1,\ldots,\sigma'_l}} \sum_{\substack{a_1,\ldots,a_L \\ a'_1,\ldots,a'_L}} & M^{\sigma_1}_{a_1} \ldots M^{\sigma_l}_{a_{l-1}a_l} M^{\sigma_{l+1}}_{a_l a_{l+1}} \ldots M^{\sigma_L}_{a_{L-1}} \\ & \times M^{*\sigma'_1}_{a'_1} \ldots M^{*\sigma'_l}_{a'_{l-1}a'_l} M^{*\sigma_{l+1}}_{a'_l a'_{l+1}} \ldots M^{*\sigma_L}_{a'_{L-1}} \left|\sigma_1,\ldots,\sigma_l\right\rangle \left\langle\sigma'_1,\ldots,\sigma'_l\right| ,\end{aligned} \tag{2.5.16}$$

which is shown in fig. 2.13(b). The contracted tensors of the B block form the *reduced density matrix* ρ_A

$$\rho_A = \sum_{\sigma_{l+1},\ldots,\sigma_L} \sum_{a_{l+1},\ldots,a_{L-1}} M^{\sigma_{l+1}}_{a_l a_{l+1}} \ldots M^{\sigma_L}_{a_{L-1}} M^{*\sigma_{l+1}}_{a'_l a'_{l+1}} \ldots M^{*\sigma_L}_{a'_{L-1}} . \tag{2.5.17}$$

The reduced density matrix for region B (ρ_B) can be made using the same argument but it is the left hand sites that are contracted.

Like the SVD in eq. (2.4.4), the reduced density matrices can be used to calculate the von Neumann entropy

$$S_{A|B} = -\mathrm{Tr}\rho_A \log_2 \rho_A = -\mathrm{Tr}\rho_B \log_2 \rho_B . \tag{2.5.18}$$

The entanglement entropy is the same whether calculated using ρ_A or ρ_B as it is a measure of how entangled block A is to B. It is worth noting that to take the log of a matrix it is necessary to diagonalise it and can therefore be computationally expensive. When the MPS is in a mixed state with the left region left normalised and the right region right normalised the calculation of entanglement entropy is greatly simplified, ρ_A of eq. (2.5.17) becomes

$$\begin{aligned}\rho_A &= \sum_{\sigma_{l+1},\ldots,\sigma_L} \sum_{a_{l+1},\ldots,a_{L-1}} S_{a_l a_l} B^{\sigma_{l+1}}_{a_l a_{l+1}} \ldots B^{\sigma_L}_{a_{L-1}} S_{a'_l a'_l} B^{*\sigma_{l+1}}_{a'_l a'_{l+1}} \ldots B^{*\sigma_L}_{a'_{L-1}} \\ &= \sum_{a_l} S_{a_l a_l} S^*_{a'_l a_l} = S S^\dagger .\end{aligned} \tag{2.5.19}$$

Similarly $\rho_B = S^\dagger S$. As S is real and diagonal, $\rho_A = \rho_B = S^2$. Thus eq. (2.5.18) becomes equal to (2.4.4) with the eigenvalues of the reduced density matrix equal to the squares of the singular values ($\lambda_a = s_a^2$).

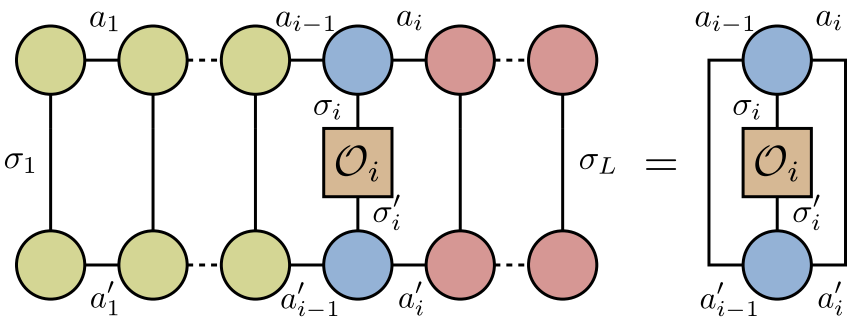

Figure 2.14: Expectation value for a single site operator.

2.5.4 Expectation Values

The expectation value of a local operator $\mathcal{O}^{[i]}$ can be calculated in a highly efficient manner due to the normalisation of the MPS, where the operator is defined as

$$\mathcal{O}^{[i]} = \sum_{\sigma_i,\sigma_i'} O^{\sigma_i,\sigma_i'} \left|\sigma_i\right\rangle \left\langle\sigma_i'\right| , \tag{2.5.20}$$

and implicitly acts with an identity on the rest of the sites. An example of such an operator is s_i^z, giving the expectation value for the z-component of the spin for site i. If the MPS bra and ket are gauged such that all sites to the left of the operator are left normalised and all to the right are right normalised, the calculation of the expectation value is almost trivial, i.e.

$$\begin{aligned}
\langle\Psi|\mathcal{O}^{[i]}|\Psi\rangle = &\sum_{\sigma_1,\ldots,\sigma_L} \sum_{\substack{a_1,\ldots,a_{L-1}\\ a_1',\ldots,a_{L-1}'}} A^{*\sigma_1}_{a_1'} \ldots A^{*\sigma_{i-1}}_{a_{i-2}'a_{i-1}'} M^{*\sigma_i'}_{a_{i-1}'a_i'} B^{*\sigma_{i+1}}_{a_i'a_{i+1}'} \ldots B^{*\sigma_L}_{a_{L-1}'} \\
&\times O^{\sigma_i\sigma_i'} A^{\sigma_1}_{a_1} \ldots A^{\sigma_{i-1}}_{a_{i-2}a_{i-1}} M^{\sigma_i}_{a_{i-1}a_i} B^{\sigma_{i+1}}_{a_ia_{i+1}} \ldots B^{\sigma_L}_{a_{L-1}} \\
&\times \langle\sigma_1\ldots\sigma_{i-1}|\sigma_1\ldots\sigma_{i-1}\rangle \langle\sigma_i'|\sigma_i'\rangle \langle\sigma_i|\sigma_i\rangle \langle\sigma_{i+1}\ldots\sigma_L|\sigma_{i+1}\ldots\sigma_L\rangle \\
= &\sum_{\sigma_i\sigma_i'} \sum_{a_{i-1}a_i} M^{*\sigma'}_{a_{i-1}a_i} O^{\sigma_i\sigma_i'} M^{\sigma_i}_{a_{i-1}a_i},
\end{aligned} \tag{2.5.21}$$

where the normalisation of the MPS was used to set the sites to the left and right of the operator equal to the identity. The diagrammatic representation is given in fig. 2.14. In a similar way to fig. 2.12 it can be shown that (2.5.21) can be performed in $O(\chi^2 d)$.

The expectation value of operators is not limited to one site. Local operators can be contracted onto as many sites as desired to create such expectations as two-

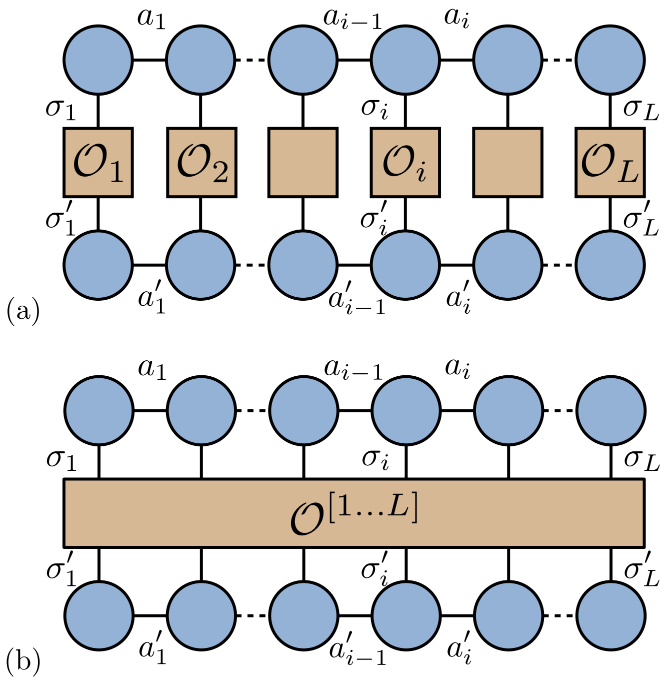

Figure 2.15: Expectation value for (a) a single site operator acting on all sites and (b) a general operator acting on all sites.

point correlation functions or the total s^z taking the form

$$\langle\Psi|\,\mathcal{O}^{[1]}\ldots\mathcal{O}^{[L]}\,|\Psi\rangle = \sum_{\substack{\sigma_1,\ldots\sigma_L\\ \sigma_1',\ldots,\sigma_L'}}\ \sum_{\substack{a_1,\ldots,a_{L-1}\\ a_1',\ldots,a_{L-1}'}} M^{*\sigma_1'}_{a_1'}\ldots M^{*\sigma_L'}_{a_{L-1}'}\,O^{\sigma_1\sigma_1'}\ldots O^{\sigma_L\sigma_L'}M^{\sigma_1}\ldots M^{\sigma_L'}, \tag{2.5.22}$$

or diagrammatically as shown by fig. 2.15(a). Gauging the MPS to a mixed canonical form will provide a simplification to the contraction if there are not operators acting on the edge sites. In eq. (2.5.22) there is no such simplification as all sites are acted on by operators, so is calculated in $O(L\chi^3 d)$. The most general form of expectation value is where the operator is not local and represented by a $2L$ index tensor,

$$\langle\Psi|\,\mathcal{O}^{[1\ldots L]}\,|\Psi\rangle = \sum_{\substack{\sigma_1,\ldots\sigma_L\\ \sigma_1',\ldots,\sigma_L'}}\ \sum_{\substack{a_1,\ldots,a_{L-1}\\ a_1',\ldots,a_{L-1}'}} M^{*\sigma_1'}_{a_1'}\ldots M^{*\sigma_L'}_{a_{L-1}'}\,O^{\sigma_1\sigma_1'\ldots\sigma_L\sigma_L'}M^{\sigma_1}\ldots M^{\sigma_L'}, \tag{2.5.23}$$

given by fig. 2.15(b). An example of an operator that takes this form is the Hamiltonian. Unlike the local operators, contracting full tensor operators means having the operator for all sites stored in memory, thus the memory and time requirement grows exponentially in system size as $O(d^L)$.

2.6 Matrix Product Operators

The computational cost of storage and manipulation of general tensor operators scales exponentially with system size as discussed in the previous section. The idea of a *matrix product operator* (MPO) [30, 46, 47] is to use the ideas of MPSs to transform the tensor operators into a form that can potentially be contracted efficiently. A general operator can be written as

$$\mathcal{O} = \sum_{\substack{\sigma_1,\dots\sigma_L \\ \sigma_1',\dots,\sigma_L'}} O^{\sigma_1\sigma_1'\dots\sigma_L\sigma_L'} \left|\sigma_1\dots\sigma_L\right\rangle \left\langle\sigma_1'\dots\sigma_L'\right| . \tag{2.6.1}$$

This can be decomposed using SVDs in the same way as the MPS derivation for (2.5.4). Proceeding as before with O' being the matrix resulting from a reshaping of tensor O, we have

$$\begin{aligned} O^{\sigma_1\sigma_1'\dots\sigma_L\sigma_L'} = O'_{\sigma_1\sigma_1',(\sigma_2\sigma_2'\dots\sigma_L\sigma_L')} &= \sum_{b_1} U^{[1]}_{\sigma_1\sigma_1',b_1} S^{[1]}_{b_1,b_1} V^{[1]\dagger}_{b_1,(\sigma_2\sigma_2'\dots\sigma_L\sigma_L')} \\ &= \sum_{b_1} W^{\sigma_1\sigma_1'}_{b_1} O'_{(\sigma_2\sigma_2',b_1),(\sigma_3\sigma_3'\dots\sigma_L\sigma_L')} \\ &= \sum_{b_1,b_2} W^{\sigma_1\sigma_1'}_{b_1} W^{\sigma_2\sigma_2'}_{b_1b_2} O'_{(\sigma_3\sigma_3',b_2),(\sigma_4\sigma_4'\dots\sigma_L\sigma_L')} \\ &= \sum_{b_1,\dots,b_{L-1}} W^{\sigma_1\sigma_1'}_{b_1} W^{\sigma_2\sigma_2'}_{b_1b_2} \dots W^{\sigma_{L-1}\sigma_{L-1}'}_{b_{L-2}b_{L-1}} W^{\sigma_L\sigma_L'}_{b_{L-1}} \end{aligned} \tag{2.6.2}$$

The standard form for an MPO is therefore

$$\mathcal{O} = \sum_{\substack{\sigma_1,\dots\sigma_L \\ \sigma_1',\dots,\sigma_L'}} \sum_{b_1,\dots,b_{L-1}} W^{\sigma_1\sigma_1'}_{b_1} W^{\sigma_2\sigma_2'}_{b_1b_2} \dots W^{\sigma_{L-1}\sigma_{L-1}'}_{b_{L-2}b_{L-1}} W^{\sigma_L\sigma_L'}_{b_{L-1}} \left|\sigma_1\dots\sigma_L\right\rangle \left\langle\sigma_1'\dots\sigma_L'\right| . \tag{2.6.3}$$

As usual this derivation can be performed pictorially as in fig. 2.16.

The expectation value $\langle\Psi|\mathcal{O}|\Psi\rangle$ is found by contracting the MPO with an MPS bra and ket

$$\begin{aligned} \langle\Psi|\mathcal{O}|\Psi\rangle = \sum_{\substack{\sigma_1,\dots,\sigma_L \\ \sigma_1',\dots,\sigma_L'}} \sum_{\substack{a_1,\dots,a_L \\ a_1',\dots,a_L'}} \sum_{b_1,\dots,b_L} & M^{*\sigma_1'}_{a_1'} W^{\sigma_1\sigma_1'}_{b_1} M^{\sigma_1}_{a_1} M^{*\sigma_2'}_{a_1'a_2'} W^{\sigma_2\sigma_2'}_{b_1b_2} M^{\sigma_2}_{a_1a_2} \times \dots \\ & \dots \times M^{*\sigma_{L-1}'}_{a_{L-2}'a_{L-1}'} W^{\sigma_{L-1}\sigma_{L-1}'}_{b_{L-2}b_{L-1}} M^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} M^{*\sigma_L'}_{a_{L-1}'} W^{\sigma_L\sigma_L'}_{b_{L-1}} M^{\sigma_L}_{a_{L-1}} . \end{aligned} \tag{2.6.4}$$

Equation (2.6.4) is a clear example of the power of diagrammatic MPS. With all of

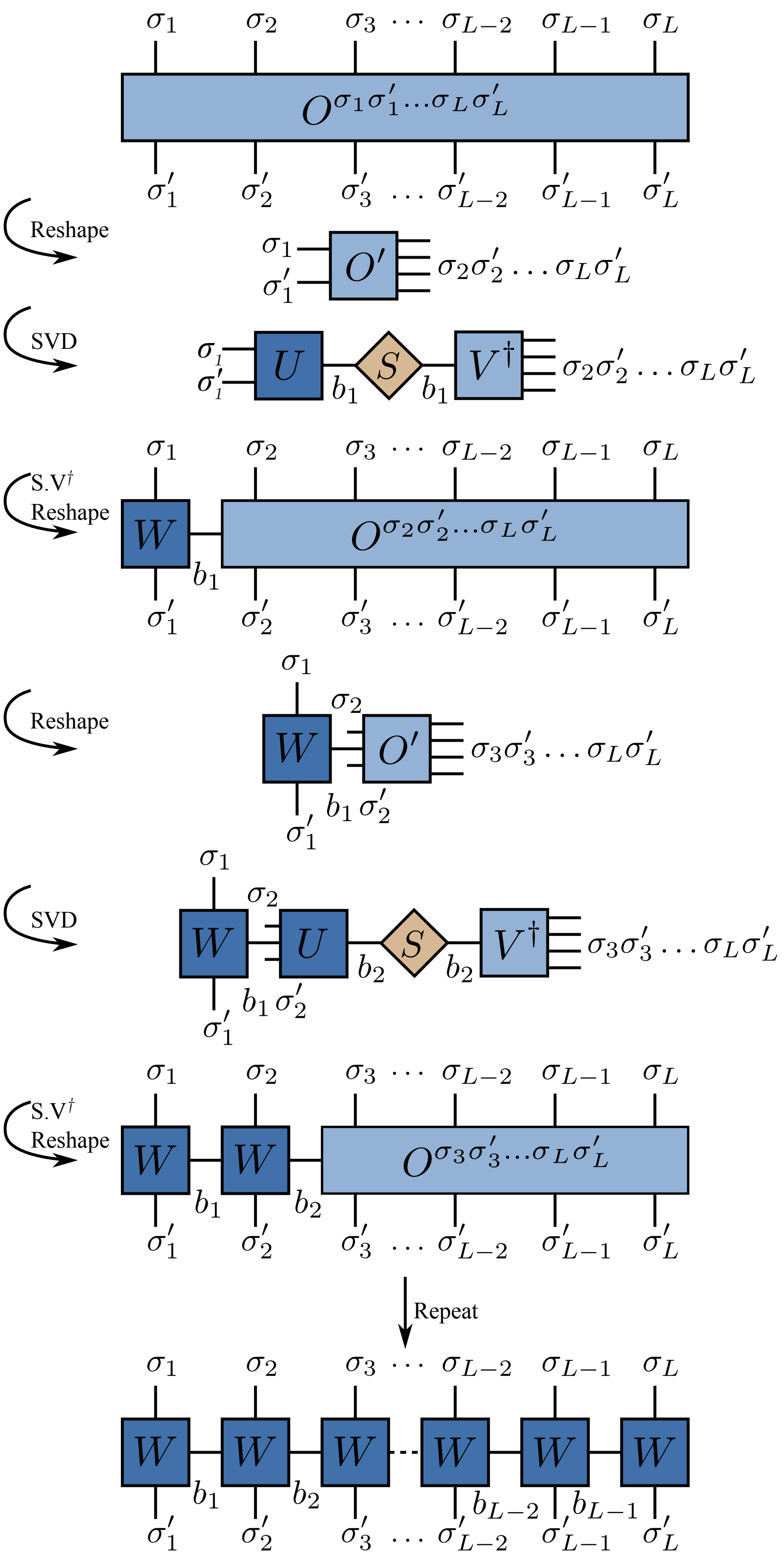
$O^{\sigma_1\sigma_1'\ldots\sigma_L\sigma_L'}$
σ_1 σ_2 σ_3 $\cdots$ σ_{L-2} σ_{L-1} σ_L
σ_1' σ_2' σ_3' $\ldots$ σ_{L-2}' σ_{L-1}' σ_L'
Reshape
O'
$\sigma_2\sigma_2'\ldots\sigma_L\sigma_L'$
SVD
U
S
$V^\dagger$
b_1
S.V†
Reshape
W
$O^{\sigma_2\sigma_2'\ldots\sigma_L\sigma_L'}$
$\sigma_3\sigma_3'\ldots\sigma_L\sigma_L'$
b_2
$O^{\sigma_3\sigma_3'\ldots\sigma_L\sigma_L'}$
Repeat
b_{L-2}
b_{L-1}

Figure 2.16: Diagrammatic derivation of the standard form of the MPO of eq. (2.6.3).

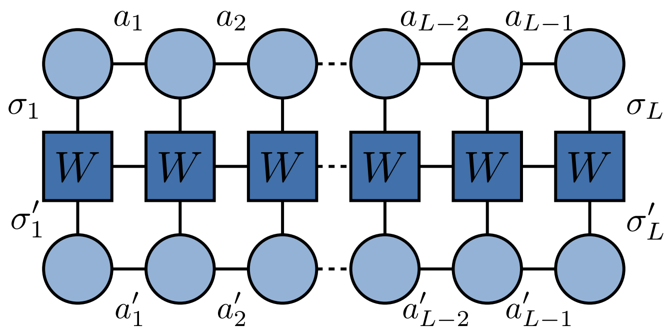

Figure 2.17: Diagrammatic form of the expectation value as given by eq. (2.6.4). As usual the circles are MPS tensors and the squares MPO tensors.

the indices explicitly written the expression is rather opaque yet the diagram in fig. 2.17 is easy to understand. Many operators have an MPO representation that has a constant MPO *bond dimension* D_W, that is the indices b_i have a constant number of elements. Most noticeably many common Hamiltonians have this property, for example the Heisenberg and Hubbard models, meaning that calculating energy expectation values becomes efficient taking $O(L\chi^3 D_W d)$.

2.6.1 Explicit form of Matrix Product Operators

Many of the most common lattice Hamiltonians can be written as an MPO with a constant bond dimension D_W, so it is useful to understand how these explicit MPOs are created and manipulated. For more complicated MPOs it is helpful to use graphical representations. Throughout this section *matrix product diagrams* will be used but so-called *finite weight automata* are also common [29]. Matrix product diagrams consist of two columns of points representing the first and second indices of a matrix, then a labelled arrow connecting a point of the first column to one on the second column is the element in the matrix for those indices. Matrix multiplication is then simply two of these placed end on end and the matrix elements are the sum of all the paths that connect the appropriate indices (see fig. 2.18 for an example).

The most simple one dimensional nearest-neighbour Hamiltonian with OBCs takes the form

$$H = \sum_{i=1}^{L-1} A_i A_{i+1}. \tag{2.6.5}$$

When written explicitly in terms of tensors this is [36]

$$
\begin{aligned}
H^{\sigma_1\sigma'_1\cdots\sigma_L\sigma'_L} = A_1^{\sigma_1\sigma'_1} \otimes A_2^{\sigma_2\sigma'_2} \otimes \mathbb{1}^{\sigma_3\sigma'_3} \otimes \cdots \otimes \mathbb{1}^{\sigma_i\sigma'_i} \otimes \cdots \otimes \mathbb{1}^{\sigma_L\sigma'_L} \\
+ \mathbb{1}^{\sigma_1\sigma'_1} \otimes A_2^{\sigma_2\sigma'_2} \otimes A_3^{\sigma_3\sigma'_3} \otimes \cdots \otimes \mathbb{1}^{\sigma_i\sigma'_i} \otimes \cdots \otimes \mathbb{1}^{\sigma_L\sigma'_L} + \dots \\
\dots + \mathbb{1}^{\sigma_1\sigma'_1} \otimes \cdots \otimes A_i^{\sigma_i\sigma'_i} \otimes A_{i+1}^{\sigma_{i+1}\sigma'_{i+1}} \otimes \cdots \otimes \mathbb{1}^{\sigma_L\sigma'_L} + \dots \\
\dots + \mathbb{1}^{\sigma_1\sigma'_1} \otimes \cdots \otimes \mathbb{1}^{\sigma_i\sigma'_i} \otimes \cdots \otimes A_{L-1}^{\sigma_{L-1}\sigma'_{L-1}} \otimes A_L^{\sigma_L\sigma'_L},
\end{aligned}
\tag{2.6.6}
$$

which can be written as an MPO with tensors

$$
W_{1b_1}^{[1]\sigma_1\sigma'_i} = \begin{pmatrix} \mathbb{1}^{\sigma_1\sigma'_1} & A_1^{\sigma_1\sigma'_1} & 0 \end{pmatrix}, \tag{2.6.7}
$$

$$
W_{b_{i-1}b_i}^{[i]\sigma_i\sigma'_i} = \begin{pmatrix} \mathbb{1}^{\sigma_i\sigma'_i} & A_i^{\sigma_i\sigma'_i} & 0 \\ 0 & 0 & A_i^{\sigma_i\sigma'_i} \\ 0 & 0 & \mathbb{1}^{\sigma_i\sigma'_i} \end{pmatrix}, \tag{2.6.8}
$$

$$
W_{b_{L-1}1}^{[L]\sigma_L\sigma'_L} = \begin{pmatrix} 0 \\ A_L^{\sigma_L\sigma'_L} \\ \mathbb{1}^{\sigma_L\sigma'_L} \end{pmatrix}. \tag{2.6.9}
$$

The matrix product diagrammatic form of eq. (2.6.8) is shown in fig. 2.18(b). The first and last sites are special cases, they are the first row and last column of the general case, respectively. These have the diagrammatic form shown in fig. 2.18(a) and 2.18(c), respectively. The product of multiple matrices is represented by concatenating multiple of these diagrams as shown in fig. 2.18(d). The resulting operator is the sum of the possible paths across the diagram where the binary operator between the A_i is a tensor product. This gives the explicit form of eq. 2.6.6 as desired.

On-site terms are simple to include in the MPO and have a clear representation in the diagram. They take the top right element of the MPO and connect the top to the bottom of the diagram directly on that site. An example of a Hamiltonian with on-site and nearest neighbour terms is

$$
H = \sum_{i=1}^{L-1} A_i A_{i+1} + \sum_{i=1}^{L} B_i, \tag{2.6.10}
$$

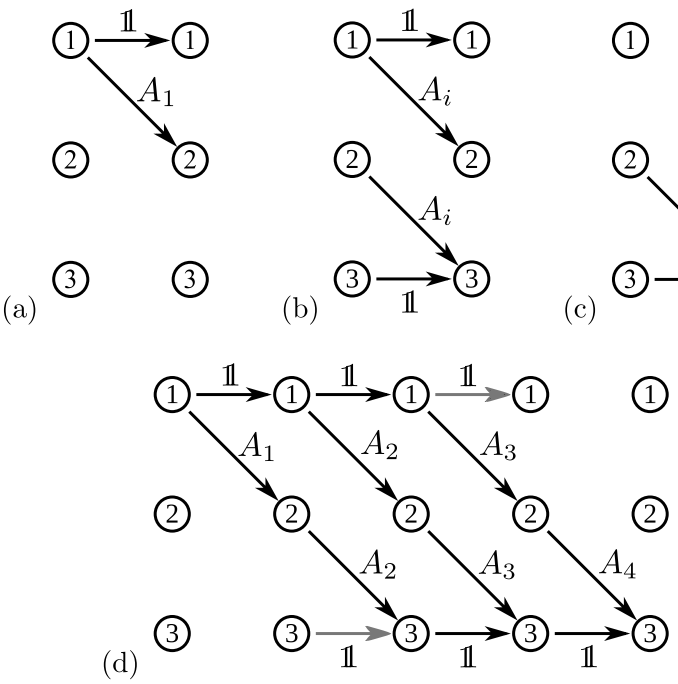

Figure 2.18: Matrix product diagrams for nearest neighbour Hamiltonian showing (a) bulk terms of eq. (2.6.8), (b) the left hand boundary site as given by eq. 2.6.7, (c) the right hand boundary site, given by eq. 2.6.9 and (d) four sites contracted. The circles represent the indices of the matrix, the left hand column gives the b_{i-1} index of W_{b_{i-1},b_i}, the right hand column gives the b_i index. The connecting paths give the value at that site in the matrix. The greyed out arrows in (d) denote elements that are present in the matrix for a site but are not used by any full path.

where B_i are the on site terms. This can be written as the following MPO,

$$W^{[1]}_{1,b_1} = \begin{pmatrix} \mathbb{1} & A_1 & B_1 \end{pmatrix}, \tag{2.6.11}$$

$$W^{[i]}_{b_{i-1},b_i} = \begin{pmatrix} \mathbb{1} & A_i & B_i \\ 0 & 0 & A_i \\ 0 & 0 & \mathbb{1} \end{pmatrix}, \tag{2.6.12}$$

$$W^{[L]}_{b_{L-1},1} = \begin{pmatrix} B_L \\ A_L \\ \mathbb{1} \end{pmatrix}, \tag{2.6.13}$$

where for clarity the σ_i indices have been omitted. The bulk terms have the diagrammatic form as shown in fig. 2.19.

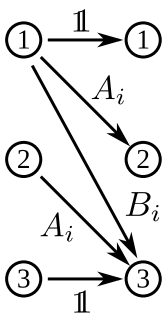

Figure 2.19: Matrix product diagram of a single site of an MPO Hamiltonian with nearest neighbour interactions A and on-site term B.

It is possible to use matrix product operators to produce representations of more elaborate operators with more than simple nearest neighbour interactions [36]. A Hamiltonian with next-to-nearest neighbour interactions would take the form

$$H = \sum_{i=1}^{L-2} A_i A_{i+2}, \tag{2.6.14}$$

which has the MPO

$$W^{[1]}_{1,b_1} = \begin{pmatrix} \mathbb{1} & A_1 & 0 & 0 \end{pmatrix}, \tag{2.6.15}$$

$$W^{[i]}_{b_{i-1},b_i} = \begin{pmatrix} \mathbb{1} & A_i & 0 & 0 \\ 0 & 0 & \mathbb{1} & 0 \\ 0 & 0 & 0 & A_i \\ 0 & 0 & 0 & \mathbb{1} \end{pmatrix}, \tag{2.6.16}$$

$$W^{[L]}_{b_{L-1},1} = \begin{pmatrix} 0 \\ 0 \\ A_L \\ \mathbb{1} \end{pmatrix}. \tag{2.6.17}$$

The matrix product diagram of a five site system is given in fig. 2.20 and shows clearly how the next-to-nearest neighbour terms in the MPO arise. This form can be generalised to interactions of any range by increasing the size of the virtual index further and including more identities in the middle. Therefore the size of the virtual indices increases with the range of the interaction.

The final example is an exponentially decaying Hamiltonian. The MPO representation is interesting because it is very compact, containing terms of all

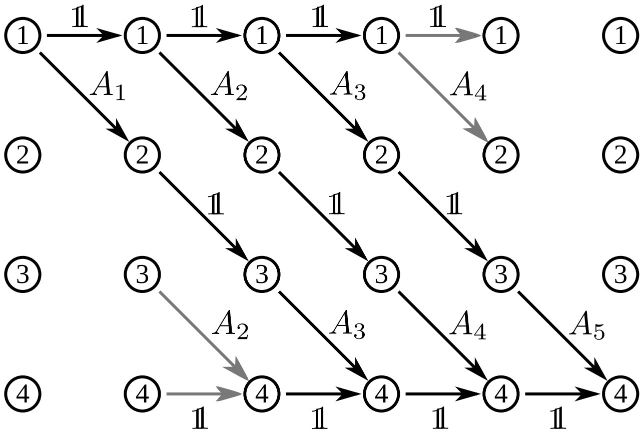

Figure 2.20: Matrix product diagram of a five site system with the next-to-nearest neighbour Hamiltonian MPO of eqs. (2.6.15, 2.6.16, 2.6.17).

orders without a growing MPO bond dimension. Such a Hamiltonian takes the form

$$H = \sum_{i=1}^{L-1} \sum_{q=1}^{L-i} \beta^q A_i A_{i+q}, \tag{2.6.18}$$

where β is a constant less than one for a decay and q gives the range of the interaction. The MPO for this Hamiltonian is [36]

$$W^{[1]}_{1,b_1} = \begin{pmatrix} \mathbb{1} & A_1 & 0 \end{pmatrix}, \tag{2.6.19}$$

$$W^{[i]}_{b_{i-1},b_i} = \begin{pmatrix} \mathbb{1} & A_i & 0 \\ 0 & \beta\mathbb{1} & \beta A_i \\ 0 & 0 & \mathbb{1} \end{pmatrix}, \tag{2.6.20}$$

$$W^{[L]}_{b_{L-1},1} = \begin{pmatrix} 0 \\ \beta A_L \\ \mathbb{1} \end{pmatrix}. \tag{2.6.21}$$

From fig. 2.21 it is clear that element $(2,2)$ of the MPO gives the exponential decay. This horizontal path allows interactions of arbitrary distance but contributes an extra factor of β for each increase in q, hence giving the desired factor of β^q.

The operators themselves that take the place of A and B in the discussion above depend on the model under consideration. Section 2.7 will concentrate on the

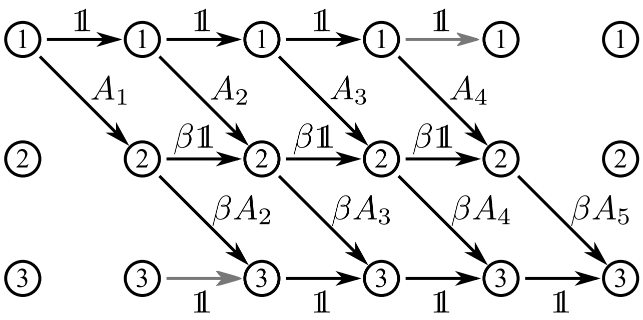

Figure 2.21: Matrix product diagram of a five site system with the exponentially decaying Hamiltonian MPO of eqs. (2.6.19, 2.6.20, 2.6.21).

spin-1/2 Heisenberg model with an external magnetic field h

$$H_{Heis} = J \sum_{i=1}^{L-1} \vec{s}_i \cdot \vec{s}_{i+1} - h \sum_{i=1}^{L} s_i^z, \tag{2.6.22}$$

where $\vec{s} = (s^x, s^y, s^z)$. This is a quantum spin model and as such the operators will be the Pauli spin operators

$$s^x = \frac{1}{2}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad s^y = \frac{1}{2}\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad s^z = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.6.23}$$

It is often convenient to create raising and lowering operators, which will be used in the following sections

$$s^+ = s^x + is^y = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \tag{2.6.24}$$

$$s^- = s^x - is^y = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}. \tag{2.6.25}$$

The spin-1/2 Hamiltonian has physical or *spin* indices that take two values. The explicit form of the W tensors is [48]

$$W_{1,b_1}^{\sigma_1,\sigma_1'} = \begin{pmatrix} [\mathbb{1}]^{\sigma_1,\sigma_1'} & \frac{J}{2}[s^+]^{\sigma_1,\sigma_1'} & \frac{J}{2}[s^-]^{\sigma_1,\sigma_1'} & J[s^z]^{\sigma_1,\sigma_1'} & -h[s^z]^{\sigma_1,\sigma_1'} \end{pmatrix}, \tag{2.6.26}$$

$$W_{b_{i-1},b_i}^{\sigma_i,\sigma_i'} = \begin{pmatrix} [\mathbb{1}]^{\sigma_i,\sigma_i'} & \frac{J}{2}[s^+]^{\sigma_i,\sigma_i'} & \frac{J}{2}[s^-]^{\sigma_i,\sigma_i'} & J[s^z]^{\sigma_i,\sigma_i'} & -h[s^z]^{\sigma_i,\sigma_i'} \\ 0 & 0 & 0 & 0 & [s^-]^{\sigma_i,\sigma_i'} \\ 0 & 0 & 0 & 0 & [s^+]^{\sigma_i,\sigma_i'} \\ 0 & 0 & 0 & 0 & [s^z]^{\sigma_i,\sigma_i'} \\ 0 & 0 & 0 & 0 & [\mathbb{1}]^{\sigma_i,\sigma_i'} \end{pmatrix}, \tag{2.6.27}$$

$$W_{b_{L-1},1}^{\sigma_L,\sigma_L'} = \begin{pmatrix} -h[s^z]^{\sigma_L,\sigma_L'} \\ [s^-]^{\sigma_L,\sigma_L'} \\ [s^+]^{\sigma_L,\sigma_L'} \\ [s^z]^{\sigma_L,\sigma_L'} \\ [\mathbb{1}]^{\sigma_L,\sigma_L'} \end{pmatrix}, \tag{2.6.28}$$

where “0” is understood as a zero matrix with indices σ_i and σ_i'. The matrix product diagrams for these operators are in fig. 2.22. Multiplying the matrices recovers the familiar Heisenberg Hamiltonian

$$H_{Heis} = J \sum_{i=1}^{L-1} \left[s_i^z s_{i+1}^z + \frac{1}{2} \left(s_i^+ s_{i-1}^- + s_i^- s_{i-1}^+ \right) \right] - h \sum_{i=1}^{L} s_i^z. \tag{2.6.29}$$

Many Hamiltonians are based on particle creation, $c^\dagger$, and annihilation, c, operators that act on each site. A common example is the Hubbard model [49]

$$H_{Hub} = -t \sum_{i=1}^{L-1} \sum_{\sigma=\uparrow,\downarrow} \left(c_{i,\sigma}^\dagger c_{i+1,\sigma} + c_{i+1,\sigma}^\dagger c_{i,\sigma} \right) + U \sum_{i=1}^{L} n_{i,\uparrow} n_{i,\downarrow}, \tag{2.6.30}$$

where $c_{i,\sigma}^\dagger$ and $c_{i,\sigma}$ create and annihilate particles of spin up or down at site i. The number operators $n_{i,\uparrow} = c_{i,\uparrow}^\dagger c_{i,\uparrow}$ and $n_{i,\downarrow} = c_{i,\downarrow}^\dagger c_{i,\downarrow}$ count the number of spin up and down particles at each site. Fermions obey anti-commutation relations

$$\begin{aligned} \left\{ c_{i,\sigma}^\dagger, c_{j,\tau} \right\} &= \delta_{i,j}\delta_{\sigma,\tau}, \\ \left\{ c_{i,\sigma}^\dagger, c_{j,\tau}^\dagger \right\} &= \{c_{i,\sigma}, c_{j,\tau}\} = 0, \end{aligned} \tag{2.6.31}$$

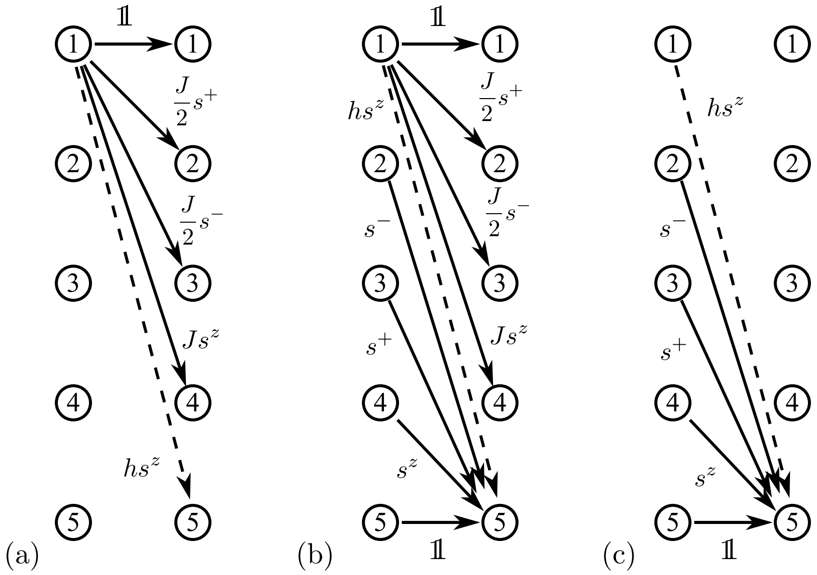

Figure 2.22: Matrix product diagram of the MPO for the full Heisenberg Hamiltonian for (a) site 1, (b) site i and (c) site L.

with matrix operators for the fermionic Hubbard model taking the form

$$c_{i,\uparrow} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad , \quad c_{i,\downarrow} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad , \quad n_{i,\uparrow}n_{i,\downarrow} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \tag{2.6.32}$$

To enforce the anti-commutation on multiple sites it is necessary to introduce a matrix P that contains information about the number of fermions to the left of the active site [50, 51]

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} . \tag{2.6.33}$$

The matrix gives a phase -1 when the site is occupied by one fermion, and $+1$ when occupied by zero or two. When the operators act on one of multiple sites they implicitly act on an extended basis, for a two site example

$$c_{1,\sigma} \rightarrow (c_{1,\sigma} \otimes \mathbb{1}). \tag{2.6.34}$$

The P matrix then acts on the site to the left

$$c_{2,\sigma} \to (P \otimes c_{2,\sigma}). \tag{2.6.35}$$

Using the identity

$$(A \otimes B)(C \otimes D) = AC \otimes BD, \tag{2.6.36}$$

P can be introduced into the Hamiltonian by multiplication by the *left* of each pair of operators. For example

$$\begin{aligned}(c^\dagger_{1,\sigma} \otimes \mathbb{1})(P \otimes c_{2,\sigma}) &= c^\dagger_{1,\sigma}P \otimes \mathbb{1}c_{2,\sigma} \\ (P \otimes c^\dagger_{2,\sigma})(c_{1,\sigma} \otimes \mathbb{1}) &= Pc_{1,\sigma} \otimes c^\dagger_{2,\sigma}\mathbb{1}.\end{aligned} \tag{2.6.37}$$

The MPO tensors will therefore be

$$W^{[1]}_{1,b_1} = \begin{pmatrix} \mathbb{1} & -tc^\dagger_{1,\uparrow}P & -tPc_{1,\uparrow} & -tc^\dagger_{1,\downarrow}P & -tPc_{1,\downarrow} & Un_{1,\uparrow}n_{1,\downarrow} \end{pmatrix}, \tag{2.6.38}$$

$$W^{[i]}_{b_{i-1},b_i} = \begin{pmatrix} \mathbb{1} & -tc^\dagger_{i,\uparrow}P & -tPc_{i,\uparrow} & -tc^\dagger_{i,\downarrow}P & -tPc_{i,\downarrow} & Un_{i,\uparrow}n_{i,\downarrow} \\ 0 & 0 & 0 & 0 & 0 & c_{i,\uparrow} \\ 0 & 0 & 0 & 0 & 0 & c^\dagger_{i,\uparrow} \\ 0 & 0 & 0 & 0 & 0 & c_{i,\downarrow} \\ 0 & 0 & 0 & 0 & 0 & c^\dagger_{i,\downarrow} \\ 0 & 0 & 0 & 0 & 0 & \mathbb{1} \end{pmatrix}, \tag{2.6.39}$$

$$W^{[L]}_{b_{L-1},1} = \begin{pmatrix} Un_{L,\uparrow}n_{L,\downarrow} \\ c_{L,\uparrow} \\ c^\dagger_{L,\uparrow} \\ c_{L,\downarrow} \\ c^\dagger_{L,\downarrow} \\ \mathbb{1} \end{pmatrix}. \tag{2.6.40}$$

2.7 Finite Spin-1/2 Heisenberg DMRG using MPS

When building a DMRG algorithm there are many indices to take care of, therefore before starting it is useful to define an indexing convention. Throughout this discussion the convention will be to label physical indices first, top to bottom of the diagram, then virtual indices, also top to bottom. The indices relating to bras will be primed and kets unprimed. Explicitly this takes the form

$$\left[\sigma_i, \sigma'_i, a_{i-1}, b_{i-1}, a'_{i-1}, a_i, b_i, a'_i\right], \tag{2.7.1}$$

where the a are the virtual indices for the MPS, the b are the virtual indices for the MPO and the σ are the physical indices.

The starting point for the DMRG algorithm is an MPS

$$|\Psi\rangle = \sum_{\sigma_1,\ldots,\sigma_L} \sum_{a_1,\ldots,a_{L-1}} M^{\sigma_1}_{1,a_1} M^{\sigma_2}_{a_1 a_2} \ldots M^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} M^{\sigma_L}_{a_{L-1},1} |\sigma_1, \ldots, \sigma_L\rangle \, , \tag{2.7.2}$$

where the M are understood as unnormalised MPS tensors with random elements to begin with, although the MPS form of a known state can also be used. In terms of the bond indices the first site is a row vector with a placeholder for the column index, similarly the final site is a column vector. The maximal size of the tensors in the form $[\sigma_i, a_{i-1}, a_i]$ is

$$(d,1,d),(d,d,d^2),\ldots,(d,d^{\frac{L}{2}-1},d^{\frac{L}{2}}),(d,d^{\frac{L}{2}},d^{\frac{L}{2}-1}),\ldots,(d,d^2,d),(d,d,1) \tag{2.7.3}$$

where d is the spin dimension. In the case of spin-1/2 Heisenberg $d = 2$, therefore the tensors have size

$$(2,1,2),(2,2,4),(2,4,8),\ldots,(2,8,4),(2,4,2),(2,2,1). \tag{2.7.4}$$

When performing DMRG we do not want to keep all of the information for all states in the MPS, only the most relevant information for calculating the ground state. In standard DMRG, a density matrix is formed from the ground state, which is diagonalised and the highest m states are used to truncate the basis. With MPS the truncation is performed by limiting the size of the bond dimension χ. The standard general method for doing so is to perform an SVD, keep the largest χ singular values and resize U and V appropriately.

The DMRG sweep [30] will start at the left hand side of the chain which requires a right-normalised MPS. The normalisation is performed iteratively using SVDs in a manner reminiscent of the derivation of the canonical form (2.5.6). Starting with the MPS of eq. (2.7.2), $M^{\sigma_L}_{a_{L-1},1}$ is reshaped into a matrix by combing the σ_L index with the placeholder 1

$$|\Psi\rangle = \sum_{\sigma_1,\ldots,\sigma_L} \sum_{a_1,\ldots,a_{L-1}} M^{\sigma_1}_{1,a_1} M^{\sigma_2}_{a_1 a_2} \ldots M^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} M_{a_{L-1},(\sigma_L,1)} |\sigma_1, \ldots, \sigma_L\rangle \, . \tag{2.7.5}$$

An SVD is then performed, the $V^\dagger$ is reshaped into the right normalised B tensor whilst the U and S are combined with the M tensor to the left

$$
\begin{aligned}
|\Psi\rangle = &\sum_{\sigma_1\ldots,\sigma_L}\sum_{a_1,\ldots,a_{L-1}} M^{\sigma_1}_{1,a_1} M^{\sigma_2}_{a_1a_2}\ldots M^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} \\
&\qquad\qquad \times U^{[L]}_{a_{L-1},a_{L-1}} S^{[L]}_{a_{L-1},a_{L-1}} V^{\dagger[L]}_{a_{L-1},(\sigma_L,1)} |\sigma_1,\ldots,\sigma_L\rangle \\
|\Psi\rangle = &\sum_{\sigma_1,\ldots,\sigma_L}\sum_{a_1,\ldots,a_{L-2}} M^{\sigma_1}_{1,a_1} M^{\sigma_2}_{a_1a_2}\ldots \tilde{M}^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} B^{\sigma_L}_{a_{L-1},1} |\sigma_1,\ldots,\sigma_L\rangle\,, \qquad (2.7.6)
\end{aligned}
$$

where $\tilde{M}$ is the contraction of M, U and S

$$
\tilde{M}^{\sigma_{L-1}}_{a_{L-2},a_{L-1}} = \sum_{a_{L-1}} M^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} U^{[L]}_{a_{L-1},a_{L-1}} S^{[L]}_{a_{L-1},a_{L-1}}. \qquad (2.7.7)
$$

This is then repeated for each site in order $L-1 \rightarrow 2$. Note that in general the Ms are three-index tensors which are reshaped into a matrix by combining the σ_i and a_i indices. The result is

$$
|\Psi\rangle = \sum_{\sigma_1,\ldots,\sigma_L}\sum_{a_1,\ldots,a_{L-1}} \tilde{M}^{\sigma_1}_{1,a_1} B^{\sigma_2}_{a_1a_2}\ldots B^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} B^{\sigma_L}_{a_{L-1},1} |\sigma_1,\ldots,\sigma_L\rangle\,. \qquad (2.7.8)
$$

The last M is reshaped, decomposed by SVD and $V^\dagger$ reshaped to B as before but now the U and S are just scalars which can be combined to a number $\sqrt{N}$, where N is simply the norm of the state

$$
\begin{aligned}
|\Psi\rangle &= \sum_{\sigma_1,\ldots,\sigma_L}\sum_{a_1,\ldots,a_{L-1}} U^{[1]}_{1,1} D^{[1]}_{1,1} B^{\sigma_1}_{1,a_1} B^{\sigma_2}_{a_1a_2}\ldots B^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} B_{a_{L-1},(\sigma_L,1)} |\sigma_1,\ldots,\sigma_L\rangle \\
&= \sum_{\sigma_1,\ldots,\sigma_L}\sum_{a_1,\ldots,a_{L-1}} \sqrt{N} B^{\sigma_1}_{1,a_1} B^{\sigma_2}_{a_1a_2}\ldots B^{\sigma_{L-1}}_{a_{L-2}a_{L-1}} B^{\sigma_L}_{a_{L-1},1} |\sigma_1,\ldots,\sigma_L\rangle\,. \qquad (2.7.9)
\end{aligned}
$$

If the state is to be normalised then N is set to 1. Thus state is fully right-normalised as in eq. (2.5.7). The diagrammatic representation of this normalisation is in fig. 2.23.

The next step is to include a Hamiltonian MPO for the Heisenberg model given by eq. (2.6.27). At the current point, as the MPS was constructed using random tensors, $\langle\Psi|H|\Psi\rangle$ gives the energy of a random MPS wavefunction. The DMRG algorithm is a variational approach that builds a set of MPS tensors iteratively whose energy expectation value converges to the ground state. The variational MPS algorithm is essentially the same as the finite *White style* DMRG in that the wavefunction is calculated over sweeps of the system where the accuracy increases with every step.

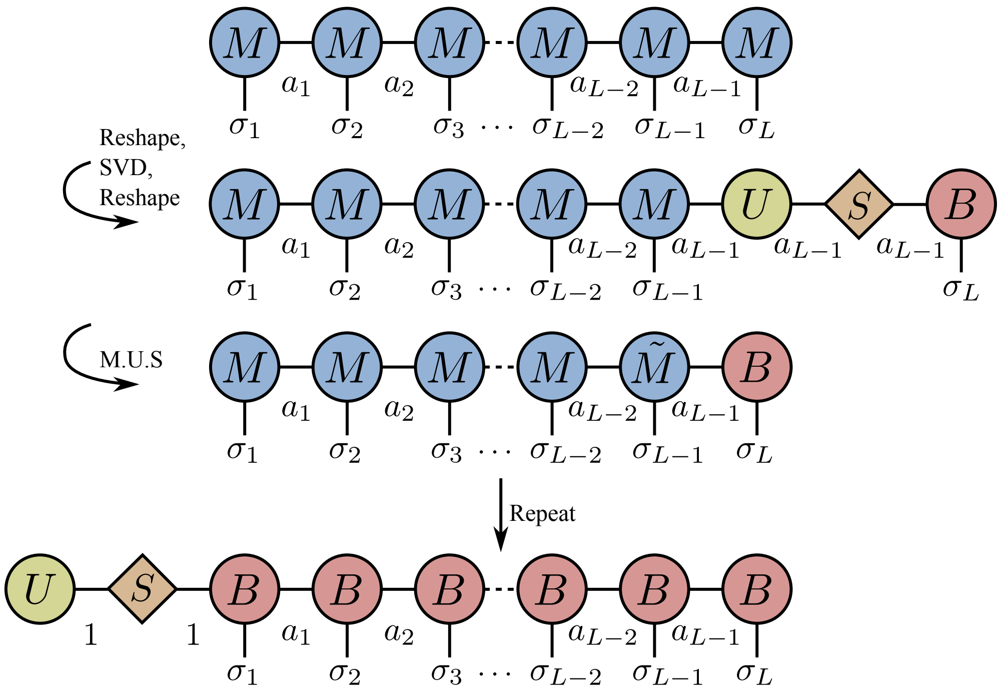

Figure 2.23: Procedure to right normalise an MPS as in eq. (2.7.9).

To perform an update to a tensor it is necessary to have the left and right *blocks* either side of the active site (l). The left block $L^{[l]}$ contains all of the information for the sites to the left of the active site (sites $1 \to l-1$) with each MPS tensor left normalised

$$L^{[l]}_{a_{l-1},b_{l-1},a'_{l-1}} = \sum_{\substack{\sigma_1,\dots,\sigma_{l-1}\\ \sigma'_1,\dots,\sigma'_{l-1}}} \sum_{\substack{a_1,\dots,a_{l-2}\\ a'_1,\dots,a'_{l-2}}} \sum_{b_1,\dots,b_{l-2}} \left(A^{*\sigma'_1}_{1,a'_1} W^{\sigma_1\sigma'_1}_{1,b_1} A^{\sigma_1}_{1,a_1}\right) \times \dots$$

$$\dots \times \left(A^{*\sigma'_{l-1}}_{a'_{l-2}a'_{l-1}} W^{\sigma_{l-1}\sigma'_{l-1}}_{b_{l-2}b_{l-1}} A^{\sigma_{l-1}}_{a_{l-2}a_{l-1}}\right). \qquad (2.7.10)$$

Similarly the right block contains the information for the sites to the right of the active site (sites $l+1 \to L$) with each MPS tensor right normalised

$$R^{[l]}_{a_l,b_l,a'_l} = \sum_{\substack{\sigma_{l+1},\dots,\sigma_L\\ \sigma'_{l+1},\dots,\sigma'_L}} \sum_{\substack{a_{l+1},\dots,a_{L-1}\\ a'_{l+1},\dots,a'_{L-1}}} \sum_{b_{l+1},\dots,b_{L-1}} \left(B^{*\sigma'_{l+1}}_{a'_l a'_{l+1}} W^{\sigma_{l+1}\sigma'_{l+1}}_{b_l b_{l+1}} B^{\sigma_{l+1}}_{a_l a_{l+1}}\right) \times \dots$$

$$\dots \times \left(B^{*\sigma'_L}_{a'_{L-1},1} W^{\sigma_L\sigma'_L}_{b_{L-1},1} B^{\sigma_L}_{a_{L-1},1}\right). \qquad (2.7.11)$$

The diagrams of the left and right block are shown in figs. 2.24(a+b). As the first

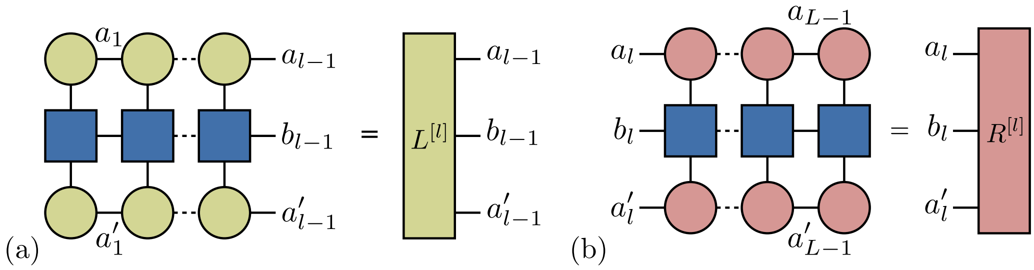

Figure 2.24: Diagram showing the construction of the (a) left block of eq. (2.7.10) and (b) right block of eq. (2.7.11).

active site is on the left hand boundary the left block is non-existent, but a dummy block $L^{[0]}_{1,1,1} = 1$ is used so that the process can be iterated more effectively. The right block is the contraction of all 2 → L of the remainder of the chain. It is very costly to contract the whole block for every step as in eq. (2.7.11), it is more efficient to build it iteratively starting with site L. By contracting site by site, the right blocks for the whole chain are built as the intermediate blocks for sites $L-1 \rightarrow 2$ can be saved. Each step in the iteration is calculated using the block to the right of it

$$R^{[i]}_{a_i,b_i,a'_i} = \sum_{\sigma_{i+1},\sigma'_{i+1},a_{i+1},b_{i+1},a'_{i+1}} B^{*\sigma'_{l+1}}_{a'_l a'_{l+1}} W^{\sigma_{l+1}\sigma'_{l+1}}_{b_l b_{l+1}} B^{\sigma_{l+1}}_{a_l a_{l+1}} R^{[i+1]}_{a_{i+1},b_{i+1},a'_{i+1}}. \tag{2.7.12}$$

As with the left block at $i = 1$, the right block at $i = L$ is a scalar, $R^{[L+1]}_{1,1,1} = 1$. Contraction in this way is therefore $O(\chi^3 D_W d)$.

An effective Hamiltonian tensor is built from the left and right blocks along with the W tensor for the site

$$H^{[l]\sigma_l,\sigma'_l}_{a_{l-1},a'_{l-1},a_l,a'_l} = \sum_{b_{l-1},b_l} L^{[l]}_{a_{l-1},b_{l-1},a'_{l-1}} W^{\sigma_l\sigma'_l}_{b_{l-1}b_l} R^{[l]}_{a_l,b_l a'_l}, \tag{2.7.13}$$

which is shown in fig. 2.25(a). This can be reshaped into a Hamiltonian matrix $H^{[l]}_{(\sigma_l,a_{l-1},a_l),(\sigma'_l,a'_{l-1},a'_l)}$ and can be used in the eigenvalue equation

$$H^{[l]}\psi = E\psi, \tag{2.7.14}$$

where E is the effective energy for a given MPS tensor ψ. The effective Hamiltonian $H^{[l]}$ can be diagonalised as a matrix, but in this form it is size $O(\chi^4 d^2)$. Retaining the tensor structure allows the application of MPS trial tensor (ψ) to the effective Hamiltonian in $O(\chi^3 D_W d)$ as shown in fig. 2.25(b). By setting this combination

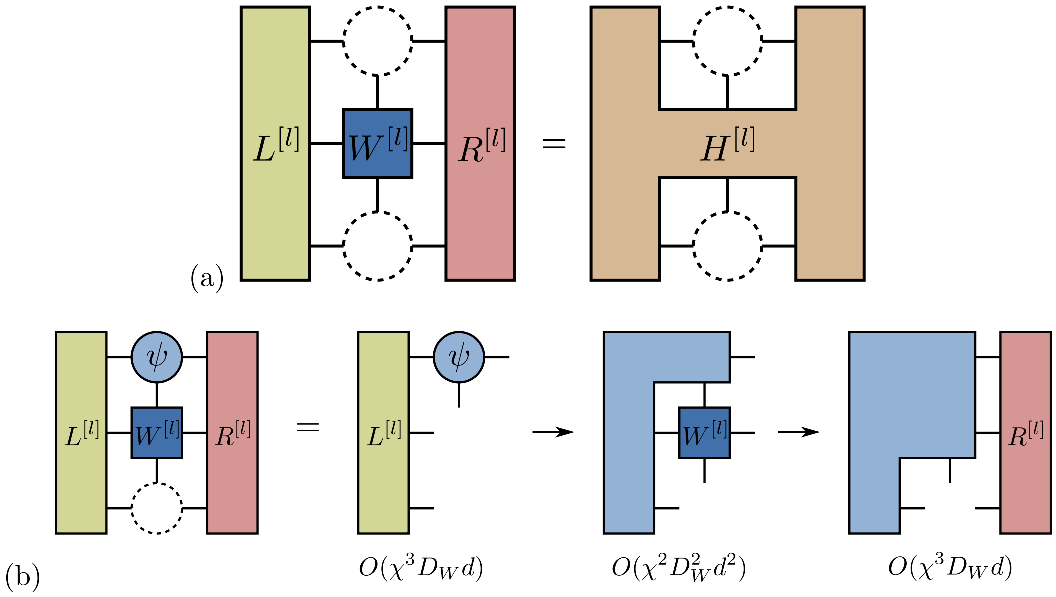

Figure 2.25: (a) Diagrammatic form of the effective Hamiltonian as given by eq. (2.7.13) (b) Cost of application of a trial MPS tensor ψ to the effective Hamiltonian.

as the new trial and repeating the application to the effective Hamiltonian it is possible to obtain the MPS tensor that minimises the energy. This is known as the *power method* of diagonalising a matrix. In practice the eigenvalue problem is solved using a sparse matrix Lanczos or Jacobi-Davidson algorithm where the eigenvector corresponding to the lowest eigenvalue is reshaped to the unnormalised M tensor for site l. This is then left-normalised by SVD with the S and V matrices contracted with $B^{\sigma_{l+1}}$, in the opposite way to the right normalisation of eq. (2.7.7).

The active site is then moved one to the right and the process is repeated to get the M tensor for the new site. Because of the way the right block was built, the right block for this site is already saved so it is only necessary to build the new left block before building the Hamiltonian. This is repeated until the end of the chain is reached, at which point the direction is changed and the new active site is one to the left. This moving from one end of the other is called *sweeping* and is performed until the energy has converged within a preset threshold or a maximum number of sweeps has been performed. An outline of the DMRG algorithm as described in this section is provided in fig. 2.26.

As currently described, the DMRG algorithm works, but has a tendency to get stuck in local minima rather than the global minimum for the system as desired.

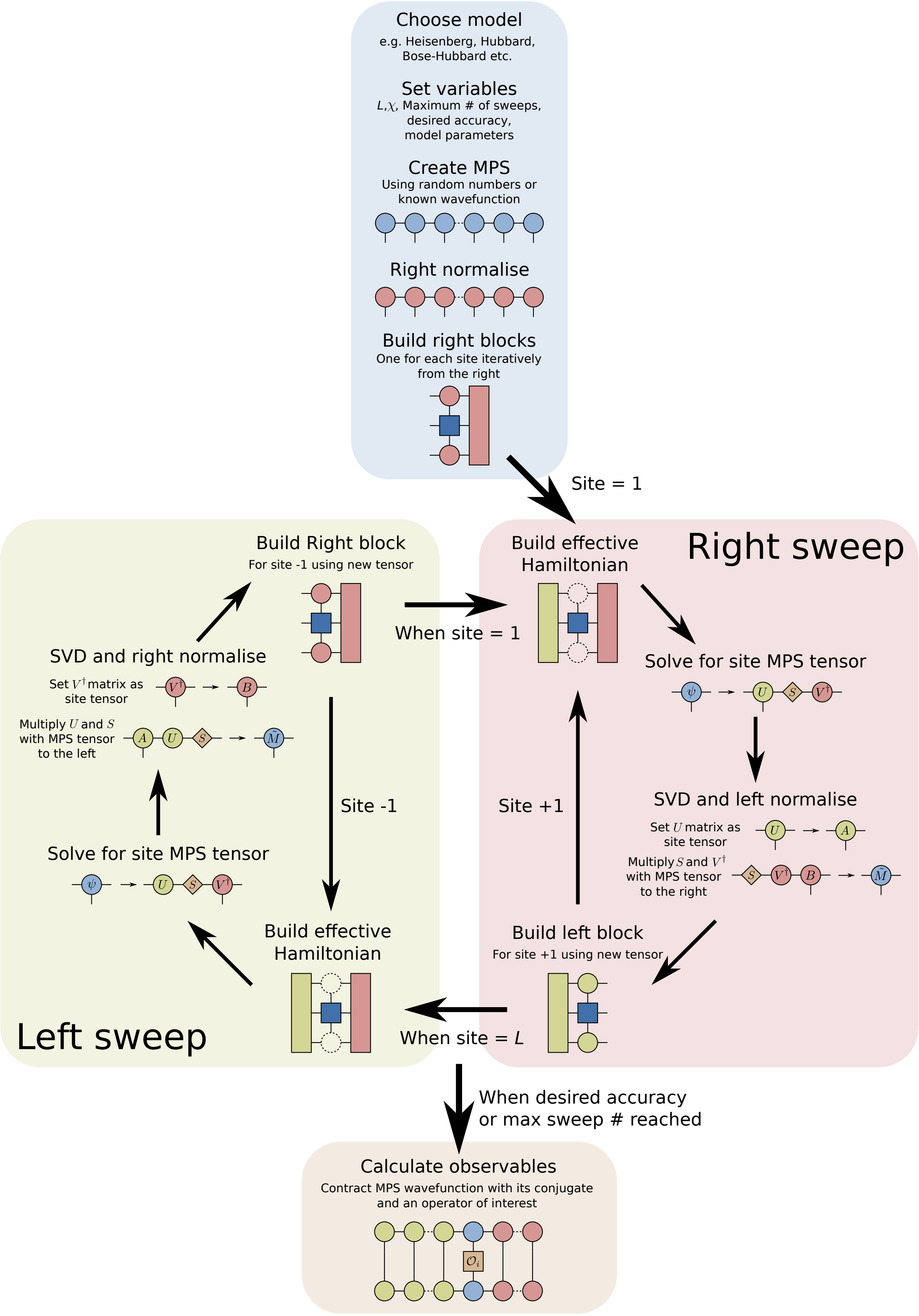
Choose model
e.g. Heisenberg, Hubbard, Bose-Hubbard etc.
Set variables
L,χ, Maximum # of sweeps, desired accuracy, model parameters
Create MPS
Using random numbers or known wavefunction
Right normalise
Build right blocks
One for each site iteratively from the right
Site = 1
Right sweep
Build effective Hamiltonian
Solve for site MPS tensor
ψ → U S $V^\dagger$
SVD and left normalise
Set U matrix as site tensor
U → A
Multiply S and $V^\dagger$ with MPS tensor to the right
S $V^\dagger$ B → $\tilde{M}$
Build left block
For site +1 using new tensor
Site +1
When site = L
Left sweep
Build effective Hamiltonian
Solve for site MPS tensor
ψ → U S $V^\dagger$
SVD and right normalise
Set $V^\dagger$ matrix as site tensor
$V^\dagger$ → B
Multiply U and S with MPS tensor to the left
A U S → $\tilde{M}$
Build Right block
For site -1 using new tensor
Site -1
When site = 1
When desired accuracy or max sweep # reached
Calculate observables
Contract MPS wavefunction with its conjugate and an operator of interest
$\mathcal{O}_i$

Figure 2.26: Overview of the DMRG algorithm described in detail in the text. The blue box signifies the setup stage, the pink box is the right sweep, the green box is the left sweep and the brown box is the output of operator expectation values.

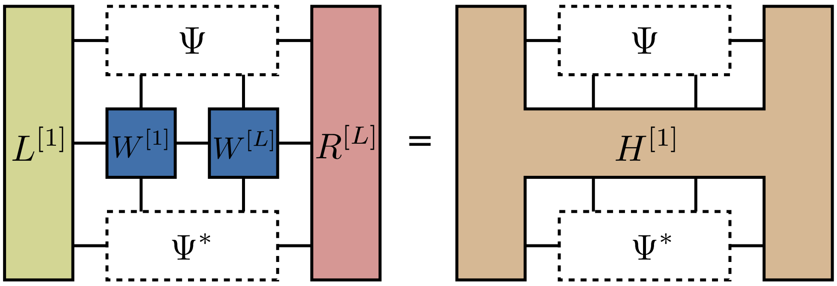

Figure 2.27: First step of the *infinite* DMRG algorithm. The effective Hamiltonian is made up of the MPO tensors for sites 1 and L.

The reason for this is that with the random starting point there is no guarantee that the global minimum will be found at each step, particularly when using *one-site* DMRG. There are various ways round this problem, two of which will be discussed in the subsequent sections; *two-site* DMRG and a *modified density-matrix* approach, both of which enlarge the *ansatz space* by coupling the active site to nearby sites, hopefully letting the global minimum dominate over the local.

2.7.1 Two-Site DMRG

Two site DMRG is in many ways a direct translation of the original DMRG algorithm [15] discussed in section 2.1, to MPS in that one considers two sites between the left and right blocks at each step of the algorithm. Although not always necessary, the two site algorithm also allows an infinite warm-up to get a first prediction before the sweeping begins, which was traditionally the method of creating a system of the desired size. This first prediction will be less likely to lead to the algorithm getting stuck in local minima and should allow convergence to be reached in fewer sweeps.

The following discussion of the infinite DMRG algorithm follows ref. [30] but will almost entirely use diagrammatic algebra as it is more transparent and easier to translate to computer algorithms. That said, every diagram can be written explicitly as formulae if desired. The starting point is two lone active sites, 1 and L with Hamiltonian MPO tensors for each site $W^{[1]}$ and $W^{[L]}$. The Hamiltonian matrix is built from these MPO tensors and dummy left and right blocks as used previously, as shown in fig. 2.27. This is diagonalised, the lowest energy eigenvector reshaped to become Ψ. An SVD is performed on Ψ, U is reshaped and stored as A^{σ_1}, $V^\dagger$ as B^{σ_L}, shown in fig. 2.28(a), remembering that the left virtual index of site 1 and the right virtual index of site L are dummy. Blocks $L^{[2]}$ and $R^{[L-1]}$ are built from the new A and B tensors and appropriate MPO tensors. The matrix of

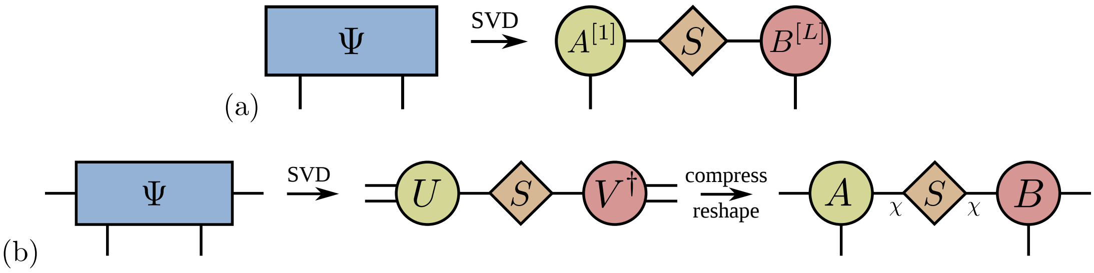

Figure 2.28: SVD of the lowest energy eigenvector after diagonalisation (a) for the first iteration creating the boundary tensors and (b) subsequent iterations where tensor dimensions are truncated χ.

singular values S is discarded to preserve normalisation before two more sites are introduced; 2 and $L-1$.

The remainder of the iterations of the infinite algorithm are performed in a similar manner to the first. Insert two sites, build Hamiltonian matrix, then diagonalise with the lowest energy eigenvector reshaped into Ψ. An SVD is performed as before but only the χ left-singular vectors of U with largest singular values are reshaped to become A^{σ_i} and similarly the χ right-singular vectors of $V^\dagger$ are reshaped to $B^{\sigma_{L+1-i}}$, shown in fig. 2.28(b). This is known as an SVD compression, which truncates the size of the MPS tensors, setting the bond dimension to χ. After the final iteration the last matrix of singular values is set to an identity for normalisation resulting in a MPS state of with bond dimension χ.

The finite algorithm follows the same idea as the single site algorithm, but this time the pair of active sites moves along the chain. The infinite algorithm leaves the MPS in *mixed-canonical form*, with the left hand side of the chain made up of left-normalised tensors (A) and the right hand side of the chain made up of right-normalised tensors (B), thus it makes sense to start the sweeping in the middle of the chain with $i = (L/2) + 1$.

The effective Hamiltonian is built up as before from $L^{[i]}$, $R^{[i+1]}$, $W^{[i]}$ and $W^{[i+1]}$. It is then diagonalised, to obtain Ψ, upon which an SVD is performed. The U is reshaped to A^{σ_i}, the D and $V^\dagger$ are then contracted with $B^{\sigma_{i+1}}$ and reshaped to provide a prediction P for the diagonalisation of the next step, shown in fig. 2.29(a). This is repeated until $i = L-1$ at which point the sweep has reached the right hand end of the chain. The same sequence is performed on the left sweep except after diagonalisation and SVD $V^\dagger$ is reshaped to $B^{\sigma_{i+1}}$ and $A^{\sigma_{i-1}}$ contracted with U and D to create the prediction for the next site (fig. 2.29(b)). This is repeated until $i = L-1$ at which point the sweep has reached the right hand end of the chain.

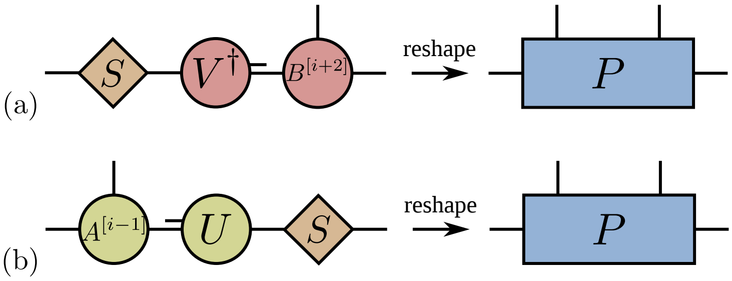

Figure 2.29: Prediction P for the next diagonalisation in (a) a right sweep (b) a left sweep.

These left and right sweeps are repeated until convergence is reached.

2.7.2 Modified Density Matrix

The modified density matrix approach was introduced by White [52] to tackle the issue of single site DMRG getting stuck in local minima. The issue arises because the environment block is incomplete so there are certain components of the wavefunction that span the two blocks that should be present but are not. The procedure introduces a modified density matrix to couple the active site to the system block in order to include these missing parts of the wavefunction that may be missed by a single site density matrix.

The notion of a density matrix has been largely omitted in the DMRG algorithms discussed so far because the SVD truncation performs the same task as a density-matrix truncation of traditional DMRG. The density matrix for an MPS tensor after diagonalisation $\psi^{\sigma_l}_{a_{l-1}a_l}$ is

$$\rho_{(\sigma_l a_{l-1} a_l),(\sigma'_l a'_{l-1} a'_l)} = |\psi\rangle\langle\psi| = \psi^{\sigma_l}_{a_{l-1}a_l}\psi^{\dagger\sigma'_l}_{a'_{l-1}a'_l}, \tag{2.7.15}$$

shown diagrammatically in fig. 2.30(a). There are two types of reduced density matrix depending on which part of the system is traced over

$$\rho_{A(\sigma_l a_{l-1}),(\sigma'_l a'_{l-1})} = \mathrm{Tr}_B |\psi\rangle\langle\psi| = \sum_{a_l}\psi^{\sigma_l}_{a_{l-1}a_l}\psi^{\dagger\sigma'_l}_{a'_{l-1}a'_l}, \tag{2.7.16}$$

$$\rho_{B(\sigma_l a_l),(\sigma'_l a'_l)} = \mathrm{Tr}_A |\psi\rangle\langle\psi| = \sum_{a_{l-1}}\psi^{\sigma_l}_{a_{l-1}a_l}\psi^{\dagger\sigma'_l}_{a'_{l-1}a'_l}, \tag{2.7.17}$$

shown in fig. 2.30(b+c). In traditional DMRG it is these density matrices that are diagonalised (although two site versions), ρ_A for a right sweep and ρ_B for a left

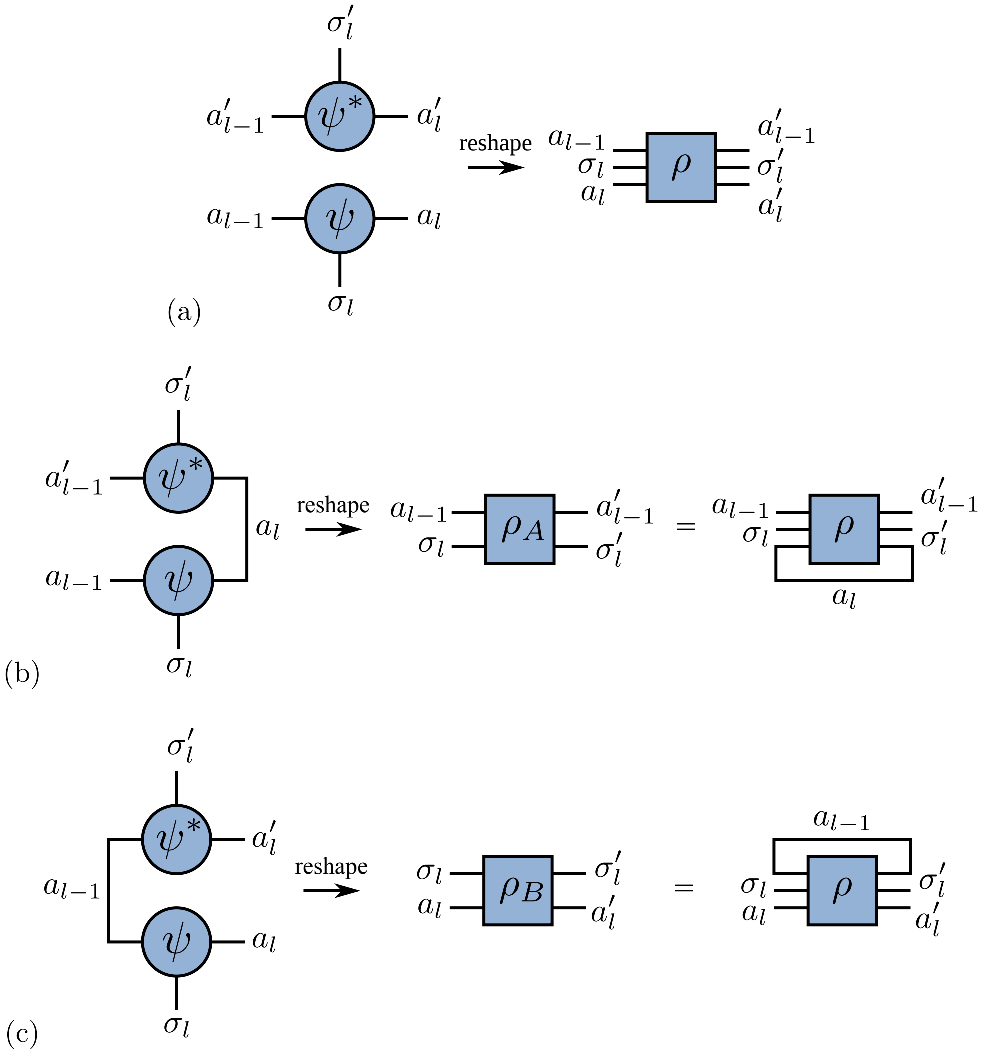

Figure 2.30: (a) Density matrix ρ made from two MPS tensors. Reduced density matrices (b) ρ_A and (c) ρ_B made by tracing over B and A degrees of freedom of the full density matrix.

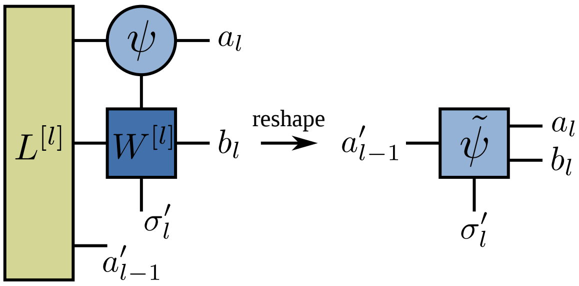

Figure 2.31: Wavefunction for use in the modified density matrix.

sweep. The eigenvectors of the largest χ eigenvalues are kept to truncate the basis. The MPS form of the modified density matrix is given by ref. [30]. The modification comes from coupling the ground state wavefunction to the system block, reshaping and contracting to give a perturbation to the normal density matrix

$$\begin{aligned}\rho_A \to \rho'_A &= \rho_A + \alpha\tilde{\rho}_A, \\ &= \mathrm{Tr}_B \left|\psi\right\rangle \left\langle\psi\right| + \alpha \mathrm{Tr}_B |\tilde{\psi}\rangle\langle\tilde{\psi}|,\end{aligned} \tag{2.7.18}$$

where α gives the weight of the modification. The starting value of α depends on the problem at hand, ref. [30] suggests 0.0001 for the spin-1/2 Heisenberg model. Throughout the sweeps α will be reduced to zero. The perturbation is constructed by contracting the ground state wavefunction with the left block and W tensor for the site

$$\tilde{\psi}^{\sigma'_l}_{a'_{l-1}a_l b_l} = \sum_{\sigma_l}\sum_{a_{l-1}}\sum_{b_{l-1}} L^{[l]}_{a_{l-1},b_{l-1},a'_{l-1}} W^{\sigma_l\sigma'_l}_{b_{l-1}b_l} \psi^{\sigma_l}_{a_{l-1}a_l}, \tag{2.7.19}$$

as shown in fig. 2.31. The perturbed density matrix in full is

$$[\tilde{\rho}_A]^{\sigma',\tilde{\sigma}'}_{a'_{l-1},\tilde{a}'_{l-1}} = \sum_{b_l}\sum_{a_l} \tilde{\psi}^{\sigma'_l}_{a'_{l-1}a_l b_l} \tilde{\psi}^{*\tilde{\sigma}'_l}_{\tilde{a}'_{l-1}a_l b_l}, \tag{2.7.20}$$

where the tilded indices make the distinction between the indices of $\tilde{\psi}$ and $\tilde{\psi}^*$.

Just as with the original DMRG algorithm this density matrix is diagonalised and the χ states with the largest eigenvalues are stored in columns as a matrix. The A tensor for this site is made by reshaping the matrix of column vectors into a three index tensor. This approach is repeated for each site. For the right sweep the equivalent formulae are created but the other side of the system is traced over

$$\rho_B \to \rho'_B = \rho_B + \alpha\tilde{\rho}_B. \tag{2.7.21}$$

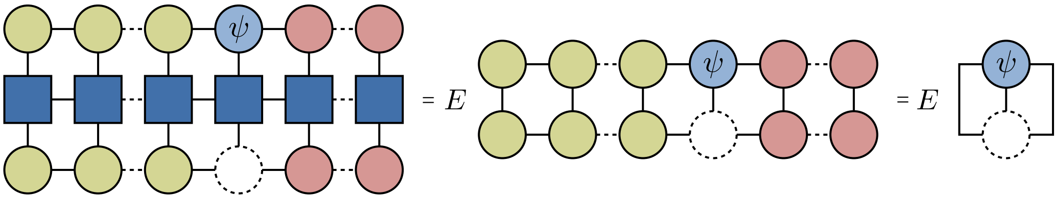

Figure 2.32: Eigenvalue equation for OBCs. The mixed canonical form of the MPS ensures that the problem is simplified from a generalised to a standard eigenvalue problem.

After diagonalisation the eigenvectors of the density matrix need to be stored as rows in a matrix and reshaped to give the right normalisation of the B tensors.

The modified density matrix approach is akin to an annealing process, whereby the contribution to the density matrix from this modification acts like heat or noise and it is gradually removed allowing the system to find the global minimum rather than a local one.

2.8 Periodic Boundary Conditions

The DMRG algorithm describe in section 2.7 is for OBCs, which is where DMRG is most efficient. The reason for this is down to the normalisation of the MPS. When an MPS tensor at site l is being updated the diagonalisation is implicitly solving a generalised eigenvalue problem [30]

$$H^{[l]}\psi = EN^{[l]}\psi, \tag{2.8.1}$$

where $H^{[l]}$ is the effective Hamiltonian operator (2.7.13) for site l and $N^{[l]}$ is the normalisation for site l defined as

$$N^{[l]} = \sum_{\sigma_1,\ldots,\sigma_L} \sum_{\substack{a_1,\ldots,a_{L-1}\\ a'_1,\ldots,a'_{L-1}}} A^{\sigma_1}_{a_1} \ldots A^{\sigma_{l-1}}_{a_{l-2}a_{l-1}} B^{\sigma_{l+1}}_{a_l a_{l+1}} \ldots B^{\sigma_L}_{a_{L-1}} \\ \times A^{*\sigma_1}_{a'_1} \ldots A^{*\sigma_{l-1}}_{a'_{l-2}a'_{l-1}} B^{*\sigma_{l+1}}_{a'_l a'_{l+1}} \ldots B^{*\sigma_L}_{a'_{L-1}}. \tag{2.8.2}$$

Equation (2.8.1) is shown diagrammatically in fig. 2.32. Due to the left and right normalisation of the A and B tensors $N^{[l]}$ is simply an identity matrix, which reduces the generalised eigenvalue problem (2.8.1) to the standard eigenvalue problem of eq. (2.7.14).

PBCs have long been known to be a problem for DMRG [15] but are nevertheless very useful, particularly in obtaining certain order parameters to classify the phase of a system, such as the superfluid density, which will be discussed in chapter 3. The following sections will introduce the traditional *poor man's PBC* as well as the natural method of introducing periodicity using MPS.

2.8.1 Poor Man's PBC

The essential idea of the *poor man's PBC* is to use the efficiency and simplicity of open boundary DMRG algorithm but apply it to an open system using one long range interaction to connect the first and last sites through the rest of the chain. It is only the Hamiltonian MPO that needs changing to implement this idea. Taking the nearest neighbour Hamiltonian of eq. (2.6.5) as a simple example, adding a PBC term gives

$$H = \sum_{i=1}^{L-1} A_i A_{i+1} + A_1 A_L. \tag{2.8.3}$$

The PBC term is included in the MPO by inserting an extra *identity row* in the $W^{[i]}$ that allows the interaction between the first and the last site. The MPO tensors take the form

$$\begin{aligned}
W^{[1]\sigma_1\sigma'_i}_{1b_1} &= \begin{pmatrix} \mathbb{1}^{\sigma_1\sigma'_1} & A_1^{\sigma_1\sigma'_1} & 0 & A_1^{\sigma_1\sigma'_1} \end{pmatrix}, & (2.8.4)\\
W^{[i]\sigma_i\sigma'_i}_{b_{i-1}b_i} &= \begin{pmatrix} \mathbb{1}^{\sigma_i\sigma'_i} & A_i^{\sigma_i\sigma'_i} & 0 & 0 \\ 0 & 0 & A_i^{\sigma_i\sigma'_i} & 0 \\ 0 & 0 & \mathbb{1}^{\sigma_i\sigma'_i} & 0 \\ 0 & 0 & 0 & \mathbb{1}^{\sigma_i\sigma'_i} \end{pmatrix}, & (2.8.5)\\
W^{[L]\sigma_L\sigma'_L}_{b_{L-1}1} &= \begin{pmatrix} 0 \\ A_L^{\sigma_L\sigma'_L} \\ \mathbb{1}^{\sigma_L\sigma'_L} \\ A_L^{\sigma_L\sigma'_L} \end{pmatrix}, & (2.8.6)
\end{aligned}$$

which is given as a matrix product diagram in fig. 2.33. When multiplied out fully with tensor products this is

$$\begin{aligned}
H^{\sigma_1\sigma'_1\ldots\sigma_L\sigma'_L} = {} & A_1^{\sigma_1\sigma'_1} \otimes A_2^{\sigma_2\sigma'_2} \otimes \mathbb{1}^{\sigma_3\sigma'_3} \otimes \cdots \otimes \mathbb{1}^{\sigma_i\sigma'_i} \otimes \cdots \otimes \mathbb{1}^{\sigma_L\sigma'_L} \\
& + \mathbb{1}^{\sigma_1\sigma'_1} \otimes A_2^{\sigma_2\sigma'_2} \otimes A_3^{\sigma_3\sigma'_3} \otimes \cdots \otimes \mathbb{1}^{\sigma_i\sigma'_i} \otimes \cdots \otimes \mathbb{1}^{\sigma_L\sigma'_L} + \ldots \\
& \cdots + \mathbb{1}^{\sigma_1\sigma'_1} \otimes \cdots \otimes A_i^{\sigma_i\sigma'_i} \otimes A_{i+1}^{\sigma_{i+1}\sigma'_{i+1}} \otimes \cdots \otimes \mathbb{1}^{\sigma_L\sigma'_L} + \ldots \\
& \cdots + A_1^{\sigma_1\sigma'_1} \otimes \mathbb{1}^{\sigma_2\sigma'_2} \otimes \cdots \otimes \mathbb{1}^{\sigma_{L-1}\sigma'_{L-1}} \otimes A_L^{\sigma_L\sigma'_L},
\end{aligned} \tag{2.8.7}$$

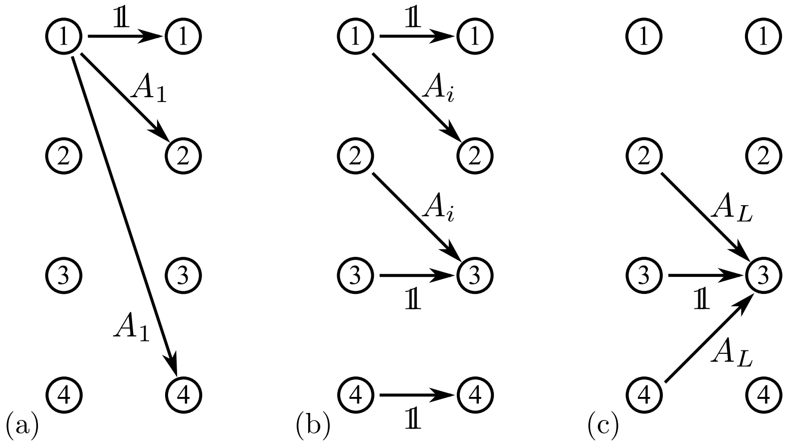

Figure 2.33: Matrix product diagram for the MPO tensors of a nearest neighbour Hamiltonian with PBCs for (a) site 1, (b) site i and (c) site L.

which is the desired form for a periodic system.

As a more useful example, the spin-1/2 Heisenberg Hamiltonian has the following MPO tensors

$$W^{[1]}_{1,b_1} = \begin{pmatrix} \mathbb{1} & \frac{J}{2}s^+ & \frac{J}{2}s^- & Js^z & -hs^z & s^+ & s^- & s^z \end{pmatrix}, \tag{2.8.8}$$

$$W^{[i]}_{b_{i-1},b_i} = \begin{pmatrix} \mathbb{1} & \frac{J}{2}s^+ & \frac{J}{2}s^- & Js^z & -hs^z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s^- & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s^+ & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & s^z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbb{1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbb{1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \mathbb{1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbb{1} \end{pmatrix}, \tag{2.8.9}$$

$$W^{[L]}_{b_{L-1},1} = \begin{pmatrix} -hs^z \\ s^- \\ s^+ \\ s^z \\ \mathbb{1} \\ \frac{J}{2}s^- \\ \frac{J}{2}s^+ \\ Js^z \end{pmatrix}, \tag{2.8.10}$$

where the spin indices have been omitted for clarity. The matrix product diagram

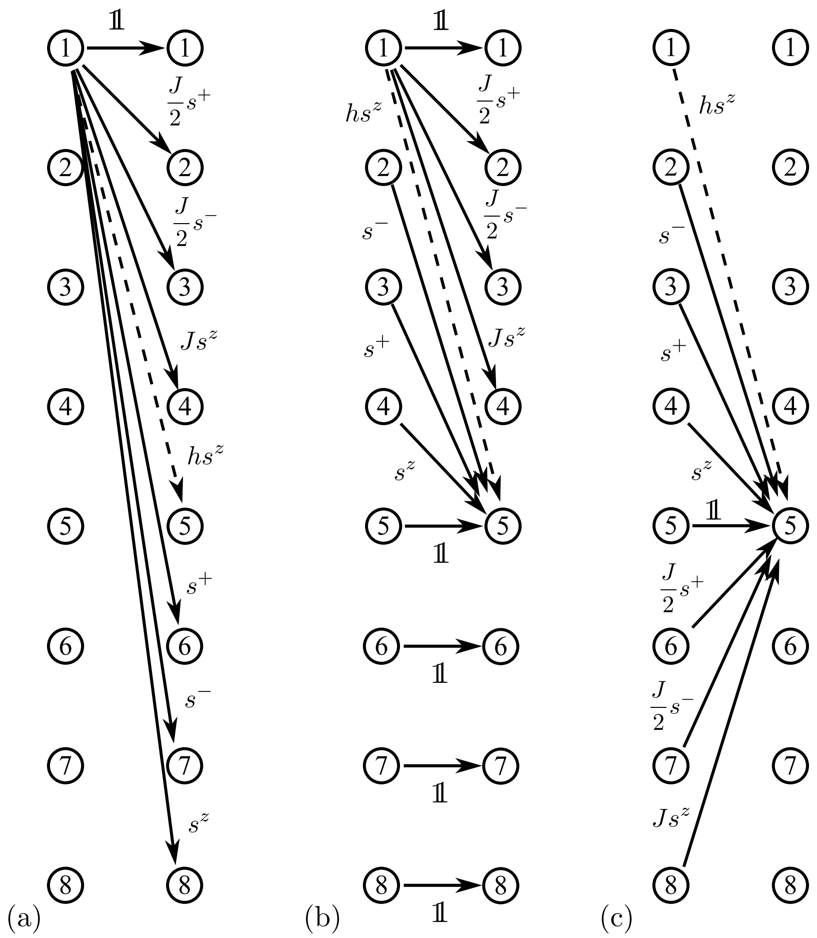

Figure 2.34: Matrix product diagram for the MPO tensors of a Heisenberg Hamiltonian with PBCs for (a) site 1, (b) site i and (c) site L.

is given in fig. 2.34. Whilst simple to implement, this method of introducing periodicity is not optimal as there is an inherent correlation length in an MPS, as will be discussed in more detail in chapter 4. This means that interaction between two sites that would otherwise be next to each other are suppressed due to the distance between them. To reconcile this and obtain more accurate results, the bond dimension has to be increased at the cost of computational resources and time.

2.8.2 Matrix Product States with PBCs

One of the advantages of MPS is that there is an intuitive method of introducing PBCs. For an MPS made up of L sites, PBCs are introduced by contracting the right hand index of site L with the left hand index of site 1, which is equivalent to

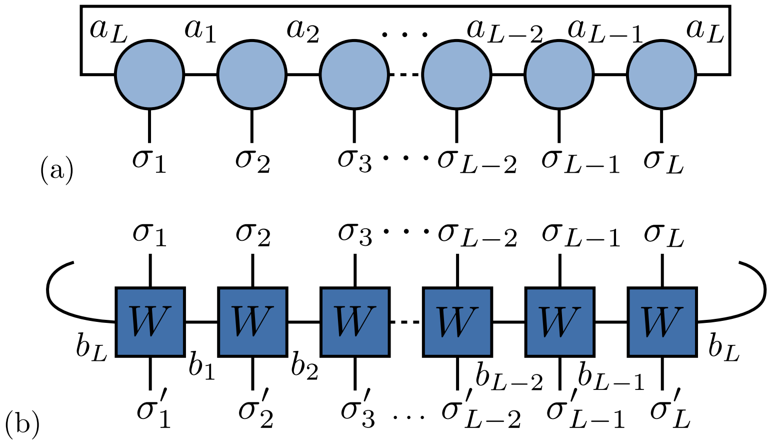
(a)
a_L a_1 a_2 … a_{L-2} a_{L-1} a_L
σ_1 σ_2 σ_3 ⋯ σ_{L-2} σ_{L-1} σ_L
(b)
σ_1 σ_2 σ_3 ⋯ σ_{L-2} σ_{L-1} σ_L
W
b_L b_1 b_2 b_{L-2} b_{L-1} b_L
σ'_1 σ'_2 σ'_3 … σ'_{L-2} σ'_{L-1} σ'_L

Figure 2.35: An (a) MPS and (b) MPO with PBCs is created by contracting the first and last tensors of the chain. The b_L connection is understood as going behind the diagram, which will make for less cluttered diagrams when discussing DMRG.

taking a trace over the bond indices [30]

$$\begin{aligned} |\Psi\rangle &= \sum_{\sigma_1,\ldots,\sigma_L} \sum_{a_1,\ldots,a_L} M^{\sigma_1}_{a_L a_1} M^{\sigma_2}_{a_1 a_2} \ldots M^{\sigma_L}_{a_{L-1} a_L} |\sigma_1,\ldots,\sigma_L\rangle \\ &= \sum_{\sigma_1,\ldots,\sigma_L} \mathrm{Tr}\left(M^{\sigma_1} M^{\sigma_2} \ldots M^{\sigma_L}\right) |\sigma_1,\ldots,\sigma_L\rangle . \end{aligned} \tag{2.8.11}$$

Diagrammatically this corresponds to drawing an adjoining line between the ends of the chain, effectively constructing a circle of tensors. This is shown in fig. 2.35(a). In a similar manner, MPOs can be constructed with PBCs by contracting the tensors on sites 1 and L

$$\begin{aligned} \mathcal{O} &= \sum_{\substack{\sigma_1,\ldots\sigma_L \\ \sigma'_1\ldots\sigma'_L}} \sum_{b_1,\ldots,b_L} W^{\sigma_1\sigma'_1}_{b_L b_1} W^{\sigma_2\sigma'_2}_{b_1 b_2} \ldots W^{\sigma_L\sigma'_L}_{b_{L-1} b_L} |\sigma_1\ldots\sigma_L\rangle\langle\sigma'_1\ldots\sigma'_L| \\ &= \sum_{\substack{\sigma_1,\ldots\sigma_L \\ \sigma'_1\ldots\sigma'_L}} \mathrm{Tr}\left(W^{\sigma_1\sigma'_1} W^{\sigma_2\sigma'_2} \ldots W^{\sigma_L\sigma'_L}\right) |\sigma_1\ldots\sigma_L\rangle\langle\sigma'_1\ldots\sigma'_L| , \end{aligned} \tag{2.8.12}$$

which is shown diagrammatically in fig. 2.35(b). In matrix product diagrams a trace over the bond index is equivalent to only allowing paths that start and finish on the same number. If the trace was taken naively over an MPO formed from tensors of the form of eq. (2.6.8), the result would be two identity matrices. This can be clearly seen in the matrix product diagram. The way to solve the problem is to

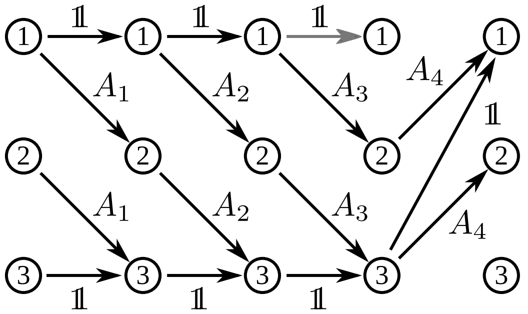

Figure 2.36: Matrix product diagram of a four site Hamiltonian MPO with PBCs.

create tensors for sites 1 and L that allow non-trivial paths that join up under the trace. For the simple nearest neighbour Hamiltonian (2.6.5), the first tensor can be the same as the rest (2.6.8) and the last tensor takes the form

$$[W^{[L]}_{PBC}]_{b_{L-1},b_L} = \begin{pmatrix} 0 & 0 & 0 \\ A_L & 0 & 0 \\ \mathbb{1} & A_L & 0 \end{pmatrix}. \tag{2.8.13}$$

An example of a four-site Hamiltonian MPO with PBCs is given by fig. 2.36.

Now that a set of PBC MPOs has been created, it is possible to start creating the DMRG algorithm. As mentioned, one of the important differences between PBC and OBC chains is the normalisation. With PBCs a generalised eigenvalue problem of eq. (2.8.1) needs to be solved. The normalisation is also an issue when building the left and right blocks. The fact that the right and left sides of the chain are joined means that the notion of a *left* and *right* block is lost. One of the crucial differences between the two blocks in the open case was that one had left normalisation and the other had right. This change in normalisation cannot occur with PBC as the normalisation would change part way through the block. Following the suggestion of ref. [25], one of the normalisations is chosen for one sweep direction. Here, for the left or *anti-clockwise* direction, right normalised tensors are used so that the standard SVD methods can be used without alteration.

The DMRG algorithm for PBCs is essentially the same as for open systems, remembering to connect the left and right ends of the chain. Despite the notion of left and right blocks being lost, the blocks will still be used as a means of efficiently calculating and storing the contracted tensors. As before, the algorithm starts with a set of right-normalised MPS tensors, however the bond dimension is set as χ for every site. The *left block* is calculated for sites 2 to L iteratively by contraction as

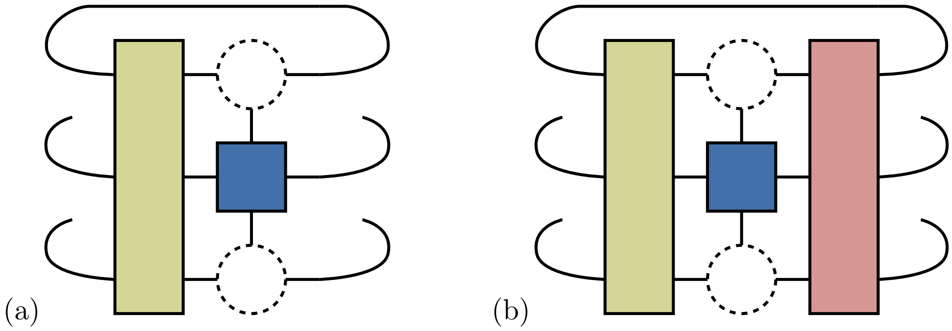

Figure 2.37: Effective Hamiltonian (a) for site L built from the left block and (b) for site i built from the left and right blocks with the left hand indices *wrapped round* to become the right hand side of the system under PBCs.

before. As the eigenvalues and eigenvectors are found using a generalised eigenvalue equation, the normalisation tensor N needs to be built for each diagonalisation. This will be done in the same way as the left and right blocks but without using an MPO.

Starting with a left sweep, the effective Hamiltonian is built from the left block $L^{[L]}$ *wrapped round* so that the left hand legs become the right hand side of the system as shown in fig. 2.37(a). The ground state eigenvector is found using a generalised eigensolver algorithm and is right normalised by performing an SVD. As before the prediction for the next site is calculated by contracting the MPS tensor for site $L-1$ with the U and S matrices of the SVD. The newly found MPS tensor for site L is contracted with $W^{[L]}$ and its conjugate to form the right block for the next iteration and contracted with its conjugate alone to form the right normalisation block for the next step. The update for the following sites is performed in a similar manner except that the Hamiltonian tensor is formed by contracting the left hand legs of the left block with the right hand legs of the right block as shown in fig. 2.37(b). This procedure is repeated until site 1, at which point the algorithm changes direction and the right sweep begins. The algorithm proceeds as normal until convergence.

Although now a much more natural and effective method for modelling PBCs this comes at the cost of using the more complicated and expensive generalised eigenvalue problem as well as the left and right blocks now have more indices making the process scale as $O(\chi^5 D_W^2)$. More recently this last issue was resolved by performing sweeps on thirds of the chain at a time [26, 27] in a manner more reminiscent of the OBC algorithm. This new PBC algorithm is once again $O(\chi^3)$ but is more complicated to implement.

2.9 Conclusions

In this chapter we have introduced the basic concepts of MPS based DMRG, including the fundamental tensor operations and a detailed description of the variational update. These principles underly all tensor network algorithms and will provide a foundation for the chapters that follow. MPS DMRG is heralded as one of the most powerful and versatile methods for the study of one dimensional quantum systems. It is a conceptual advance on White's DMRG that has enabled a better understanding of the strengths and weaknesses of the method, allowed new applications and paved the way for the field of tensor networks.

Chapter 3

Phases of the Disordered Bose-Hubbard Model

3.1 Introduction

The DMRG algorithm described in chapter 2 is a remarkably powerful tool for the study of one dimensional quantum systems. It is therefore an ideal method for the analysis of disordered bosons on a one dimensional lattice. This has been of great interest in both theoretical and experimental physics for many years due to the existence of a quantum phase transition from a superfluid to insulator at zero temperature [53]. The experimental study of phase transitions in bosonic systems is possible using Josephson junction arrays [54], thin films [55, 56] and, more recently, optical lattices [57]. The introduction of disorder causes a further phase transition into a localised *Bose glass* phase, which is an insulating phase but remains compressible [53]. Recently it has been possible to introduce disorder in a controlled manner in optical lattices using speckle potentials to study these phases experimentally [58, 59, 60, 61]. Analytical results for even clean systems are limited to an approximate Bethe-ansatz solution where the basis is truncated at two bosons per site [62]. For disordered systems Giamarchi and Schulz used renormalization group techniques to determine the weak disorder physics given the Luttinger parameter K [63, 64].

Numerical approaches provide perhaps the most effective means of garnering information. QMC has been used in 1, 2 and 3 dimensions [65, 66, 67, 68, 69, 70], but these methods become difficult in the limit of zero temperature. DMRG is inherently a zero temperature method and White style DMRG has been applied to the problem with mixed success [71, 72]. Interestingly, the phase diagrams from QMC

[66] and DMRG [72], whilst qualitatively similar, are quantitatively rather different. In recent years the use of entanglement properties as a means of deciphering phases has become commonplace [73, 74, 75, 76, 77, 78]. Entanglement is a measurement of a wavefunction's non-locality and as such it is an ideal means of analysing various phases. Modern numerical techniques such as tensor networks and DMRG obtain entanglement information as part of the update algorithms, so large amounts of information about the phase is gathered *for free* [30].

3.2 The Bose-Hubbard Model

In this chapter we use the ITensor libraries [79] to perform a DMRG simulation of the disordered Bose-Hubbard model in the form of a variational update of an MPS [20, 30] as discussed in chapter 2. The disordered Bose-Hubbard model is made up of bosonic creation $b_i^\dagger$ and annihilation b_i operators on sites of a 1 dimensional lattice. The Hamiltonian [72] is

$$H = -\frac{t}{2}\sum_i (b_i^\dagger b_{i+1} + \text{h.c.}) + \sum_i \frac{U}{2} n_i(n_i - 1) + \mu_i n_i, \tag{3.2.1}$$

where $n_i = b_i^\dagger b_i$ is the local occupation or *number operator* that gives the number of bosons on site i. The strength of the kinetic or *hopping* term is given by t, the Hubbard U term provides an on-site repulsion when $U > 0$ and the on-site or chemical potential strength is determined by μ_i. The disorder in the system is introduced to the on-site term by selecting the value of μ_i at each site i from a box distribution of values from $-\Delta\mu/2$ to $\Delta\mu/2$. Throughout the chapter $\Delta\mu$ will be used as a measure of the strength of the disorder on the chemical potential. We have adopted the conventions from Rapsch et. al. [72] in that t and U are both divided by two and throughout the rest of the analysis we keep $t = 1$.

The basis used for the model is the number of particles and as bosons do not obey the Pauli exclusion principle, all particles can occupy one site. This means that the one-site basis dimension needs be be as large as there are particles in the system. In this chapter we will be looking at systems with up to 300 particles; including a full basis would greatly increase computational cost. It is therefore necessary to truncate the number of bosons that can occupy each site. We set the maximum number to 5 bosons which corresponds to the current limit in the ITensor code. This is consistent with ref. [80] who find that a higher particle number does not effect the results appreciably for $U > 0$ [71, 72, 81]. Using the truncated basis, the operators have matrix form

$$b = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{3} & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{4} & 0 \\ 0 & 0 & 0 & 0 & 0 & \sqrt{5} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} , \quad n = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 5 \end{pmatrix} \qquad (3.2.2)$$

The MPO tensors are then

$$W^{[i]} = \begin{pmatrix} \mathbb{1} & -\frac{t}{2}b^\dagger & -\frac{t}{2}b & \mu n + \frac{U}{2}n(n-\mathbb{1}) \\ 0 & 0 & 0 & b \\ 0 & 0 & 0 & b^\dagger \\ 0 & 0 & 0 & \mathbb{1} \end{pmatrix} , \qquad (3.2.3)$$

with standard boundary terms as described in section 2.6.1. Periodic boundary conditions will be implemented using the *poor man's PBC* method of section 2.8.1. The reason for choosing this as opposed to the full periodic MPS of section 2.8.2, is that the ITensor code doesn't have the full implementation in the current release and the poor man's PBCs are adequate for the one observable that periodicity is needed for.

3.3 Observables

Previous studies of the phases of the Bose-Hubbard model [72, 81] concentrate on order parameters appropriate for each phase. The superfluid phase is determined by a non-zero *superfluid fraction* ρ_s. This is defined as the difference between the ground state energies of a chain with PBCs and anti-periodic boundary conditions (APBCs)

$$\rho_s = \frac{2L^2}{\pi^2 N}\left[E_0^{\mathrm{APBC}} - E_0^{\mathrm{PBC}}\right] , \qquad (3.3.1)$$

where L is chain length and N is the number of bosons [72]. The motivation behind this definition is that a change in boundary conditions will affect the extended superfluid state but will leave the localised phases unchanged. Thus the superfluid fraction is zero for the Mott insulator and Bose glass phases so it should show where the superfluid phase is in the phase diagram. Superfluid fraction is not ideal as it is a property of periodic systems while we are interested in the Bose-Hubbard model with OBCs. Furthermore, as discussed in chapter 2, DMRG with PBCs converges slower and requires more resources than open systems. Also, as ρ_s is the difference between two energies, two DMRG calculations have to be performed for each set of parameters.

The Mott insulator can be differentiated from the Bose glass phase by the existence the *Mott gap*; an energy gap between the ground and first excited state. While DMRG ordinarily finds the ground state of the system, low lying excited states can be found iteratively by orthogonalising with respect to the lower lying states [30]. It is simpler for the Bose Hubbard chain to use the fact that the energy of the excited state is equal to the difference between the chemical potential for particle ($\mu_p = E_{N+1} - E_N$) and hole ($\mu_h = E_N - E_{N-1}$) excitations [72]. This means that the energy gap can be found by calculating the energies of the $N+1$ and $N-1$ particle sectors

$$E_g = E_{N+1} - 2E_N + E_{N-1}. \tag{3.3.2}$$

As discussed for ρ_s, the difference between energies of different systems requires multiple DMRG runs. For the energy gap calculated in this way it is necessary to run the DMRG three times for each set of parameters.

The two-point correlation function $\langle b_i^\dagger b_j \rangle$ also referred to as the *one-particle density matrix* [72] or *bosonic Green's function* [81] provides information regarding the localisation of the wavefunction. For the Bose glass and Mott insulating phases the correlation function decays exponentially [81]

$$\langle\langle b_i^\dagger b_j \rangle\rangle \propto e^{-|i-j|/\xi}, \tag{3.3.3}$$

where ξ is the correlation length and $\langle\langle \ldots \rangle\rangle$ denotes the expectation value when averaged over all pairs of sites separated by $|i-j|$ and all disorder realisations. Extended phases like the superfluid are not localised so the correlation length diverges. In the absence of disorder the superfluid phase will be described by Luttinger liquid theory, hence the correlation function will admit a power law decay

$$\langle\langle b_i^\dagger b_j \rangle\rangle \propto \frac{1}{|i-j|^{1/2K}}, \tag{3.3.4}$$

where K is the Luttinger parameter. K takes the value 2 for a Kosterlitz-Thouless (KT) transition from superfluid to Mott insulator [80, 81]. By utilizing an RG approach, Giamarchi and Schulz [63] showed that disorder scales to zero in the weak disorder regime when $K > 3/2$, giving a superfluid phase. On the other hand, disorder grows for $K < 3/2$ signifying a Bose glass. This was later extended [64] to

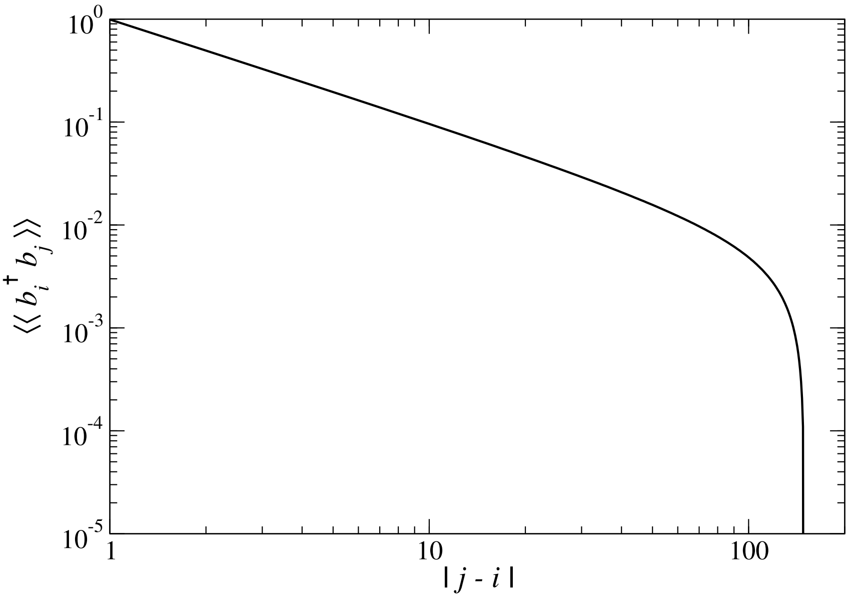

Figure 3.1: The two-point correlation function from eq. (3.3.5) for a chain of 150 sites. Averaged over all pairs of sites separated by $|j - i|$, where errors are smaller than the line width.

the *medium disorder* case ($U \sim \Delta\mu$).

Instead of a polynomial fit we use the conformal field theory (CFT) fit as described in [81] where the expected correlation for an open chain is used to fit the data and the Luttinger K can be extracted from the CFT formula. For a system size L the correlation function from the CFT is

$$\langle b_i^{\dagger} b_j \rangle \propto \left[\frac{\pi}{2L} \frac{\sqrt{\left|\sin\left(\frac{\pi i}{L}\right)\right| \left|\sin\left(\frac{\pi j}{L}\right)\right|}}{\left|\sin\left(\frac{\pi(i+j)}{2L}\right)\right| \left|\sin\left(\frac{\pi(i-j)}{2L}\right)\right|} \right]^{\frac{1}{2K}} . \qquad (3.3.5)$$

Due to the fact that this formula is for open boundaries there is no translational invariance, thus one has to calculate the correlation function for each i and j and average over all those with separation $|j - i|$. The result of which is plotted in fig. 3.1. Calculating correlation functions does not require multiple DMRG runs, but requires the calculation of an expectation value for each combination of i and j, of which there are $L(L-1)/2$. Furthermore, the accuracy of locating the KT transition from correlation functions for the Bose-Hubbard model has previously been questioned [81, 82].

When dealing with disordered systems it is necessary to disorder average, thus it is vital to reduce the computational cost of each realisation. In order to build a phase diagram using the standard order parameters one has to perform multiple calculations for each set of variables. We will concentrate on the entanglement spectrum as it encodes long-range information about the system as well as being *free* to obtain from DMRG as the singular values s_a for each bipartition of the chain are calculated as part of the algorithm. The most common entanglement measure is the entanglement entropy given by eq. (2.4.4), which gives the entanglement between regions A and B [30]. In the subsequent analysis we shall average the entanglement entropy over the possible bipartitions along the chain to obtain a number that can be used to build a phase diagram. The average entanglement entropy is very good at distinguishing between phases with high and low entanglement, for example the superfluid and Mott insulating phases.

Other information can be obtained from the singular values; Deng et al. [77] used an *entanglement spectral parameter* (ζ) to obtain the phase diagram for the extended Bose-Hubbard model. The ζ parameter is defined as the difference between the first and second, and third and fourth averaged eigenvalues of ρ_A when averaged over all bipartition positions such that $L_A + L_B = L$, i.e.

$$\zeta = \lambda_1^T - \lambda_2^T + \lambda_3^T - \lambda_4^T \tag{3.3.6}$$

where

$$\lambda_i^T = \frac{1}{L-1} \sum_{L_A=1}^{L-1} \lambda_i(L_A) = \frac{1}{L-1} \sum_{L_A=1}^{L-1} s_i^2(L_A) \tag{3.3.7}$$

is the average i-th eigenvalue. We see that the entanglement spectrum of the superfluid phase is somewhat noisy, but without too much variation along x. One of the striking features of the entanglement spectrum of the Mott insulator is that it is effectively a product state for all bipartitions, even in the presence of disorder. This means that the average entanglement entropy will be nearly zero with a negligible variance. Last, the entanglement spectra of the Bose-glass show pronounced localised peaks, resulting in a much larger variation of ζ (and $S_{\mathrm{A|B}}$) than in the superfluid phase. These findings suggest that the spatial variations of $S_{\mathrm{A|B}}$ and ζ might also be used to distinguish the phases of the disordered Bose-Hubbard model.

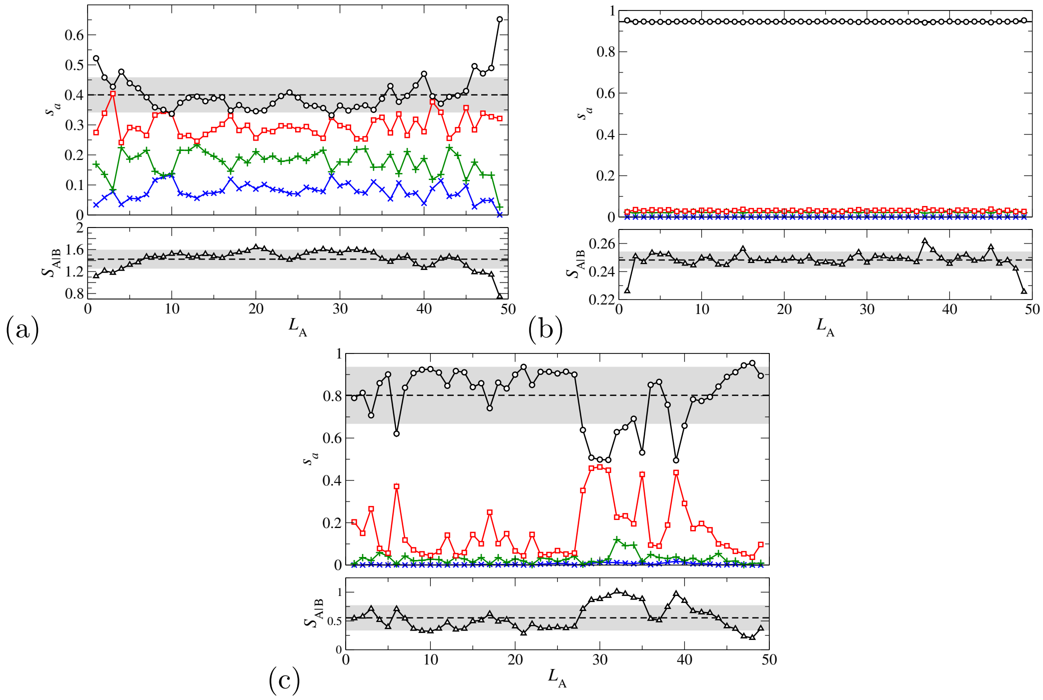

Figure 3.2: (Top) the largest four singular values, s_1 (black $\circ$), s_2 (red $\square$), s_3 (green $+$), and s_4 (blue $\times$) and (Bottom) the entanglement entropy S_{AB} (black $\triangle$) for all possible bipartition positions, L_{A}, along a chain of length $L = 50$ for (a) Superfluid with $U = 0.5$, $\Delta\mu = 1$. (b) Mott insulator with $U = 4.5$, $\Delta\mu = 1$. (c) Bose glass with $U = 4.5$, $\Delta\mu = 7$. The dashed horizontal line in the top (bottom) graph shows the average value of s_1 ($S_{\mathrm{A|B}}$) while the grey shading indicates its standard deviation when averaged over all L_{A} positions. Solid lines connecting symbols are guides to the eye.

3.4 Results

We collected most data for a modest system size of $L = 50$ in order to allow disorder-averaging over 100 samples. We increase L up to 200 for high-precision estimates of phase boundaries at certain $(U, \Delta\mu)$ points. For disordered systems, getting stuck in local minima is particularly problematic, so we use a relatively large bond dimension $\chi = 200$ and perform 20 sweeps of the chain for each sample such that the truncation error is less than 10^{-10}. We also use the modified density matrix discussed in chapter 2 to introduce a small noise term for the first few DMRG sweeps; this perturbs a perhaps bad initial wavefunction, letting the system converge faster into the desired target states.

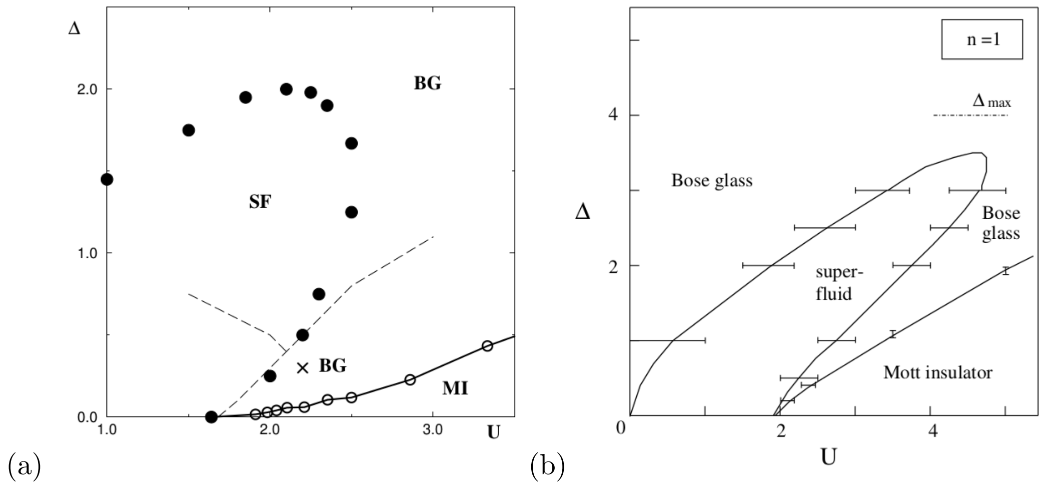

Figure 3.3: Previous phase diagrams for the disordered Bose-Hubbard model at $N/L = 1$ reproduced from (a) Prokof'ev and Svistunov [66], and (b) Rapsch et al. [72]. Note that in these diagrams $\Delta = \frac{1}{2}\Delta\mu$.

3.4.1 Density = 1

For particle density $\frac{N}{L} = 1$, in the clean case the system is in a superfluid phase for small U but transitions into a Mott insulating phase at a critical U_c. Introducing disorder enables the existence of a localized *Bose glass* phase. The possibility of a direct transition from superfluid to Mott insulator has been a contentious issue with many arguments on both sides (see references in [83]). In one dimension it was proved [84] that the transition necessarily goes via the Bose glass phase. It has subsequently been shown that there is no direct superfluid to Mott insulator transition with disorder in any dimension [83]. For reference, the previous phase diagrams of Prokof'ev and Svistunov [66], and Rapsch et al. [72] are provided in figs. 3.3 (a) and (b) respectively.

We show our results based on ζ and $S_{A|B}$ for $L = 50$ in fig. 3.4 (a) and (b), respectively. The superfluid, small $U \lesssim 1.5$, and the Mott insulator, $U \gtrsim 2$, are clearly distinguishable in both panels. The boundary of the superfluid to the Bose glass is less well defined and it is not even clear that there is a Bose glass region between the Mott insulator and the superfluid. In these situations, the two very different wavefunctions give similar average entanglement entropy. Following on from our discussion of fig. 3.2 in the last section, we also plot in fig. 3.4(c) the standard error of ζ, $\Delta\zeta$, and, similarly, (d) $\Delta S_{\mathrm{A|B}}$. In these plots the phases become clear and their boundaries are consistent with earlier work [72]. In particular, a Bose glass phase can be easily identified between Mott insulator and superfluid.

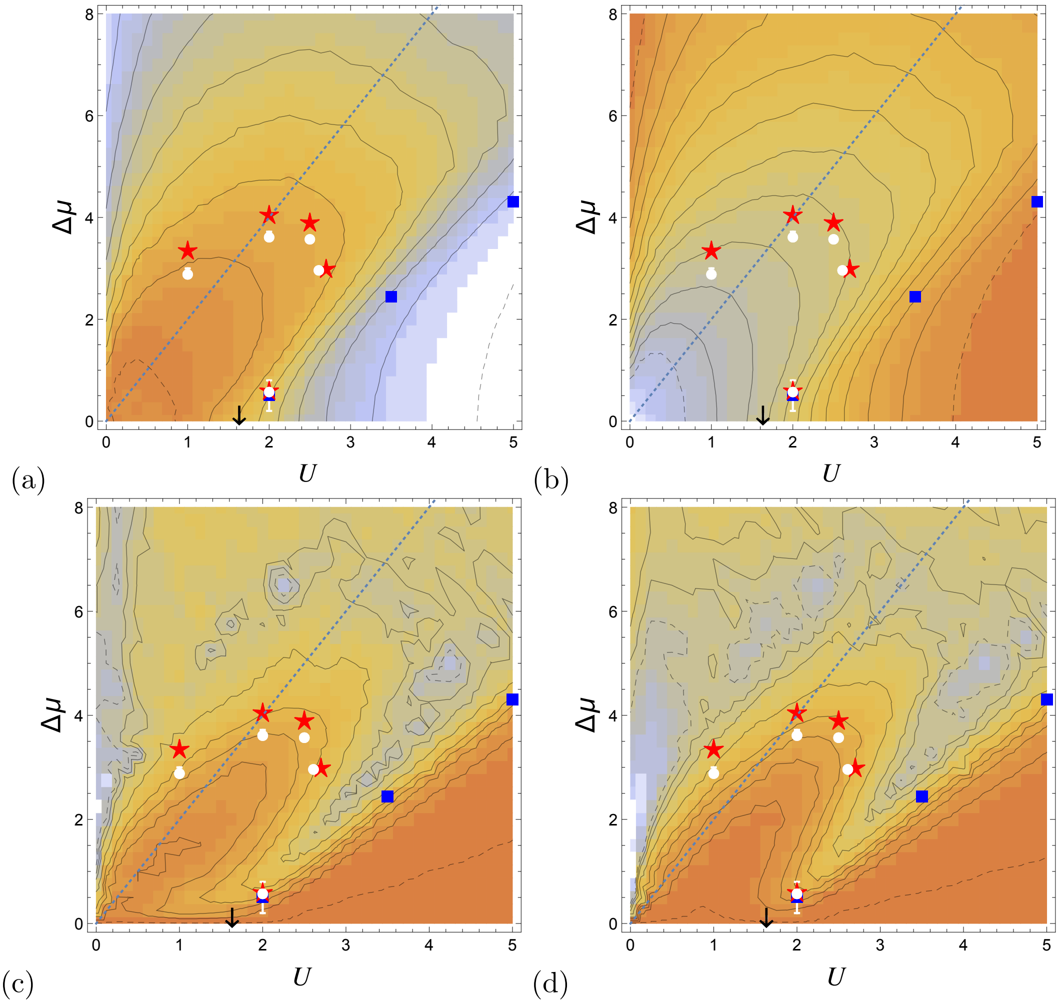

Figure 3.4: Phase diagrams for the disordered Bose-Hubbard model at $N/L = 1$ given as contour plots of (a) ζ, (b) $S_{A|B}$, (c) $\Delta\zeta$ and (d) $\Delta S_{A|B}$. The color shading goes from low (orange/dark) to high (blue/white) value and its coarse-graining reflects the $(U, \Delta\mu)$ resolution of our calculations for $L = 50$. The contour lines correspond to (a) $\zeta = 0.1, 0.2, \ldots, 0.9$, (b) $S_{A|B} = 0.3, 0.4, \ldots, 1.4$, (c) $\Delta\zeta/10^{-4} = 0.1, 1, 2, \ldots, 10$ and (d) $\Delta S_{A|B}/10^{-4} = 0.1, 1, 2, \ldots, 10$. In all cases the two extreme contours values are shown as dashed lines. The circles (white) and squares (blue) denote estimations of K and E_g from finite-size scaling for $L \to \infty$ while the stars (red) indicate the $K = 2$ values for $L = 50$ as discussed in the text. The arrow (black) denotes the expected transition in the clean case at U_c. The dotted straight line indicates $\Delta\mu = 2U$. Error bars (white) show the standard error of the mean in all cases and are not shown if within symbol size. We emphasize that the color shading does not directly indicate the transitions, but rather quantifies the change in entanglement measures.

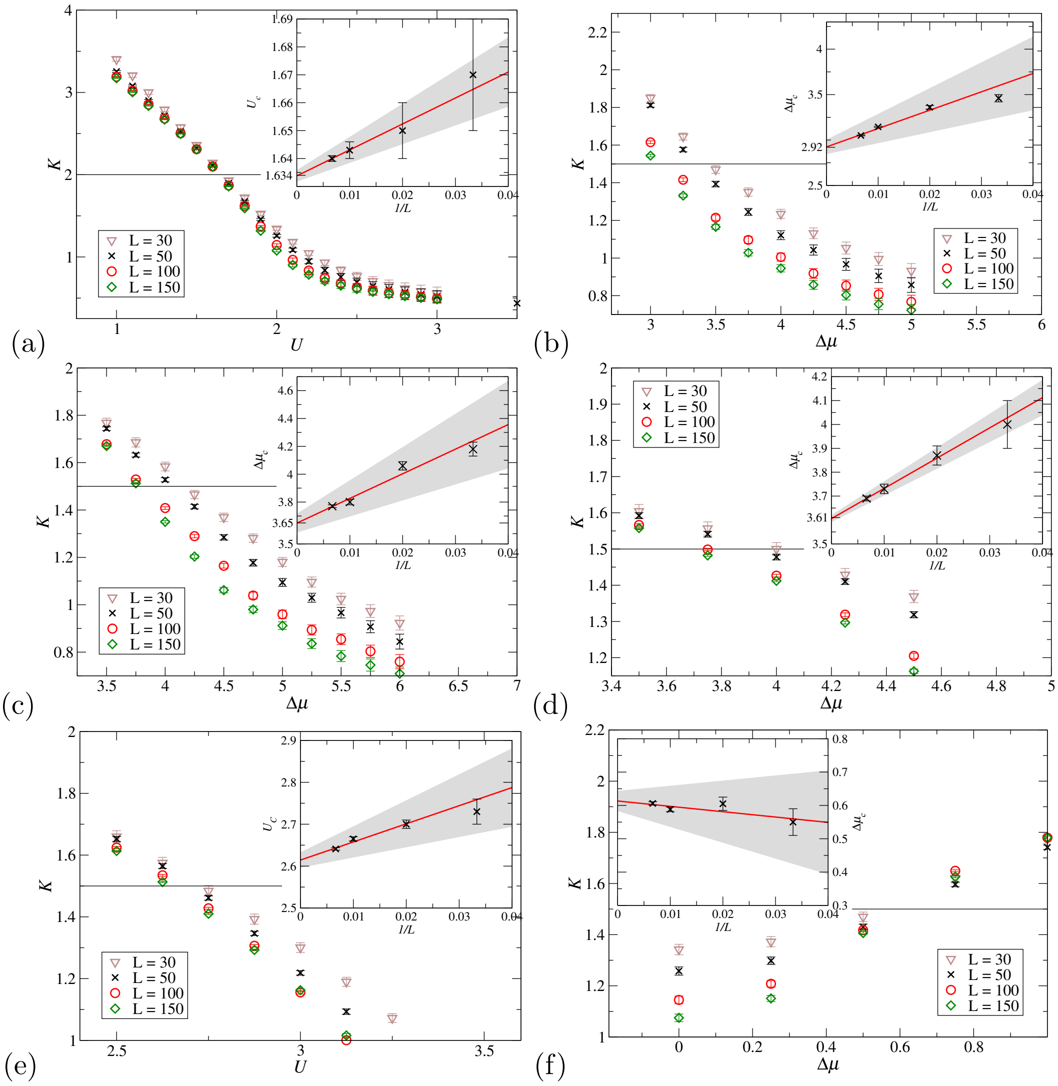

Figure 3.5: Luttinger parameter K for various lengths 30–150 at $N/L = 1$ for the points shown by white circles on the phase diagrams of fig. 3.4 at (a) $\Delta\mu = 0$, $U = 1, 1.1, \ldots, 3$; (b) $U = 1$, $\Delta\mu = 3, 3.25, \ldots, 5$; (c) $U = 2$, $\Delta\mu = 3.5, 3.75, \ldots, 6$; (d) $U = 2.5$, $\Delta\mu = 3.5, 3.75, \ldots, 4.5$; (e) $\Delta\mu = 3$, $U = 2.5, 2.625, \ldots, 3.25$; (f) $U = 2$, $\Delta\mu = 0, 0.25, \ldots, 1$. The horizontal line highlights $K = 3/2$ for the superfluid to Bose-glass transition in the presence of disorder (b-f) and $K = 2$ for the superfluid to Mott-insulator transition in the clean case (a). The inset shows the finite size scaling analysis to obtain the critical point in the thermodynamic limit.

Furthermore, we see that the contours for ζ and $S_{\mathrm{A|B}}$ in fig. 3.2 are qualitatively similar, just as those for $\Delta\zeta$ and $\Delta S_{\mathrm{A|B}}$. We emphasize that for the entanglement-based measures presented here, it is in fact possible to discern all of the phases with just a single DMRG run for each $(U, \Delta\mu, \text{disorder realisation})$ data point. This is a clear advantage in terms of numerical costs when compared to calculations based on E_g, ρ_s or K.

The standard error on ζ is calculated using a weighted standard deviation. The ζ parameter is calculated for each seed using

$$\zeta_s = \frac{1}{L-1}\sum_{L_A=1}^{L-1}\left[s_1(L_A)^2 - s_2(L_A)^2 + s_3(L_A)^2 - s_4(L_A)^2\right], \tag{3.4.1}$$

where $s_i(L_A)$ is the i-th singular value from an SVD with left block size L_A. The standard error for each seed is then

$$\Delta\zeta_s = \sqrt{\frac{1}{(L-1)^2}\sum_{L_A=1}^{L-1}\left[s_1(L_A)^2 - s_2(L_A)^2 + s_3(L_A)^2 - s_4(L_A)^2\right]^2 - \frac{\zeta_s^2}{L-1}}. \tag{3.4.2}$$

The average of ζ over all seeds is

$$\bar{\zeta} = \frac{\sum_s^N w_s\zeta_s}{\sum_s^N w_s}, \tag{3.4.3}$$

where w_s is the weight for each seed given by $w_s = 1/\Delta\zeta_s^2$ and N is the number of seeds. The final error is then

$$\Delta\bar{\zeta} = \sqrt{\frac{\sum_s^N w_s(\zeta_s - \bar{\zeta}_s)^2}{\sum_s^N w_s N^2}}. \tag{3.4.4}$$

The error on the entanglement entropy S is calculated in the same way.

In order to augment the finite-size phases identified in fig. 3.4, we perform runs with larger L and perform finite-size scaling. In order to find the superfluid-Bose glass transition in the thermodynamic limit we calculate K for various points along the boundary for system sizes $L = 30$, 50, 100 and 150 as shown in fig. 3.5(b-f). The transition is of KT type at $K = 3/2$. The corresponding points in $(U, \Delta\mu)$ which are shown as filled circles in fig. 3.4. For reference, we also plot the points where $K = 3/2$ for $L = 50$ (stars). Similarly, the superfluid-Mott insulator transition point U_c is the point on the zero disorder axis where $K = 2$ given by fig. 3.5(a). We estimate the critical value as $U_c = 1.634 \pm 0.002$. We also calculate E_g for

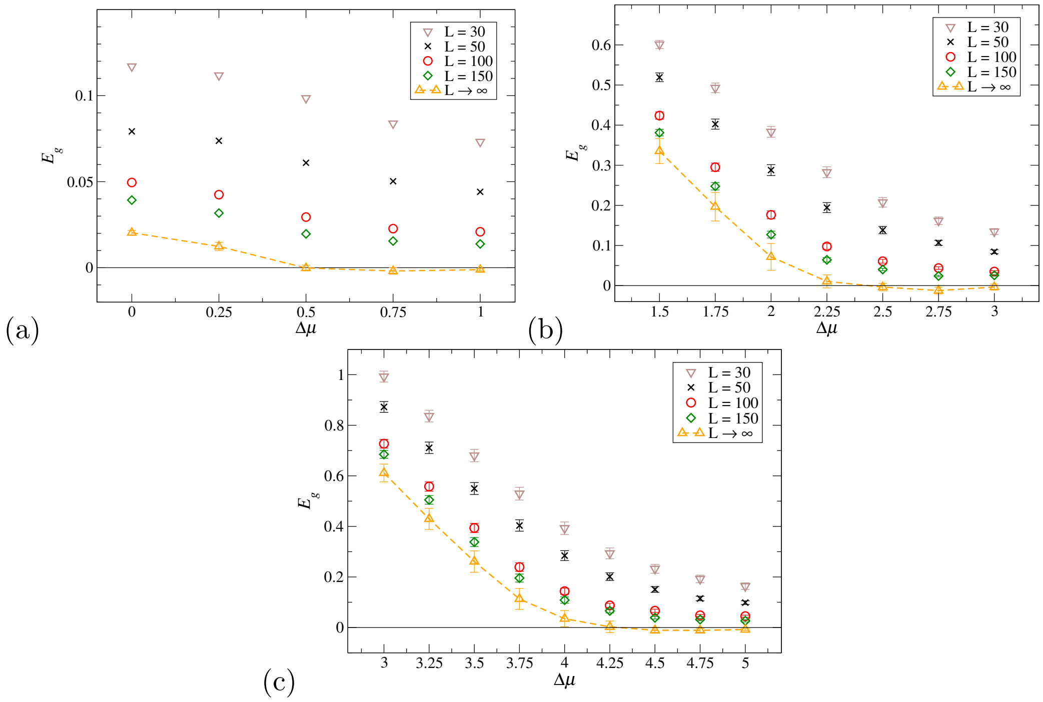

Figure 3.6: Energy gap E_g for various lengths 30–150 and the infinite size limit at $N/L = 1$ for the points shown by blue squares on the phase diagrams of fig. 3.4 at (a) $U = 2$, $\Delta\mu = 0, 0.25, \ldots, 1$; (b) $U = 3.5$, $\Delta\mu = 1.5, 1.75, \ldots, 3$; (c) $U = 5$, $\Delta\mu = 3, 3.25, \ldots, 5$. The horizontal line highlights $E_g = 0$ where the gap closes indicating a Bose-glass phase and the orange dashed line is a guide to the eye. Error bars have been omitted where smaller than symbol size.

the same system sizes and perform finite size scaling to find the Mott insulator-Bose glass boundary (fig. 3.6). These are indicated as blue squares in fig. 3.4.

The RG analysis of Refs. [63] and [64] suggests that there may be a further *Anderson glass* phase in the low $U < \Delta\mu$ region of the diagram. This would imply a critical point along the superfluid boundary at which point K at the transition becomes disorder dependent in a similar manner to the strongly disordered quantum rotor model [85, 86]. Our entanglement analysis along with E_g and K shows no sign of a transition. It does however become very difficult to observe a clear KT transition in the finite size scaling of K when $U \ll \Delta\mu$ and the truncation of the basis becomes more problematic in this region.

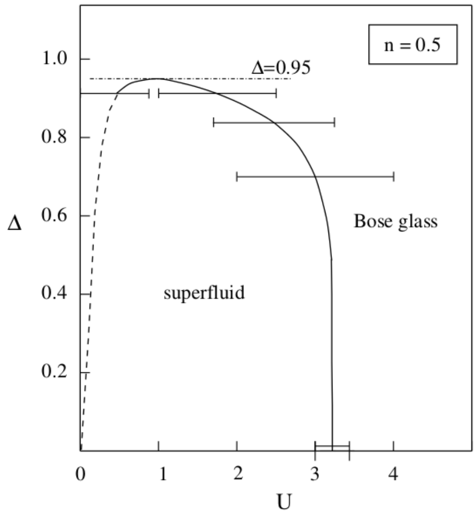

Figure 3.7: Previous phase diagram for the disordered Bose-Hubbard model at $N/L = 1/2$ reproduced from Rapsch et al. [72]. Again, in this diagram $\Delta = \frac{1}{2}\Delta\mu$.

3.4.2 Density = 1/2

When $\frac{N}{L} = \frac{1}{2}$ the clean case is simply a superfluid for all values of U [53] and when $\Delta\mu$ is increased a Bose glass phase is introduced. This is shown for the thermodynamic limit in fig. 3.7 from Rapsch at al. [72]. Our entanglement measures indicate that the superfluid phase for $L = 50$ seems to extend up to $\Delta\mu \lesssim 1$ for all $U \lesssim 5$ rather than disappearing with the presence of any disorder as in fig. 3.7. The Giamarchi-Schulz criterion [63, 72] implies that the Bose-Hubbard model should be in a Bose glass phase for $K < 3/2$. The resulting boundaries for the $L = 50$ case are indicated by the red stars in fig. 3.8. We find that the superfluid phase extends as far as $U_{K=3/2} = 3.5 \pm 0.1$, i.e. it ends somewhat earlier for low $\Delta\mu$ than suggested by our entanglement measures. In order to explore this region further, we have also calculated ρ_s for fixed $\Delta\mu = 0.5$ and sizes $L = 50$, 100, 150, and 200 as shown in fig. 3.10(a). The results for ρ_s have been computed for increased bond dimension $\chi = 400$ with 40 DMRG sweeps and 20 disorder configuration to offset the reduction in DMRG precision due to PBCs. The figure shows that for $U \gtrsim 3$, ρ_s decreases when increasing L as expected in the Bose glass phase. However, the decrease is very slow and, for the system sizes attainable by us, even seems to saturate at non-zero values. These results suggest that for finite systems, the $K = 3/2$ criterion significantly underestimates the extent of the superfluid phase, while our four entanglement measures and ρ_s predict a much larger region.

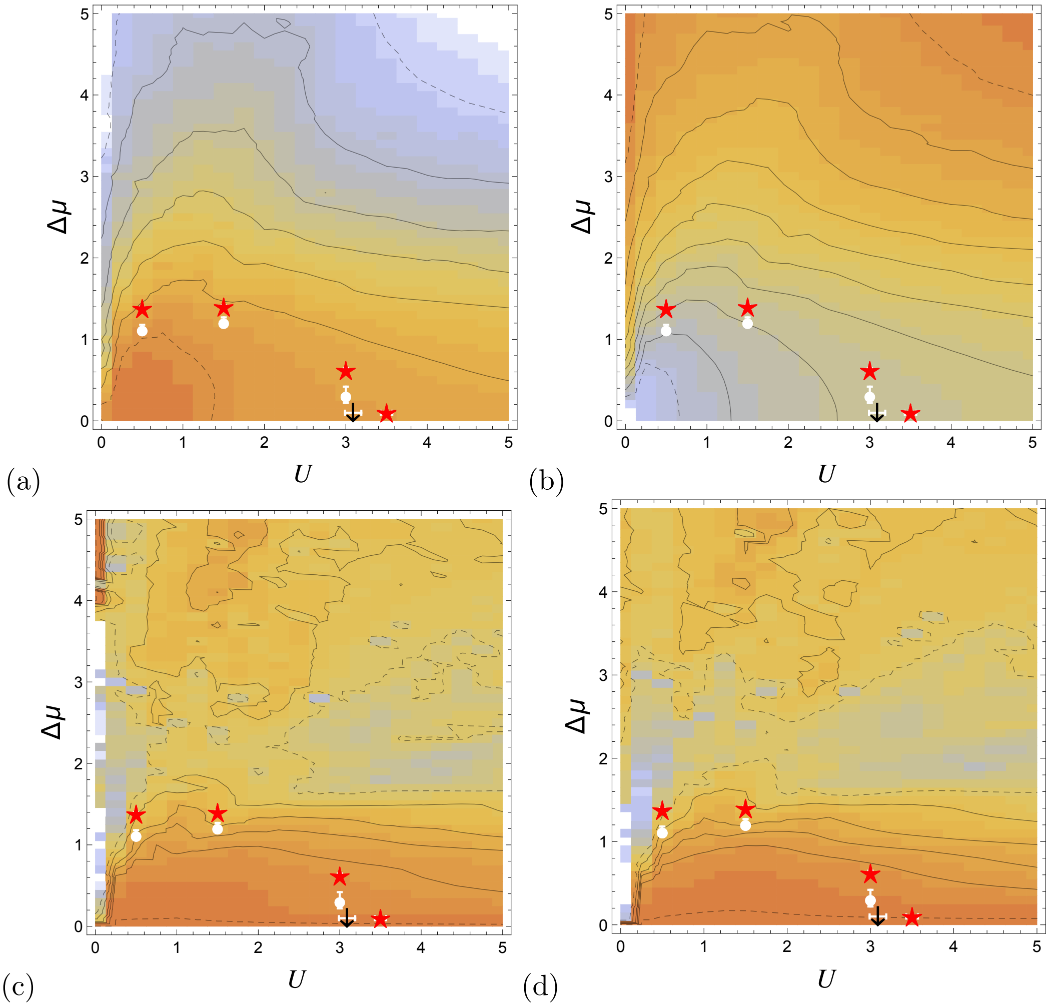

Figure 3.8: Phase diagrams for the disordered Bose-Hubbard model at $N/L = 1/2$ given as contour plots of (a) ζ, (b) $S_{A|B}$, (c) $\Delta\zeta$ and (d) $\Delta S_{A|B}$. Colors, symbols and lines (solid and dashed) denote corresponding estimates as in fig. 3.4. The black contour lines correspond to (a) $\zeta = 0.2, 0.3, \ldots, 0.8$, (b) $S_{A|B} = 0.3, 0.4, \ldots, 1.2$, (c) $\Delta\zeta/10^{-3} = 0.1, 1, 1.5, 2, 3, 4$ and (d) $\Delta S_{A|B}/10^{-3} = 0.1, 1, 1.5, 2, 3, 4$. The black arrow corresponds to U_c as discussed in the text.

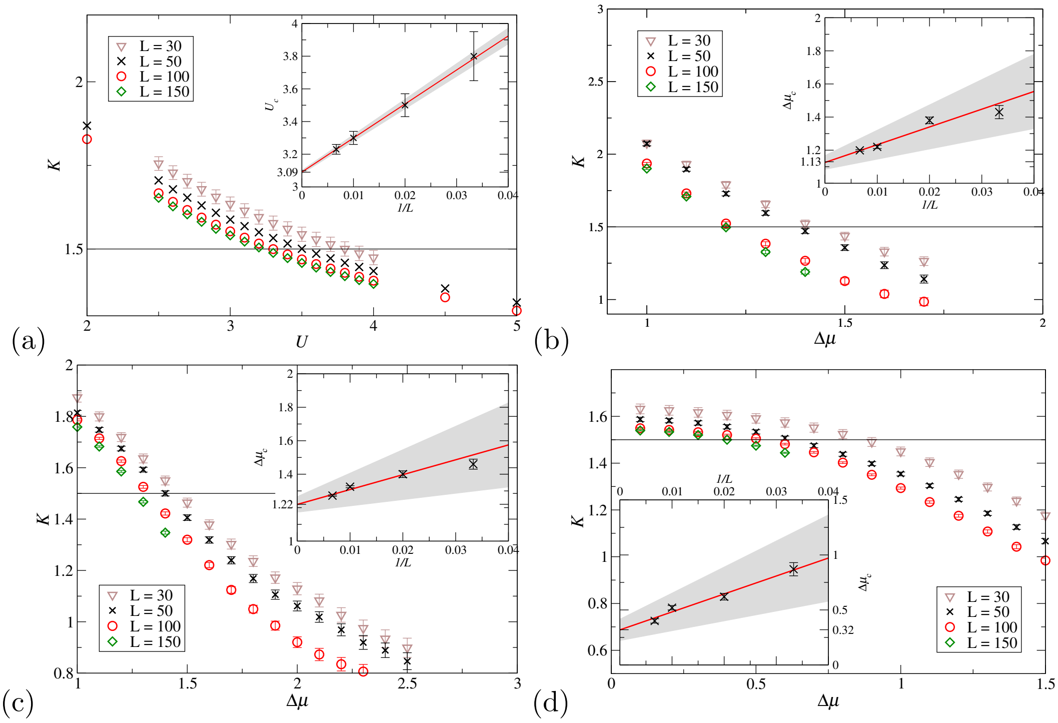

Figure 3.9: Luttinger parameter K for various lengths 30–150 at $N/L = 1/2$ for the points shown by white circles on the phase diagrams of fig. 3.8 at (a) $\Delta\mu = 0$, $U = 2.5, 2.6, \ldots, 4$; (b) $U = 0.5$, $\Delta\mu = 1, 1.1, \ldots, 2$; (c) $U = 1.5$, $\Delta\mu = 1, 1, 1, \ldots, 2.5$; (d) $U = 3$, $\Delta\mu = 0.1, 0.2, \ldots, 2$ As in fig. 3.5, the horizontal line highlights $K = 3/2$ for the superfluid to Bose-glass transition and there is no Mott-insulating phase when $N/L = 1/2$. Again, the inset shows the finite size scaling analysis to obtain the critical point in the thermodynamic limit.

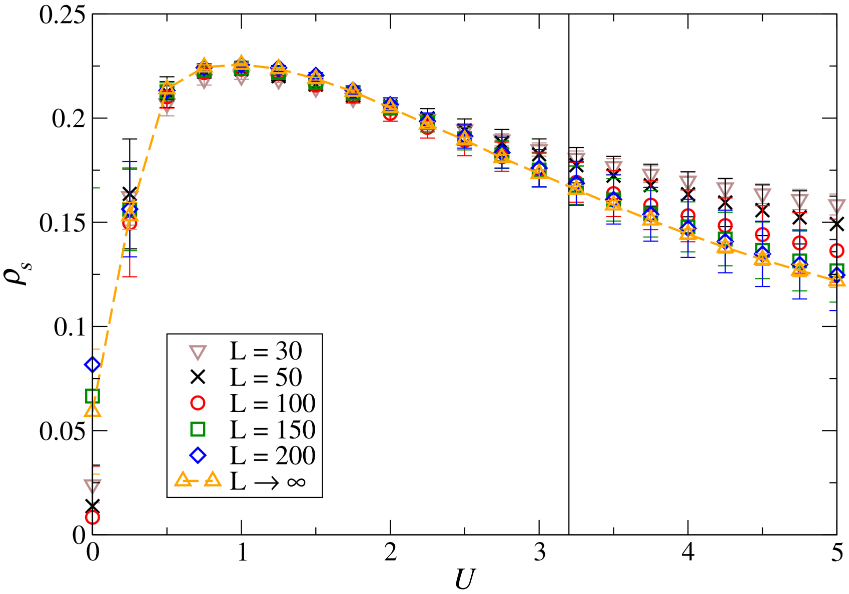

Figure 3.10: Superfluid fraction $\rho_s(U)$ for $N/L = 1/2$ with $\Delta\mu = 0.5$ for lengths $50 - 200$. The vertical line indicates $U_c = 3.09$. The inverted triangles give the finite-size scaled $L \to \infty$ limit with the dashed line a guide to the eye.

Performing a finite-size scaling analysis for K as shown in fig. 3.10(a) we find the U values, for which $K \equiv 3/2$ in the limit $L \to \infty$, converge towards a limiting value of $U_c = 3.09 \pm 0.01$ (see also fig. 3.8). This again indicates that in an infinite system, we expect the superfluid to Bose-glass transition to take place at much lower values of U_c than observed for $L = 50$. The relevance of this result is of course that experimental realizations of the Bose-Hubbard model are typically in cold atom systems, which are limited to finite system sizes, currently a typical lattice dimension is $\sim 50 - 100$ sites [57]. For values of $\Delta\mu \gtrsim 1$, the situation is less severe and we see in fig. 3.8(a+b) that our entanglement-based measures again qualitatively agree with the Giamarchi-Schulz criterion, both for $L = 50$ and estimated via finite-size scaling at $L \to \infty$.

3.4.3 Density = 2

To the best of our knowledge, the phase diagram for $N/L = 2$ has not been shown before in the literature. Due to our numerical restriction of five bosons per site, this regime is close to the limit of what can be studied reliably, particularly for small U where the lack of strong repulsion implies that most of the bosons should occupy

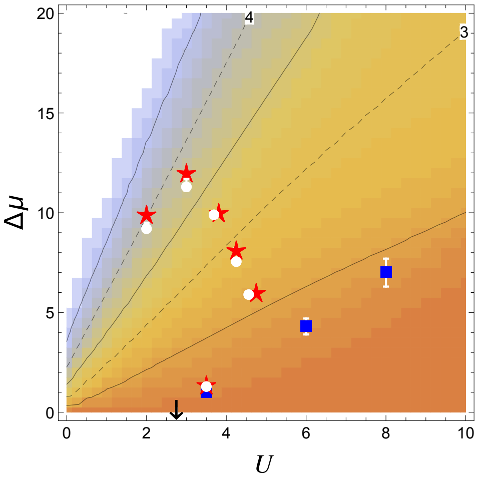

Figure 3.11: Contour plot showing average maximum site filling averaged over 100 samples. The maximum possible is limited at 5. The contour lines indicate $3, 4$ (dashed) and $2.5, 3.5, 4.5$ (solid). The results from Luttinger parameter analysis and energy gap are plotted with the same symbol convention as figs. 3.4, 3.8 and 3.12.

a small number of sites. This is illustrated by fig. 3.11, which shows that average maximum occupancy of a site is almost 5 for small U and large $\Delta\mu$ but is lower for much of the rest of the phase diagram. For large U, one might expect that we will have a Mott insulator of boson *pairs*, while a superfluid of boson pairs emerges for small U and small $\Delta\mu$. Similarly, we envisage a disordered Bose glass phase for large $\Delta\mu$. With more particles per site than in the $N/L = 1$ case, we could furthermore expect that the Mott transition at $\Delta\mu = 0$ happens at larger values of U, since the energy cost of avoiding a two-boson per site state is substantially larger. Similarly, the cost for two boson pairs to go onto the same site is $2U$, hence we expect the $2U = \Delta\mu/2$ line to characterize the superfluid phase as in the $N/L = 1$ case. In addition, one might conjecture to see a remnant of the $U = \Delta\mu/2$ condition.

In fig. 3.12, we show that our expectations are largely validated. Of particular interest is that a double lobe shape for the superfluid phase emerges and allows a possible re-entrant behaviour given a suitable cut across parameter space. The gradient of the Mott insulating phase boundary is shallower ($\sim 4/3$) when compared to $N/L = 1$. Furthermore, both the ζ and $S_{\mathrm{A|B}}$ based entanglement measures, as well as their errors, $\Delta\zeta$ and $\Delta S_{\mathrm{A|B}}$, capture the phases equally well and agree with the K and E_g estimates. Note that for $N/L = 2$, the KT superfluid-to-Mott transition at $\Delta\mu = 0$ corresponds to $K = 2$ and we use finite-size scaling for the

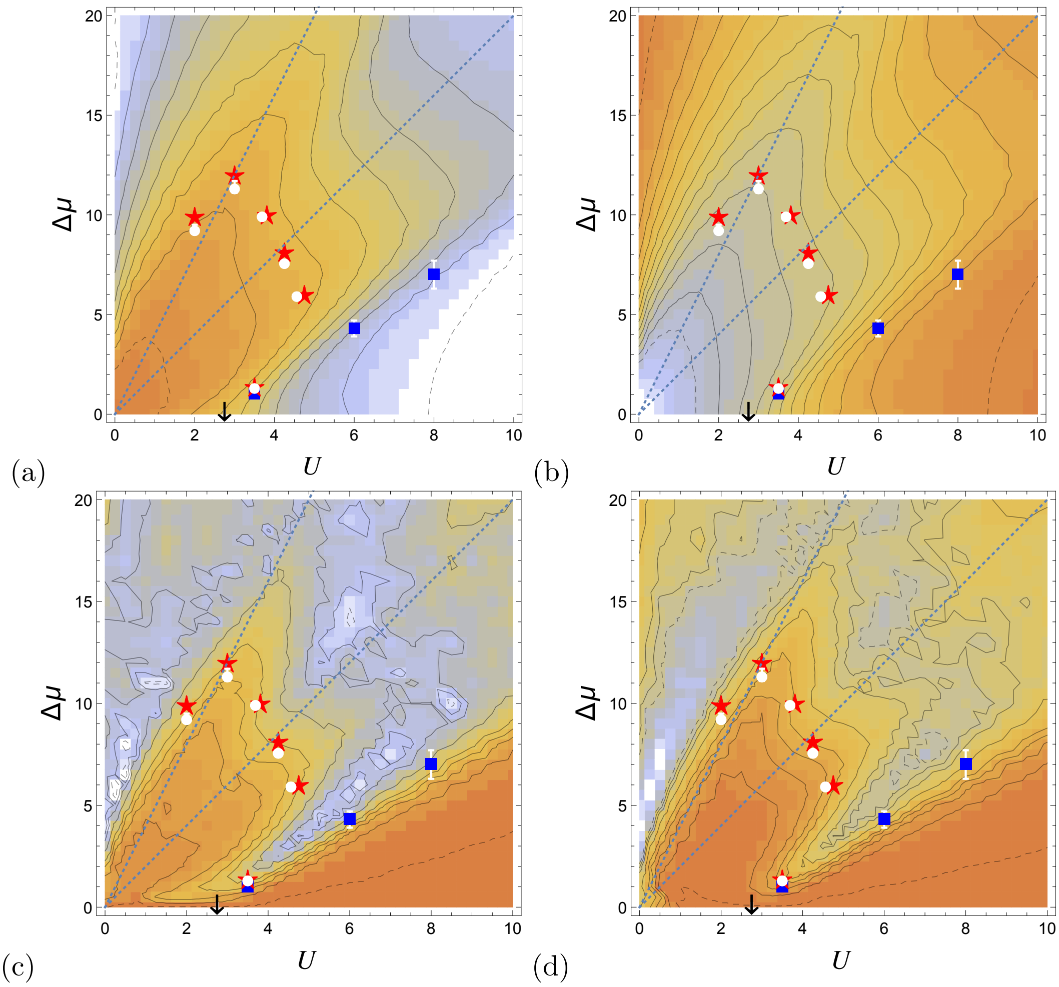

Figure 3.12: Phase diagrams for the disordered Bose-Hubbard model at $N/L = 2$ given as contour plots of (a) ζ, (b) $S_{A|B}$, (c) $\Delta\zeta$ and (d) $\Delta S_{A|B}$. Colors, symbols and lines (solid and dashed) denote corresponding estimates as in figs. 3.4 and 3.8. The contour lines correspond to (a) $\zeta = 0.1, 0.2, \ldots, 0.9$, (b) $S_{A|B} = 0.2, 0.3, \ldots, 1.4$, (c) $\Delta\zeta/10^{-4} = 0.1, 1, 2, 3, 4, 6, 8, 9, 10$ and (d) $\Delta S_{A|B}/10^{-4} = 0.1, 1, 2, 3, 4, 6, 8, 9, 10$. The two dotted straight lines indicates $\Delta\mu = 2U$ and $4U$.

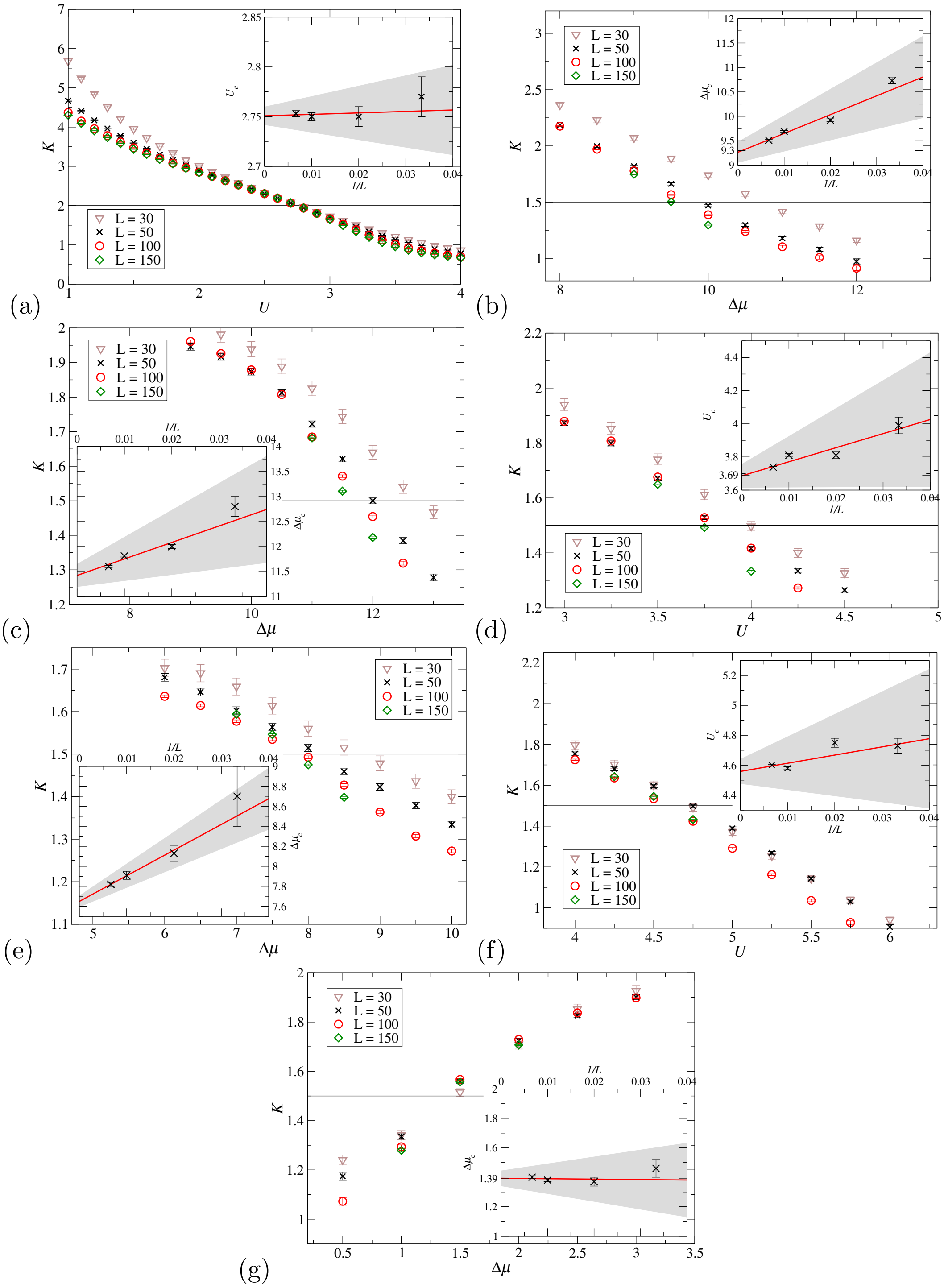

Figure 3.13: Luttinger parameter K for various lengths 30–150 at $N/L = 2$ for the points shown by white circles on the phase diagrams of fig. 3.12 at (a) $\Delta\mu = 0$, $U = 1, 1.1, \ldots, 4$; (b) $U = 2$, $\Delta\mu = 8, 8.5, \ldots, 12$; (c) $U = 3$, $\Delta\mu = 9, 9.5, \ldots, 13$; (d) $\Delta\mu = 10$, $U = 3, 3.25, \ldots, 4.5$; (e) $U = 4.25$, $\Delta\mu = 6, 6.5, \ldots, 10$; (f) $\Delta\mu = 6$, $U = 4, 4.25, \ldots, 6$; (g) $U = 3.5$, $\Delta\mu = 0.5, 1, \ldots, 3$. As with fig. 3.5, the horizontal line highlights $K = 3/2$ for the superfluid to Bose-glass transition with disorder (b-g) and $K = 2$ for the superfluid to Mott-insulator transition when clean (a). Again, the inset shows the finite size scaling analysis to obtain the critical point in the thermodynamic limit.

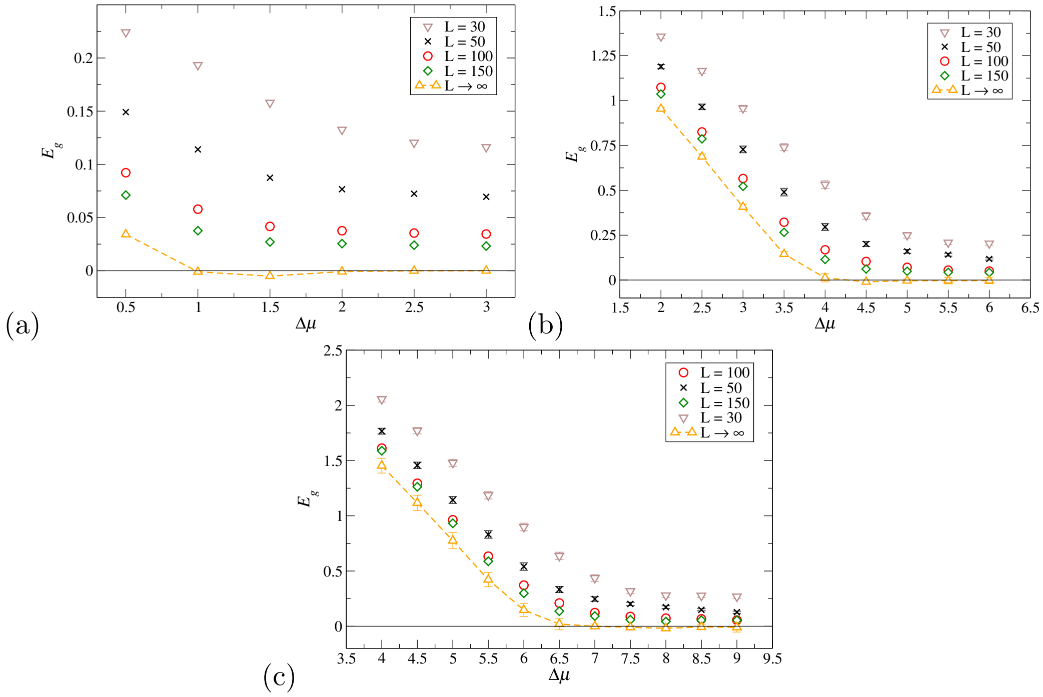

Figure 3.14: Energy gap E_g for various lengths 30–150 and the infinite size limit at $N/L = 2$ for the points shown by blue squares on the phase diagrams of fig. 3.12 at (a) $U = 3.5$, $\Delta\mu = 0.5, 1, \ldots 3$; (b) $U = 6$, $\Delta\mu = 2, 2.5, \ldots, 6$; (c) $U = 8$, $\Delta\mu = 4, 4.5, \ldots, 9$. The horizontal line highlights $E_g = 0$ where the gap closes indicating a Bose-glass phase and the orange dashed line is a guide to the eye. Error bars have been omitted where smaller than symbol size.

Luttinger parameter to find $U_c = 2.75 \pm 0.03$ as shown in fig. 3.13(a).

3.5 Conclusion

We have analysed the phase diagrams of the disordered Bose-Hubbard model for fillings $N/L = 1/2$, 1 and 2 using the entanglement-based measures ζ, $S_{\mathrm{A|B}}$, $\Delta\zeta$ and $\Delta S_{\mathrm{A|B}}$. We found that these measures are an excellent means of quickly identifying the different phases of the system while reducing the number of DMRG runs per measurement or removing the need for special boundary conditions. However, ζ and $S_{\mathrm{A|B}}$ alone did not always faithfully reproduce the phase diagrams and were sometimes misleading regarding the positions of the phase boundaries. The error-based measures, $\Delta\zeta$ and $\Delta S_{\mathrm{A|B}}$, provide a much clearer picture — the distributions of the values contain more information regarding the nature of the phase than the mean values alone.

For $N/L = 1$ the phase diagram is as calculated in previous papers [66, 70, 72] using K, E_g and ρ_s, providing strong support for the method based on using ζ, $S_{\mathrm{A|B}}$, $\Delta\zeta$ and $\Delta S_{\mathrm{A|B}}$. For $N/L = 1/2$ the diagram shows strong finite size effects and the critical U defined by the Giamarchi-Schulz criterion is not apparent for these finite systems. Finally for $N/L = 2$ the superfluid phase has a *double-lobed* appearance giving rise to re-entrance phenomena.

Chapter 4

General Tensor Networks

4.1 Success and Failure of DMRG

When performing DMRG simulations the modelling of some systems is far more successful than for others. For one dimensional systems away from criticality DMRG requires a much smaller χ than for critical systems to obtain the same accuracy [9, 19, 87, 88, 89]. The reason relates to the fact that away from criticality systems are *gapped*, that is, in the limit of infinite systems, have a largely discrete low energy spectrum with a non-zero energy difference between the ground and first excited state [90]. For gapped phases, when diagonalising the density matrix for White-style DMRG, or performing the SVD for variational MPS, the magnitudes of the eigenvalues (singular values) decrease exponentially as a function of eigenvalue (singular value) number [19]. Thus when selecting a limited number of eigenvalues (singular values) very little weight is discarded as the higher values do not significantly contribute to the overall wavefunction [87].

As an example, fig. 4.1 shows the singular values from an SVD at the centre of an MPS obtained from a DMRG simulation of the Bose-Hubbard model as discussed in chapter 3. The system size is 50 sites with $\chi = 200$ and the different symbols are different values of the Hubbard U. For $U \lesssim 1.6$ the system is in a superfluid state and is critical; for $U \gtrsim 1.6$ the system is in a Mott insulating state. When $U = 5$ the system is far away from criticality and as such the singular values decrease exponentially. The blue circles for $U = 2.5$ are for a gapped system but closer to the critical point and as such have larger contributions from higher singular value number.

Critical systems are also referred to as *gapless* systems due to the fact that in the thermodynamic limit the energy spectrum is largely continuous. For finite lattice

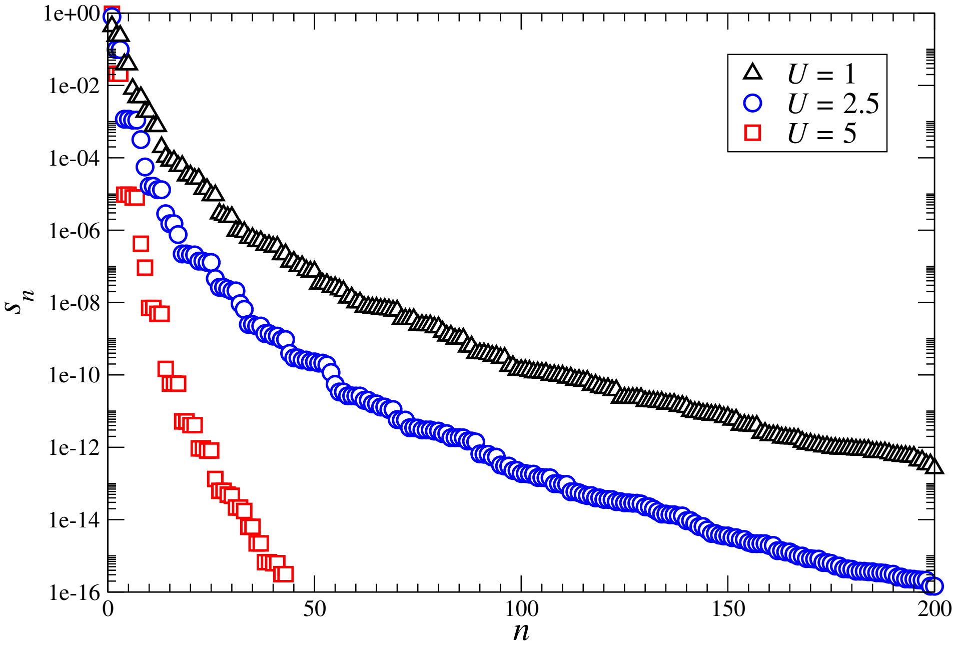

Figure 4.1: Values, s_n, of the singular values for the centre of a DMRG simulation of the Bose-Hubbard model. The system is 50 sites long with $\chi = 200$ and the different symbols are various values of Hubbard U.

systems the continuity of the energy spectrum is not obvious. The fact that there are only a finite number of eigenvalues in the spectrum means that there necessarily will be a non-zero energy difference between ground and first excited states. It is only in the infinite system limit that the gap closes [90]. The distribution of density matrix eigenvalues (singular values) in DMRG is also different for these critical systems. The eigenvalues (singular values) decrease in magnitude *polynomially*, which puts far more weight in the tails of the distribution [88, 89]. This is shown by the example of the Bose-Hubbard model in fig. 4.1. The magnitudes of the singular values for the black triangles at $U = 1$ decrease much more slowly than the red squares at $U = 5$. Thus when taking only a finite number of eigenvectors in the DMRG algorithm, more information is discarded, resulting in poorer performance. Note also that the step-like structure is due to symmetries in the Hamiltonian.

When insufficient χ is used it is not just the energy that is not correctly represented; many of the physical observables will not be correctly modelled. For example, for critical systems the two-point correlation function has a power law decay when plotted with respect to separation of points as opposed to an exponential decay for gapped states. This is illustrated by fig. 4.2(a), which shows the $\langle b_{x_1}.b^\dagger_{x_2}\rangle$ correlation function when averaged over all pairs of x_1 and x_2 for the Bose-Hubbard

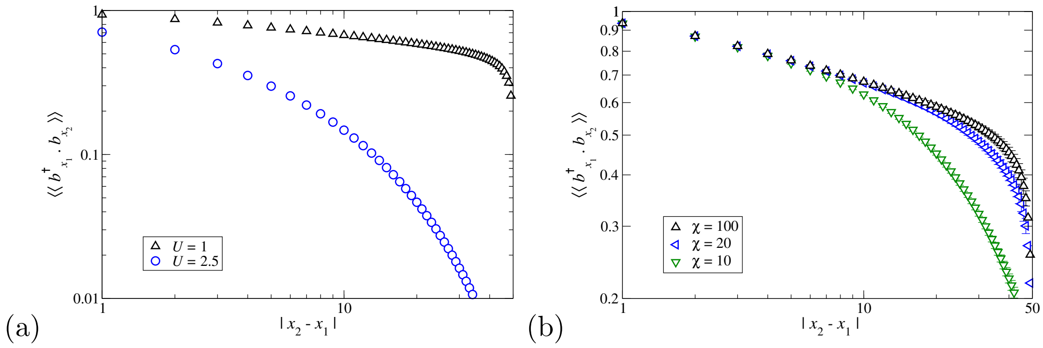

Figure 4.2: The $\langle b_{x_1}.b^\dagger_{x_2}\rangle$ correlation function averaged over all x_1 and x_2 separated by $|x_1 - x_2|$. (a) An example of the correlation function for a critical system where $U = 1$ (triangles), and a gapped state where $U = 2.5$ (circles) with $\chi = 200$. (b) Comparing different bond dimensions used for the DMRG of a system of 50 sites with $U = 1$. Different symbols represent different bond dimension, where $\chi > 100$ does not affect the form of the correlation function.

model in a critical superfluid ($U = 1$) and gapped Mott insulator ($U = 2.5$). When χ is too small the correlation will be approximately the power law for some length scale and decay exponentially for large separations. This length scale is increased by imposing a larger χ as shown in fig. 4.2(b) for the $U = 1$ superfluid phase of the Bose-Hubbard model.

Two dimensional systems have been attempted using DMRG from early on in its development [91, 92, 93], but with limited success. The distribution of density matrix eigenvalues (singular values) decreases polynomially but much slower than in the critical case [89, 94]. Thus the use of DMRG in two dimensions is restricted to small systems as the number of states that have to be kept increases exponentially with system size [9]. There has been some success by imposing PBCs on one dimension and keeping its size small, effectively making a quasi-one-dimensional cylinder [95, 96]. Increasing the cylinder circumference requires a large increase in χ, for example a circumference of 17.3 sites was achieved by Depenbrock et. al. [95] at a cost of $\chi = 16000$.

4.2 The Area Law for Entanglement Entropy

The observations described in section 4.1 are quantified by the *area law* for entanglement entropy. From first glance one would assume that the amount of entanglement for some region A with respect to B would be proportional to the volume of A.

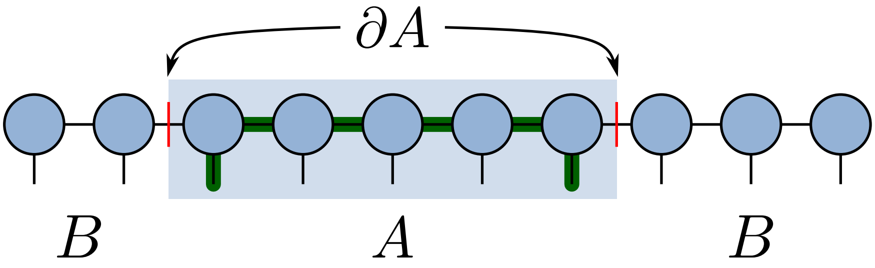

Figure 4.3: For an MPS a region A is bounded by two bonds, denoted by red lines. If the size of A is increased the number of bonds stays at 2. The bold green line highlights the minimum path between two sites.

This is true of most states of most systems, but it was found that for the ground state of gapped quantum systems the entanglement, quantified by the von Neumann entropy (eq. (2.4.4)), scales as the boundary to this region [97].

For tensor networks the *boundary* is quantified by the number of bonds n_A that connect region A to the environment. The reasoning behind this measure [44] is that if all of the tensors are identical, with a bond dimension χ the maximum contribution to the entanglement entropy per bond is $\log_2(\chi)$. Evenbly and Vidal [44] go further to suggest that for most cases of homogeneous tensor networks the entanglement per bond is approximately 1, hence $S_{A|B} \approx n_A$. This gives a reason for the excellent scaling for DMRG in one dimensional gapped systems. The boundary of a region on a one dimensional system is simply two points and does not increase when the region is expanded. An MPS has these same properties; the number of bonds that one would have to cut to separate region A from the environment (B) is 2 and does not change if the size of A is altered, as shown in 4.3. The fact that the MPS has the same entanglement properties as the ground state of a gapped one dimensional system makes it an ideal variational ansatz for such problems.

The area law also explains the poor performance of DMRG for critical and two dimensional systems. For critical systems the entanglement entropy scales logarithmically with the region size ($S_{A|B} \propto \log(L)$) [98, 99], hence the bond dimension required for accurate DMRG increases with system size. For two dimensional systems, take a square lattice with a square region within it. If the region has side length L the area law suggests that the entanglement entropy should scale as its boundary $S_{A|B} = 4L \propto L$, thus the MPS is insufficient as a variational ansatz for the two dimensional system. A more appropriate, area law conserving ansatz would be a tensor network where all sites are connected to their four neighbours to match the lattice geometry, as shown in fig. 4.4. This tensor network is known as a projected entangled pair state (PEPS) and is the natural two dimensional extension of the

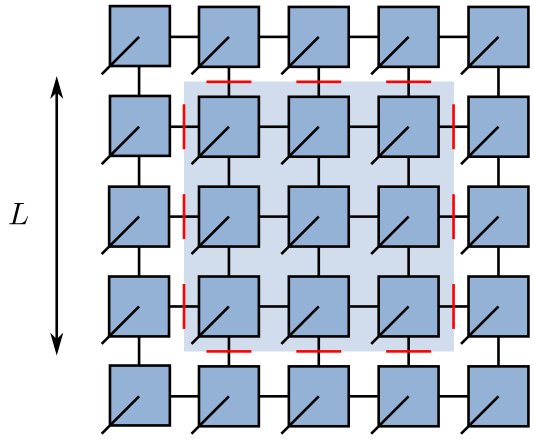

Figure 4.4: A PEPS on a square lattice, where the PEPS tensors are blue squares, the virtual indices are vertical and horizontal, and the physical indices are those pointing diagonally down. As before, the region under consideration is highlighted in blue and the bonds that separate it from the rest of the system are shown by red lines. Here the boundary scales as $4L$ and is therefore area law conserving.

MPS [35, 47, 100, 101].

Although PEPS is area law satisfying and therefore a more appropriate ansatz for a two dimensional system, its advantages come at a cost. Contracting a bra and ket PEPS state is a problem that scales exponentially with system size and there is no canonical form that gauges away the problem like in MPS [2]. Therefore to perform an update it is necessary to approximate the environment and compress it into an MPO. Even so, performing an update to a site using the traditional method [102] costs $O(\chi^{10})$. As a result, PEPS is restricted to small system sizes and χ. For example ref. [102] look at a 20×20 lattice with $\chi = 6$. Other methods have pushed this to $\chi = 60$ for a lattice on a semi-infinite cylinder [103]. Despite this large computational cost and small possible bond dimension, PEPS has proved a valuable numerical [103, 104] and analytical tool [105].

Using DMRG and PEPS as variational ansätze relies on the algorithms being able to *find* the ground state in a reasonable amount of time; that this is possible is not necessarily expected. The Hilbert space for many-body systems is incredibly large as it scales exponentially with lattice size, thus the number of possible states to search through to find the ground state is equally expansive. The fact that for a large number of quantum many-body systems the ground state is known to have be area law conserving rather than volume-law makes the problem more tractable.

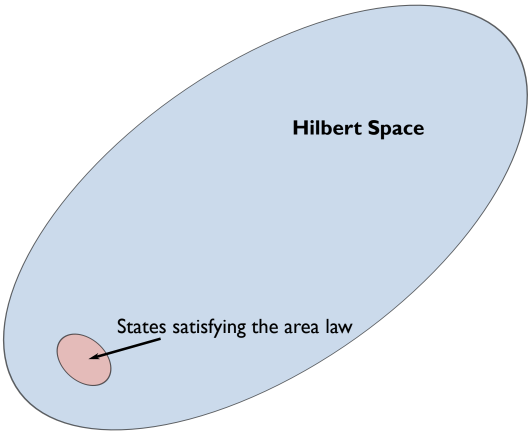

Figure 4.5: Diagrammatic representation of the exponentially large Hilbert space and the small *corner* of the space that area law conserving states occupy.

Setting the geometry of the tensor network such that it is area law conserving, such as using an MPS or PEPS, constrains the search to a small *corner* of Hilbert space [2, 4], as shown in fig. 4.5. This corner turns out to be exponentially smaller than the full Hilbert space [3], allowing algorithms to efficiently find the ground state. Increasing bond dimension χ expands this small corner, allowing states with larger than area law entanglement to be simulated. Indeed, in the limit of $\chi \to \infty$ all of Hilbert space is accessible [2].

4.3 Beyond the Area Law

There is currently a lot of excitement around the so-called AdS/CFT correspondence and possible applications in condensed matter physics [106]. AdS/CFT is most well known in high energy physics where it was noted [107] that there exists a duality between certain theories of gravity on $d+2$ dimensional Anti-de Sitter (AdS) spacetime and CFTs living on its $d+1$ dimensional boundary. In condensed matter systems, AdS/CFT can provide a geometric interpretation of renormalization group (RG) techniques since the additional *holographic* dimension can be interpreted as a scale factor in the RG coarse graining [106].

The relation to entanglement properties of quantum many body systems has its roots in the theory of black hole thermodynamics. It was found [108, 109, 110] that black holes obey thermodynamics and have a thermal entropy that scales with the surface area of the black hole. This resulted in the famous *Bekenstein-Hawking*

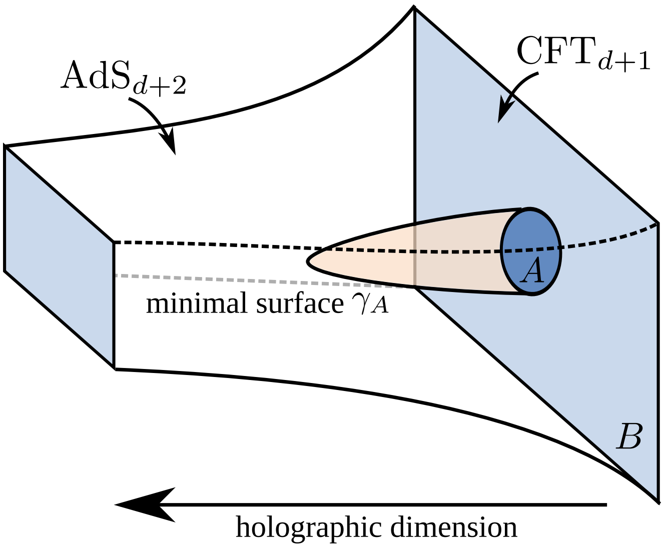

Figure 4.6: A diagrammatic representation of the AdS/CFT correspondence, showing a $(d+1)$ dimensional CFT on the boundary of a $(d+2)$ dimensional AdS spacetime. The entanglement entropy of a region A of the CFT is proportional to the minimal surface γ_A that separates A from the remainder of the CFT.

formula [111]

$$S_{BH} = \frac{\mathcal{A}_h}{4G_N}, \tag{4.3.1}$$

where $\mathcal{A}_h$ is the area of horizon, G_N is Newton's gravitational constant and BH either stands for *Bekenstein-Hawking* or *black hole*.

The ideas were extended to entanglement entropy in the context of AdS/CFT by Ryu and Takayanagi [111]. AdS/CFT suggests a duality between a $(d+1)$ dimensional CFT on the boundary of AdS$_{d+2}$ and a quantum gravity in the bulk. The entanglement entropy of a region A of the CFT is related to the size of the surface with smallest area, or *minimal surface* γ_A, within the AdS bulk that separates A from the rest of the system, rather than the boundary within the CFT as suggested by the area law. This is shown pictorially in fig. 4.6 where the holographic dimension is interpreted as the renormalisation scale. The result is

$$S_{A|B} = \frac{\mathcal{A}_{\gamma_A}}{4G_N^{(d+2)}}, \tag{4.3.2}$$

where $\mathcal{A}_{\gamma_A}$ is the area of the minimal surface γ_A and $G_N^{(d+2)}$ is Newton's constant in $d+2$ dimensions.

It has been argued recently [44, 45, 112] that tensor networks, particularly the multi-scale entanglement renormalisation ansatz (MERA) [34, 42, 113], can be a coarse grained embodiment of AdS/CFT. The structure of MERA, shown in fig. 4.7, is made up of *disentanglers* (green squares) and *isometries* (pink triangles) [34]. The intuitive argument behind its construction is that the disentanglers remove entanglement so that the isometries can then remove degrees of freedom that are no longer coupled to the system. The network is self-similar in the way that at each level of coarse graining the network looks the same and it is the direction of coarse graining, perpendicular to the physical lattice, that is the extra *holographic* dimension.

Just as with MPS and PEPS and the area law, the minimal surface γ_A is found by counting the minimum number of bonds that have to be cut to separate one region from the rest of the system. Take, for example, a region comprised of L sites within a MERA as shown in fig. 4.7. The minimum number of bonds (red lines in fig. 4.7) $n_A \approx \log(L)$, which matches the entanglement scaling of critical systems [44]. As an extension of this, it was shown [114, 115, 116] that in the continuum limit of MERA it has a metric that matches the properties of AdS/CFT.

In addition to entanglement entropy, similar arguments hold for two-point correlation functions [44, 112]. The asymptotic scaling of correlation functions should be

$$C_{\mathrm{TN}}(x_1, x_2) \sim e^{-\alpha D_{\mathrm{TN}}(x_1,x_2)}, \tag{4.3.3}$$

where D_{TN} is the path with the minimum distance or *geodesic* connecting points x_1 and x_2. For MPS $D_{\mathrm{MPS}} = |x_2 - x_1|$, as shown as the bold line in fig. 4.3. Hence

$$C_{\mathrm{MPS}}(x_1, x_2) \sim e^{-\alpha D_{\mathrm{MPS}}(x_1,x_2)} = e^{-|x_2-x_1|/\xi}, \tag{4.3.4}$$

where ξ is the correlation length, matching the exponential decay observed in fig. 4.2(b) when χ is insufficient. MERA, on the other hand has path lengths that scale logarithmically with separation of x_1 and x_2 ($D_{\mathrm{MERA}}(x_1, x_2) \propto \log_2(L)$), as shown in fig. 4.7. Thus

$$C_{\mathrm{MERA}}(x_1, x_2) \sim e^{-\alpha D_{\mathrm{MERA}}(x_1,x_2)} = |x_2 - x_1|^{-q}, \tag{4.3.5}$$

recovering the power law decay profile characteristic of critical systems.

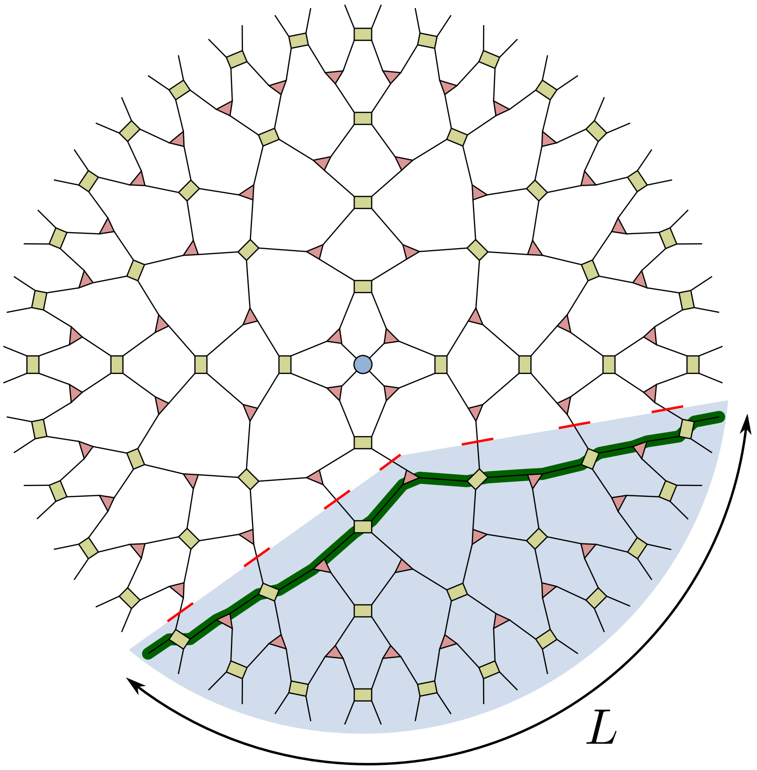

Figure 4.7: Tensor network diagram of a MERA with PBCs where the green squares are *disentanglers*, pink triangles are *isometries* and the blue circle is the top tensor. A region of the network corresponding to L sites is highlighted in blue, where the bonds making up the minimum surface are highlighted by the red lines. The bold green line highlights a *geodesic* connecting two points on the lattice.

Chapter 5

Tensor Network Strong Disorder Renormalisation

5.1 Introduction

For *disordered* quantum many-body systems, the strong-disorder renormalisation group (SDRG) provides a powerful and elegant means of analysing a system by concentrating principally on the disorder within it [117]. The approach was originally devised by Ma, Dasgupta and Hu (MDH) [118, 119] for the random anti-ferromagnetic (AFM) Heisenberg chain

$$H = \sum_{i=1}^{L-1} J_i \vec{s}_i \cdot \vec{s}_{i+1}, \tag{5.1.1}$$

which is the same as eq. 2.6.22 but the coupling constant J_i is different for each position, taking a random value $0 < J_i < J_{\text{max}}$ according to some probability distribution $P(J)$. The principle behind the SDRG is to eliminate the most strongly coupled pairs of spins and replace them with an effective interaction that couples the spins at either side of the pair, as shown in fig. 5.1(a). The pair of spins coupled by J_{max} are thought of as being *frozen* into a singlet ground state as the neighbouring interactions are significantly weaker — ultimately leading to the random singlet phase, which is the ground state of the system [120, 121]. This freezing of degrees of freedom is remarkably close to an update process in entanglement RG for tensor networks [42] and also suggests the possible usefulness of AdS/CFT for disordered spin chains. By analysing the probability of *survival* through the SDRG algorithm it is possible to predict that mean correlations will have a power-law decay [120] with negative power 2. Similarly, the entanglement entropy can be shown to scale

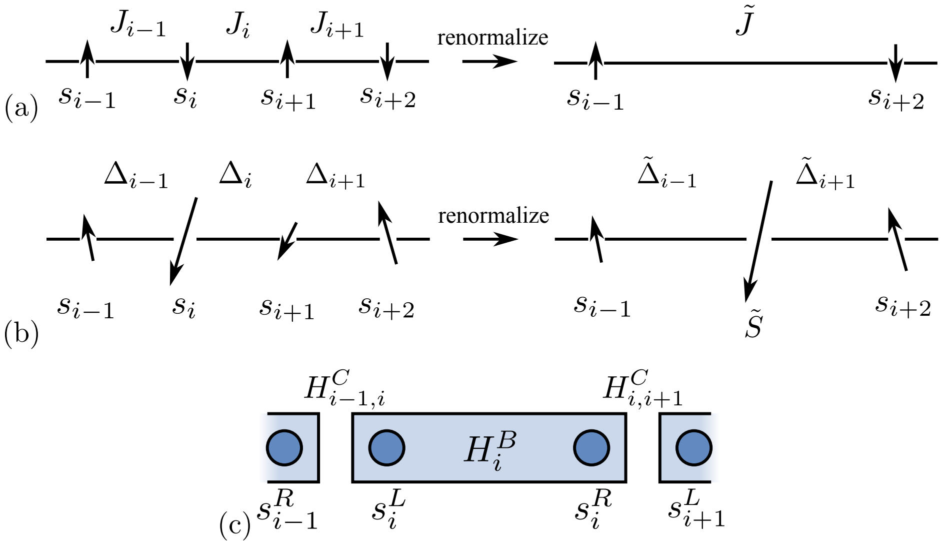

Figure 5.1: Schematic diagrams of the various SDRG variants. Horizontal lines indicate the 1D spin system. (a) Traditional MDH SDRG [118], spins $\vec{s}_i$, $\vec{s}_{i+1}$ (arrows) with the greatest coupling strength, $J_i > J_k \forall k \neq i$, are removed and replaced by an effective coupling $\tilde{J}$. (b) SDRG of Westerberg et. al. [123], spin pairs are renormalised for the largest energy gap Δ_i and replaced by an effective spin $\tilde{S}$. (c) SDRG variant of Hikihara et. al. [124], the chain is decomposed into blocks of spins described by block Hamiltonians H^B (shaded rectangles), with left and right spins, respectively, s^L and s^R (dark dots) on the boundaries of the blocks forming the coupling Hamiltonians, H^C.

logarithmically with block size [122], where the amount of entanglement between blocks A and B is quantified by the von Neumann entropy of eq. (2.4.4).

In this chapter, we develop a *self-assembling* TTN algorithm based on the previous ideas of SDRG [123, 124]. This allows us to calculate properties such as expectation values, correlation functions and entanglement entropy directly and efficiently from the geometry of the TTN. In particular, we find that the distance dependence of the spin-spin correlation function can be studied not only via direct calculation of the correlation functions, but also via the *holographic* distance dependence along the tree network connecting two sites. In section 5.2 we will briefly review the numerical SDRG of Hikihara et al. [124] and define the states and operators that form the basis for our work. Section 5.3 shows how the numerical SDRG on an MPO *self-assembles* the TTN. In section 5.4 we discuss the details of the algorithms used in the chapter. Finally, in section 5.5 we compute correlation functions and entanglement entropy (i) directly using the TTN as well as (ii) via simply counting the path lengths and connectivities in the holography. We find that both approaches give consistent results.

5.2 MPO Implementation of the SDRG

5.2.1 The Numerical SDRG

The SDRG method was extended to both ferromagnetic (FM) and AFM couplings by Westerberg et. al. [123, 125]. The approach finds the neighboring pair of spins $\vec{s}_i$, $\vec{s}_{i+1}$ with the greatest energy gap Δ_i between the ground state and excited state and combines them into a single effective spin $\tilde{S}$ (fig. 5.1(b)). The effective couplings between the new spin and its neighbours are then recalculated using Clebsch-Gordan coefficients and the new gaps $\tilde{\Delta}_{i-1}$ and $\tilde{\Delta}_i$ updated. SDRG was once more extended by Hikihara et. al. [124] to include higher states at each decimation, in the spirit of NRG [7] and DMRG [15]. This method therefore decomposes the system into blocks rather than larger spins allowing for more accurate computation of, e.g., the spin-spin correlation functions. The more states that are kept at each decimation the more accurate the description and it is exact in the limit of all states kept.

Consider a point in the algorithm where the Hamiltonian is made up of blocks H^{B}_i at each site and couplings $H^{\mathrm{C}}_{i,i+1}$ between them as in fig. 5.1. The couplings take the form of a two spin Hamiltonian

$$H^{\mathrm{C}}_{i,i+1} = J_i \vec{s}^{\,\mathrm{R}}_i \cdot \vec{s}^{\,\mathrm{L}}_{i+1}, \tag{5.2.1}$$

where $\vec{s}^{\,\mathrm{R}}_i$ is the spin operator of the right hand spin of block i and $\vec{s}^{\,\mathrm{L}}_{i+1}$ is the left hand spin of block $i+1$. In full the Hamiltonian is

$$H = \sum_{i=1}^{N_{\mathrm{B}}} H^{\mathrm{B}}_i + \sum_{i=1}^{N_{\mathrm{B}}-1} H^{\mathrm{C}}_{i,i+1}, \tag{5.2.2}$$

where N_{B} is the number of blocks.

Let us now define the gap Δ_i as the energy difference between the highest-energy SU(2) multiplet that would be kept and the lowest-energy multiplet that would be discarded in a renormalisation of block $H^{\mathrm{B}}_{i,i+1}$. The scheme works by searching for the pair of blocks with the largest gap Δ_{i_m} and then combines the coupling and the blocks that it connects into a single block

$$H^{\mathrm{B}}_{i_m,i_m+1} = H^{\mathrm{B}}_{i_m} + H^{\mathrm{C}}_{i_m,i_m+1} + H^{\mathrm{B}}_{i_m+1}. \tag{5.2.3}$$

This block and the couplings either side are then renormalised by a matrix, V_χ, of the eigenvectors corresponding to the lowest χ eigenvalues of the block, such that only full SU(2) blocks are kept. The process is repeated until the system is represented by one block. The explicit form of the algorithm is as follows [124]:

1. Find the coupling Hamiltonian with the largest gap Δ_{i_m} and create the two-site block.

2. Diagonalise the two-site block to find the $\chi \leq \chi'$ lowest eigenvalues Λ_χ and corresponding eigenvectors V_χ such that only full SU(2) multiplets are kept, where χ' is the maximum number of eigenvectors and is set at runtime.

3. Set the χ eigenvalues Λ_χ from the diagonalisation as the new two-site block, which is equivalent to renormalising the two site block with V_χ

$$\tilde{H}^{\mathrm{B}}_{i_m,i_m+1} = V_\chi^\dagger H^{\mathrm{B}}_{i_m,i_m+1} V_\chi = \Lambda_\chi. \tag{5.2.4}$$

4. Renormalise the spin operators on the right and left hand side of the new block to update the couplings

$$\begin{aligned}\tilde{\vec{s}}^{\,\mathrm{R}}_{i_m} &= V_\chi^\dagger (\mathbb{1} \otimes \vec{s}^{\,\mathrm{R}}_{i_m+1}) V_\chi \\ \tilde{\vec{s}}^{\,\mathrm{L}}_{i_m} &= V_\chi^\dagger (\vec{s}^{\,\mathrm{L}}_{i_m} \otimes \mathbb{1}) V_\chi.\end{aligned} \tag{5.2.5}$$

5. Diagonalise the neighbouring blocks to get the new gaps.

6. Remove site $i_m + 1$ and return to step 1.

This process is repeated until the whole system is described by one block.

5.2.2 Numerical SDRG as an MPO Process

Hikihara's numerical SDRG can be naturally described as a set of operations on an MPO (see chapter 2).

1. First, we contract the MPO tensors for the pair of sites with the largest gap, sites i_m and $i_m + 1$ (fig. 5.2(a)

$$W^{[i_m,i_m+1]} = \sum_{b_{i_m}} W^{\sigma_{i_m},\sigma'_{i_m}}_{b_{i_m-1},b_{i_m}} W^{\sigma_{i_m+1},\sigma'_{i_m+1}}_{b_{i_m},b_{i_m+1}}. \tag{5.2.6}$$

   Here we have $\sigma_{i_m} = 1, \ldots, \chi$ for the *physical* indices and, for the Heisenberg model (5.1.1), the *virtual* indices are $b_{i_m} = 1, \ldots, 5$.

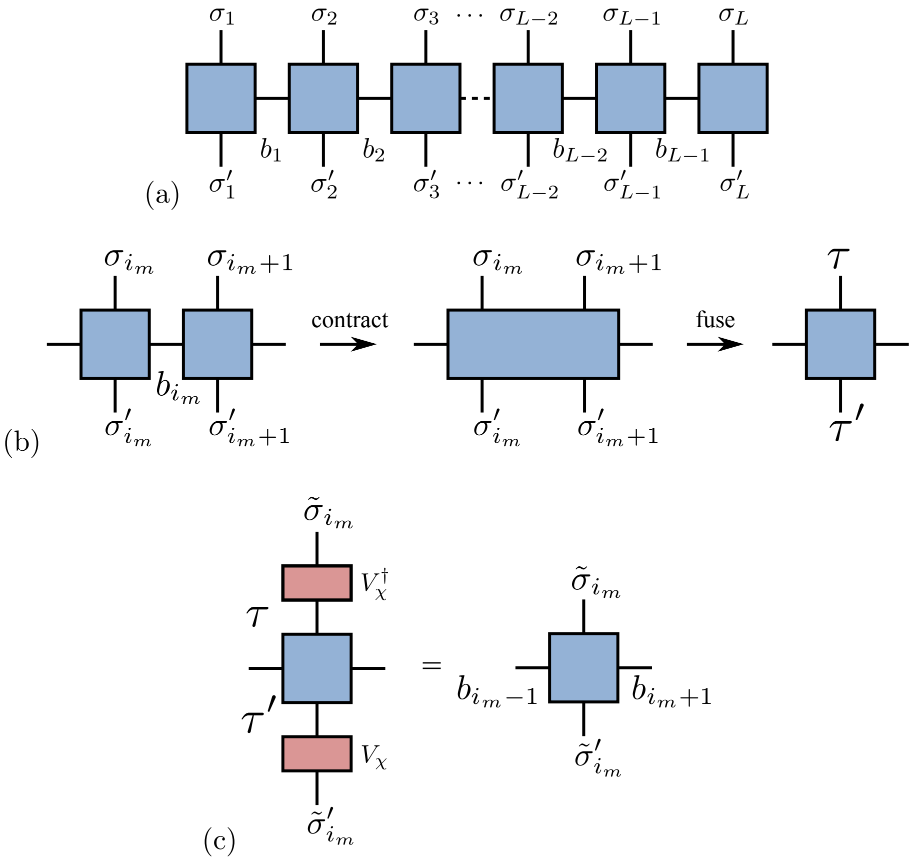

Figure 5.2: (a) Tensor network diagram of the MPO. The (vertical) σ and σ' legs denote physical indices and couple to the tensor network wavefunction and conjugate. The b's are virtual indices (in horizontal direction) and couple the local tensors (blue-shaded squares) of the MPO to each other. (b) The pair of sites with the largest gap Δ_{i_m} is found, the MPO tensors for these sites are contracted and the physical indices fused to form a matrix. (c) Contracting the matrices of eigenvectors V_χ (red-shaded rectangle) and $V_\chi^\dagger$ creates a new MPO for a coarse-grained system.

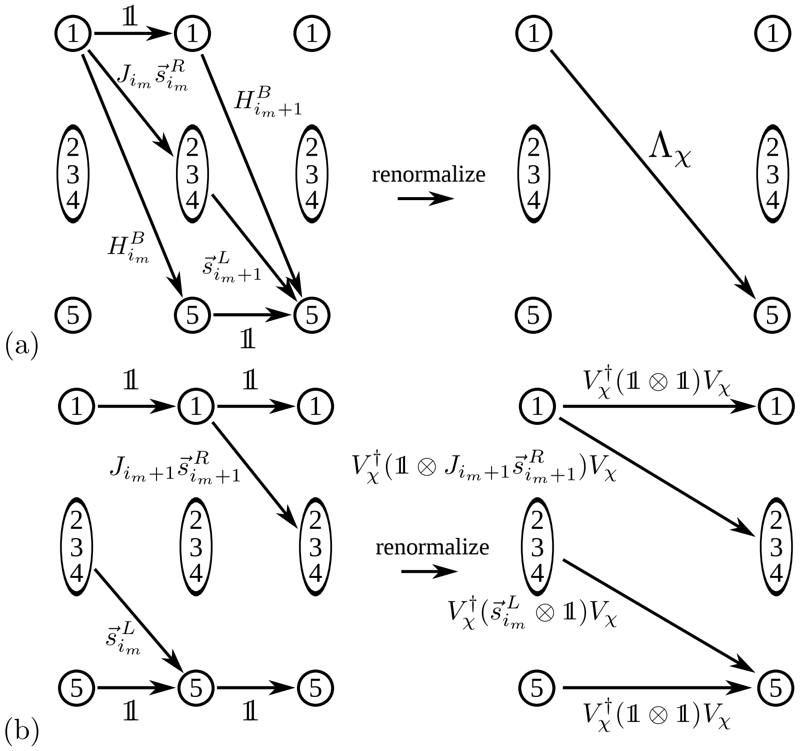
(a)
1
$\mathbb{1}$
$J_{i_m}\vec{s}^R_{i_m}$
$H^B_{i_m+1}$
2 3 4
renormalize
Λ_χ
$H^B_{i_m}$
$\vec{s}^L_{i_m+1}$
5
$\mathbb{1}$
(b)
$\mathbb{1}$
$\mathbb{1}$
$V^\dagger_\chi(\mathbb{1}\otimes\mathbb{1})V_\chi$
$J_{i_m+1}\vec{s}^R_{i_m+1}$
$V^\dagger_\chi(\mathbb{1}\otimes J_{i_m+1}\vec{s}^R_{i_m+1})V_\chi$
renormalize
$V^\dagger_\chi(\vec{s}^L_{i_m}\otimes\mathbb{1})V_\chi$
$\vec{s}^L_{i_m}$
$\mathbb{1}$
$\mathbb{1}$
$V^\dagger_\chi(\mathbb{1}\otimes\mathbb{1})V_\chi$

Figure 5.3: Matrix product diagram showing the contraction step (5.2.8) for the Heisenberg Hamiltonian (5.1.1). Circles (and ellipses) denote (combined) operator entries in the Heisenberg MPO $W^{[i,i+1]}$. (a) Renormalising the on-site components has the effect of creating a new on-site component, which is a diagonal matrix of the lowest eigenvalues Λ_χ. (b) Contracting V_χ and $V^\dagger_\chi$ has the effect of renormalising the coupling spins in the same way as the Hikihara method, storing them as the coupling components of the new MPO tensor.

2. Next, we perform an eigenvalue decomposition on the on-site components of the new MPO tensor keeping the eigenvectors of the lowest χ eigenvalues V_χ

$$\Lambda_\chi = V^\dagger_\chi(H^B_{i_m}\otimes\mathbb{1} + J_{i_m}\vec{s}^R_{i_m}\cdot\vec{s}^L_{i_m+1} + \mathbb{1}\otimes H^B_{i_m+1})V_\chi. \qquad (5.2.7)$$

   As with Hikihara's algorithm, only the χ eigenvalues that make up full SU(2) multiplets are used.

3. Then we contract V_χ and $V^\dagger_\chi$ with the new MPO tensor to perform the renormalisation (fig. 5.2(b). For the moment write the two-site combined MPO $W^{[i_m,i_m+1]}$ in terms of an effective site with index $\tau = 1,\ldots,\chi,\chi+1,\ldots,\chi^2$, i.e. $W^{\tau,\tau'}_{b_{i_m-1},b_{i_m+1}}$. Similarly, we can write the set of eigenvectors as $[V_\chi]^{\tilde{\sigma}_{i_m}}_\tau$.

Then the contraction is explicitly given as

$$W^{\tilde{\sigma}_{i_m},\tilde{\sigma}'_{i_m}}_{b_{i_m-1},b_{i_m}} = \sum_{\tau,\tau'}[V_\chi^\dagger]^{\tilde{\sigma}_{i_m}}_{\tau} W^{\tau,\tau'}_{b_{i_m-1},b_{i_m}} [V_\chi]^{\tilde{\sigma}'_{i_m}}_{\tau'}, \tag{5.2.8}$$

where $\tilde{\sigma}_{i_m} = 1, \ldots, \chi$ is the spin index of the renormalised site i_{m}. Hence we replace sites i_m and $i_m + 1$ with a single renormalised site and relabel the remaining indices. The contraction makes the on-site component of the new MPO simply a diagonal matrix of the lowest χ eigenvalues Λ_χ (fig. 5.3(a)). It also has the effect of renormalising the coupling spins just as in the Hikihara approach (fig. 5.3(b))

$$\vec{s}^{\,\mathrm{R}}_{i_m} = V_\chi^\dagger(\mathbb{1} \otimes \vec{s}^{\,\mathrm{R}}_{i_m+1})V_\chi, \tag{5.2.9}$$

$$\vec{s}^{\,\mathrm{L}}_{i_m} = V_\chi^\dagger(\vec{s}^{\,\mathrm{L}}_{i_m} \otimes \mathbb{1})V_\chi. \tag{5.2.10}$$

The contraction therefore maps two MPO tensors onto one while preserving the indexing structure of the MPO.

4. As the final step, we diagonalise the neighbouring blocks to update the distribution of gaps.

The procedure is then repeated until the system is just one site, and we diagonalise to obtain the ground state energy E_g of the system.

5.3 Tree Tensor Networks and SDRG

The MPO description of SDRG given above amounts to a coarse-graining mechanism that acts on the operator. Alternatively, we can view it as a multi-level tensor network wavefunction acting on the original operator. To illustrate this, we can *split* the τ index of $V_\chi^\dagger$ as in eq. (5.2.8) back to the original spin indices σ_{i_m}, σ_{i_m+1} to create an *isometric* tensor [34] or *isometry* $\left[V_\chi^\dagger\right]^{\tilde{\sigma}_{i_m}}_{\tau} \equiv [w]^{\tilde{\sigma}_{i_m}}_{\sigma_{i_m},\sigma_{i_m+1}}$. The isometric property means that

$$\sum_{\sigma_{i_m},\sigma_{i_m+1}} [w]^{\tilde{\sigma}_{i_m}}_{\sigma_{i_m},\sigma_{i_m+1}} [w^\dagger]^{\tilde{\sigma}'_{i_m}}_{\sigma_{i_m},\sigma_{i_m+1}} = \delta^{\tilde{\sigma}_{i_m},\tilde{\sigma}'_{i_m}}, \tag{5.3.1}$$

or $ww^\dagger = \mathbb{1} \neq w^\dagger w$ (fig. 5.4(a)). A renormalisation in the SDRG algorithm as in fig. 5.2(b+c) can then be rephrased graphically as in fig. 5.4(b). This makes the notion of mapping two MPO tensors to one immediately explicit.

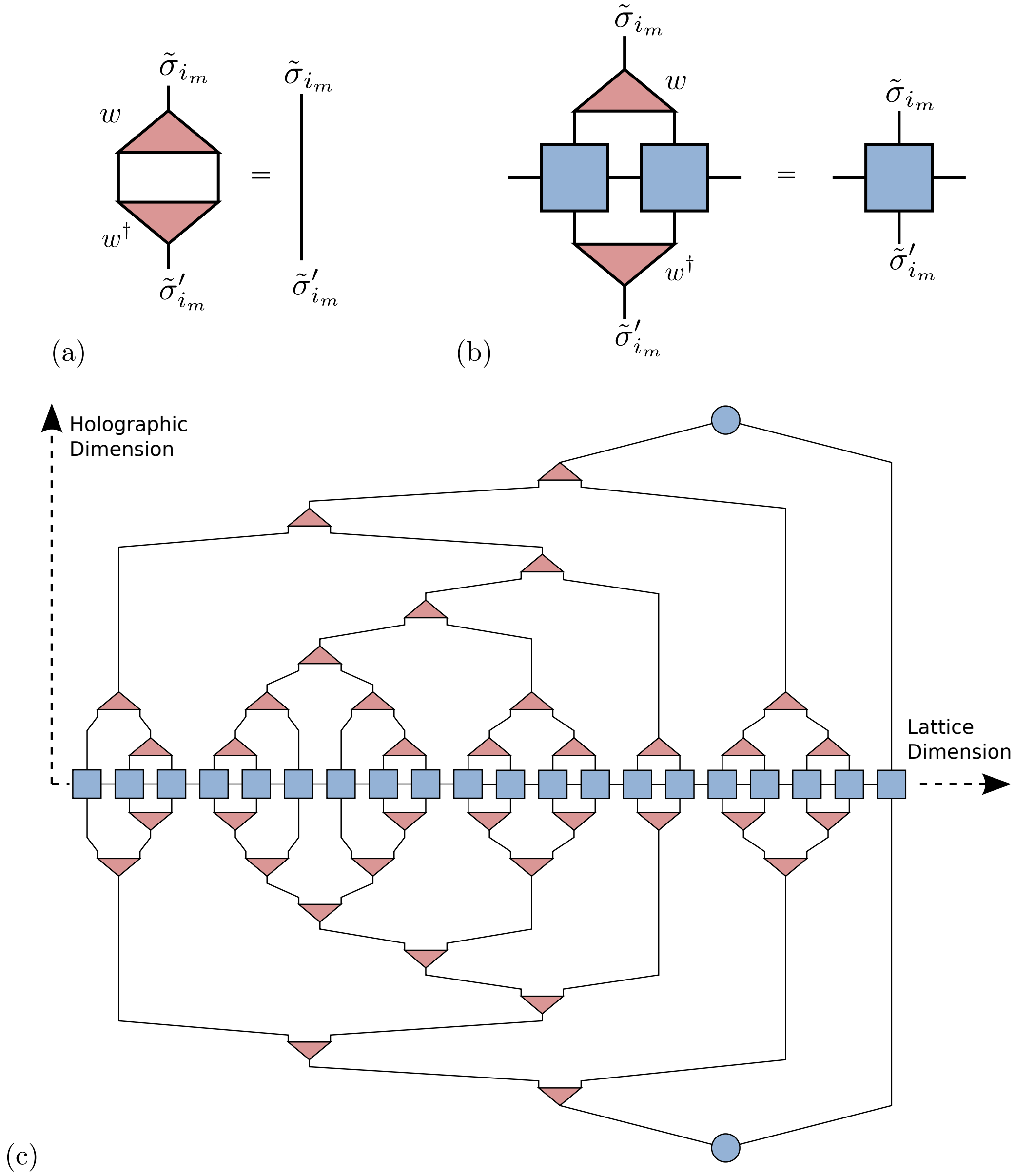

Figure 5.4: (a) Schematic representation of the isometric property $ww^\dagger = \mathbb{1}$ given by eq. (5.3.1). (b) One step in the MPO SDRG algorithm in terms of isometric tensors w. Triangles (red-shaded) denote the isometries, squares are as in fig. 5.2. (c) The SDRG algorithm as a TTN for a chain of $L = 20$ sites. The squares are the MPOs (i.e. the spin operators), triangles are isometric tensors and solid lines denote summations over physical (vertical) and virtual (horizontal) indices as before. The circle indicates the *top tensor*, i.e. the ground state eigenvector of the coarse-grained system. Lattice and holographic dimensions are indicated by the dashed arrows.

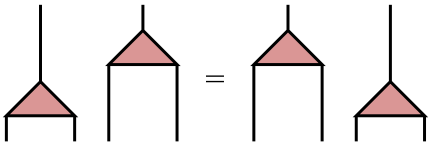

Figure 5.5: Isometries that do not share legs can be computed in any order.

When viewed in terms of isometries, the algorithm can be seen to self-assemble a tensor network based on the positions of largest gaps before each renormalisation. When written in full, it builds an inhomogeneous binary TTN as shown in fig. 5.4(c). We shall henceforth refer to this TTN approach to SDRG as *tSDRG*. Tree tensor networks are one of the major areas of tensor network research and TTNs with regular structures have been extensively studied [35, 43, 47, 126]. The isometric nature of the isometries allows for calculations to be performed in a highly efficient manner [34, 43, 127].

5.4 Algorithmic Detail

5.4.1 Indexing

The fact that the tensor network is built in the order of the SDRG algorithm means that its structure is not regular and is different for each disorder realisation. This means that we need an indexing convention that contains information about the structure of the network so that the isometries and density matrices can be contracted in the correct way. In order to calculate observables we need to know the order in which pairs of spins are renormalised and which tensors are connected to each leg of every isometry. This contains all of the structural information necessary to define the tensor network uniquely up to trivial commutations visualised in fig. 5.5. In this indexing scheme there are two relevant *dimensions*; The position on the lattice (l), drawn horizontally and the SDRG iteration number (i) drawn vertically with the start at the bottom and end at the top as shown in fig. 5.6. The process of indexing is as follows:

1. Begin the SDRG algorithm with iteration $i = 1$ and a temporary array (*current*) that contains information regarding the tensors that are *below* that point in the algorithm, initially L elements filled with zeros.

2. Store the position of the maximum J as element i in an array *order*.

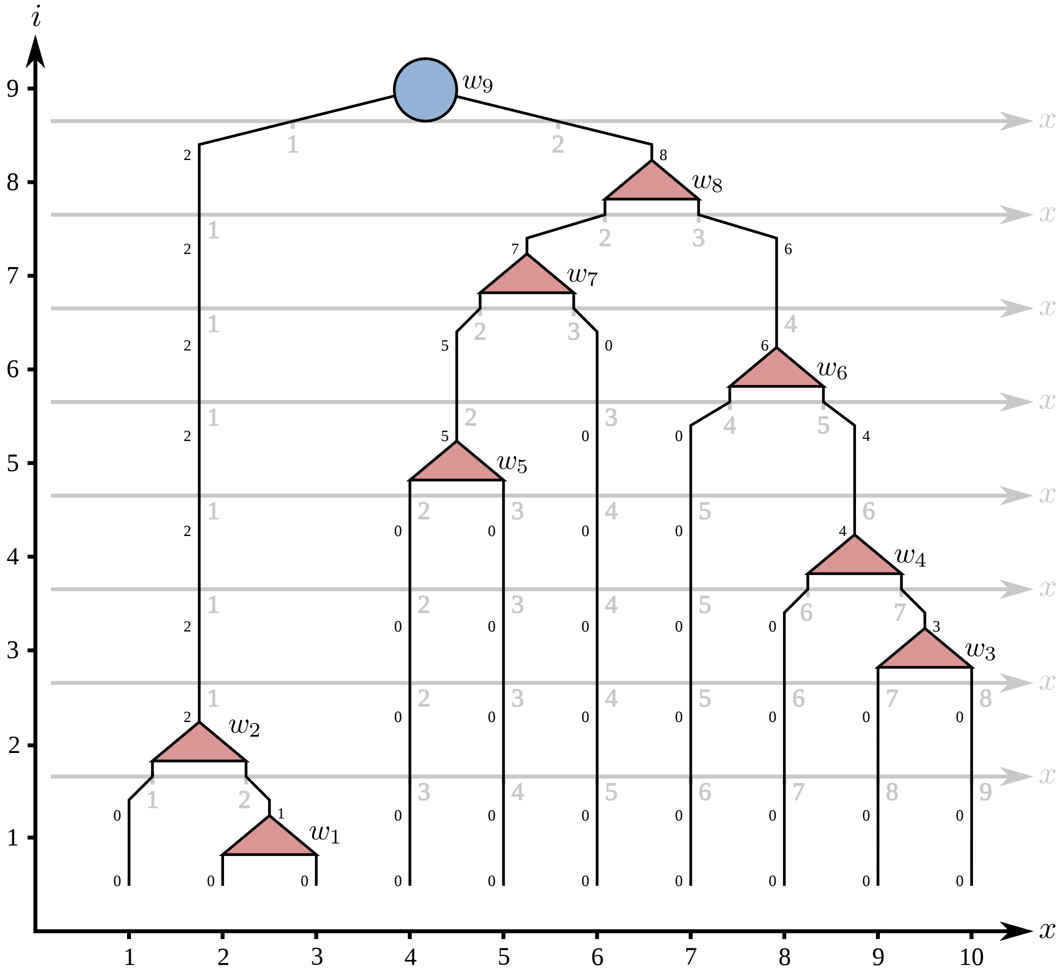

Figure 5.6: An example of how the indexing algorithm works based on a 10 site tensor network. The horizontal dimension gives position in the chain l and the vertical dimension gives SDRG iteration number i. The grey x-axes highlight the current horizontal position as the algorithm progresses. The isometries w_i are labeled by iteration number and hence there is one per i level. The numbers along the lines show the value of *current* for that leg at that point in the algorithm. It is this value which shows which isometry is connected below.

3. Store the values of $current_{Jmax}$ and $current_{Jmax+1}$ in arrays tL and tR that give the tensor connected to the lower left and right legs.

4. Update the *current* array by setting element $current_{Jmax}$ to i and removing element $current_{Jmax+1}$.

5. Finish the SDRG iteration, thus removing a site and relabelling the horizontal position. Increase i by 1 to start a new SDRG iteration and return to step 2.

At the end of the algorithm $order_i$ stores the vertical position of isometry i in the algorithm, and tL_i and tR_i store the index of the isometries that are connected to the bottom legs of each isometry. tL and tR are useful when indexing the density matrices as they show which isometry will be connected when contracting from the top down. Therefore we now have sufficient information to update the isometries going *upwards* through the network and update the density matrices going *downwards*. Figure 5.6 shows an explicit example of the indexing algorithm for a 10 site TTN. In this case the resulting arrays are

$$\begin{aligned} order &= \{2,1,7,6,2,4,2,2,1\}, \\ tL &= \{0,0,0,0,0,0,5,7,2\}, \\ tR &= \{0,1,0,3,0,4,0,6,8\}. \end{aligned} \tag{5.4.1}$$

5.4.2 Correlation Functions

An expectation value is calculated by contracting the appropriate operator, likely given as an MPO, with the whole TTN wavefunction ($|\Psi\rangle$) and its conjugate ($\langle\Psi|$). In general for a tensor network this is a very difficult and costly process. For the TTN it is made more simple by following the order in which the SDRG progresses, contracting a pair of MPO tensors with an isometry and its conjugate each step until the top of the tree.

When the expectation value is for an operator that does not act on every site, such as the two-point correlation function in fig. 5.7, only those tensors that affect the sites that the operators act on need to be included. This is known as the *causal cone* [34] and is drawn as a blue shadow in the holographic bulk. This allows for a reduction in the number of contractions that need to be performed to obtain a result. Further optimisations can be performed by storing blocks of tensor that have already been contracted for later use. This approach is particularly useful for the two-point correlator as, typically, the expectation value for all pairs of sites will

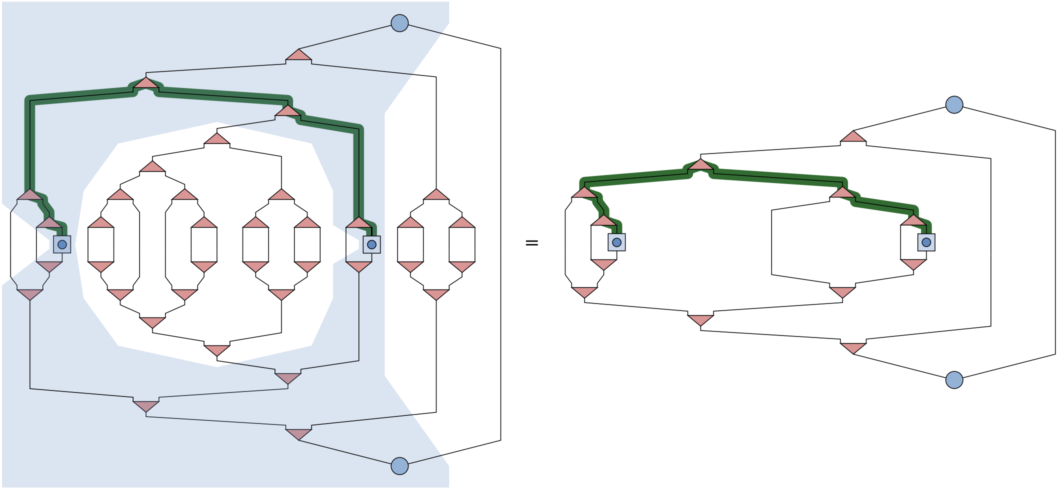

Figure 5.7: Diagram showing the TTN form of the correlation function $\langle \vec{s}_3 \cdot \vec{s}_{15} \rangle$ in the 20 site system from fig. 5.4(c). The causal cone is indicated by a light blue shaded region and the bold green line shows the path length through the TTN connecting the two sites. Otherwise lines and symbols as in fig. 5.4(c).

be needed and subsets of the tree will be reused multiple times. An algorithm to calculate two-point correlation progresses as follows:

1. Start with the left point at site $x = 1$ and right point at site $x + 1$.

2. Make a vector of the tensors that connect the left and right points to the top tensor.

3. Find the first value in the intersection of these two vectors. This gives the highest isometry in the path connecting the two points (see, for example, the bold green line in fig. 5.7).

4. Contract the spin operator for each point with the isometries and their conjugates up to the common node, tracing over any leg that is not connected to the operator.

5. Before contracting the common node store the new renormalised left point operator.

6. Contract up to the top tensor.

7. Move the right point to the right by one site. Repeat from step 2 but reuse the left block as the highest point in the path connecting the two points will

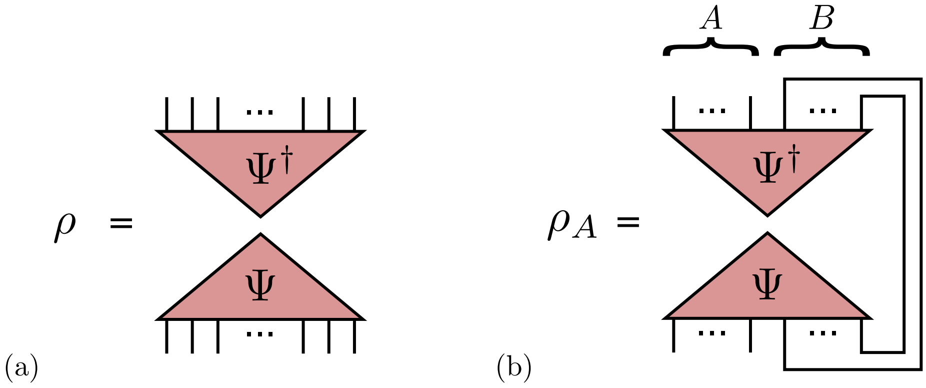

Figure 5.8: Diagrammatic form of (a) the TTN density matrix ρ and (b) the reduced density matrix ρ_A where the legs of block B have been traced over. For simplicity the large triangle represents the full TTN for wavefunction Ψ.

only ever grow, to a maximum at the top of the tree.

8. Restart with the left point across one site.

5.4.3 Entanglement Entropy

In a TTN the density matrix $\rho = |\Psi\rangle\langle\Psi|$ has a graphical form made up of a full TTN and its conjugate as shown in fig. 5.8(a). The reduced density matrix ρ_A for block A is the density matrix but with a partial trace over block B as discussed in chapter 2. As with the MPS, the reduced density matrix for the TTN is formed by contracting the legs of block B with themselves as shown in fig. 5.8(b).

Creating a reduced density matrix for a generic block is non-trivial as the number of elements in the matrix is $(d^{n_A})^2$, where d is the dimension of the single site ($d = 2$ for the spin-1/2 Heisenberg Hamiltonian) and n_A is the number of sites in the A block. However, the structure of the TTN can significantly reduce the size of the matrix and complexity of the calculation. Due to the fact that the w_i are isometric, any tensors that are wholly in block B do not need to be considered. Furthermore, any isometries that are wholly in the A block don't contribute to the entanglement entropy, so it is only the tensors that connect the two blocks that need to be contracted [43]. The random nature of the tensor network again causes problems with how to efficiently contract the network when the structure is not predefined. One way of doing so is:

1. Define a vector showing which sites are in block A and which are not. Label

block A sites with number 2 and those in B with a 0.

2. Following the pattern set out by the construction of the network, label each of the bottom legs of each tensor with a number 2 for when there is only block A below, 1 for when the tensor connects A and B and 0 for when there is just B below. For each w_i, the tensor that is connected to each of the bottom legs is stored in tL and tR so that the network can be contracted from top to bottom.

3. Using the information collected, one can then proceed in contracting the network starting from the top tensor, from left to right, contracting the tensor below if it is labelled with a 1, leaving the leg open if it is a 2 and tracing over the leg if it is a 0.

In this way the density matrix can be systematically constructed for any TTN, keeping the size to a minimum.

5.5 Results

In the following, we shall compare results for the disordered AFM Heisenberg model (5.1.1) when using a modern DMRG implementation, with those obtained from tSDRG. The set of couplings (J_i) shall always be taken from a *box-type* distribution [124], i.e. constant in the range $0 < 1 - \Delta J/2 < J_i < 1 + \Delta J/2 < 2$ and zero outside. Unless stated otherwise, we use strong disorder $\Delta J = 2^-$ in the following. We assume OBCs throughout.

5.5.1 Convergence and Ground-State Energies

In fig. 5.10 (main), we show the dependence of the disorder-averaged ground state energy per site, E_g/L, on ΔJ for constant L. We find that for both DMRG and tSDRG, the E_g/L values decrease for increasing ΔJ, i.e. the ground state energy lowers as disorder in the J_i couplings allows the system to form particularly energetically favourable spin configurations. We also see that the DMRG for the chosen values of χ and L reaches lower energies. This suggests that it is yet more efficient in finding an approximation to the true ground state energy. However, upon increasing ΔJ, the difference between DMRG and tSDRG is getting smaller. This is expected since SDRG is based on the idea that the contribution from the non singlet interactions is small, which is more accurate an assumption the greater the disorder. The figure also shows that increasing χ can considerably improve the results of the tSDRG. We expect a *variational* tSDRG to be at least as good as our

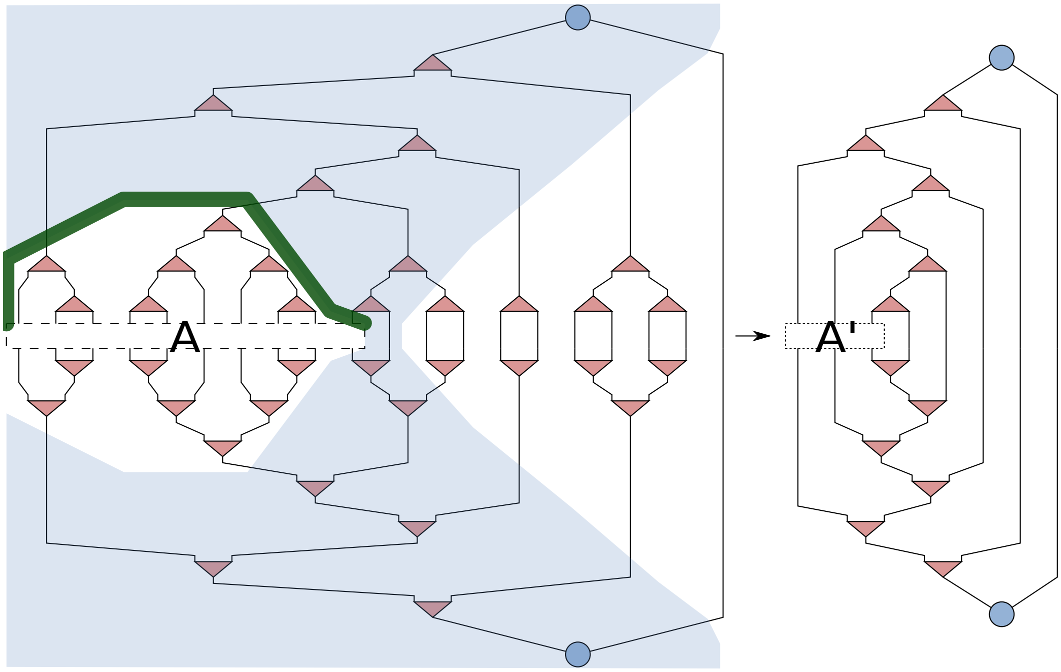

Figure 5.9: Diagram showing the TTN form of the reduced density matrix ρ_{A} for the block A indicated by the dashed rectangle 10 sites long in the 20 site system from fig. 5.4(c). Lines and symbols as in fig. 5.4(c) and 5.7. The bold line shows the minimal surface in the TTN between regions A and B (the rest of the chain). The right-hand diagram has been reduced in the horizontal direction to highlight the reduction in complexity due to the isometries.

DMRG. Here, however, we concentrate predominately on showing the validity and usefulness of a TTN approach to disordered chains. In fig. 5.10 (inset) we show E_g/L as a function of L for various values of χ at the strongest permissible disorder $\Delta J = 2^-$. We find that the values of E_g/L do not vary much anymore for system sizes $L \geq 100$. Conversely, E_g/L values for $L < 100$ are clearly dominated by the presence of OBCs.

5.5.2 Correlation Functions

The correlation functions for a strongly disordered Heisenberg chain are expected to average out to be a power-law decay [120]

$$\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle \sim \frac{(-1)^{x_2 - x_1}}{|x_2 - x_1|^2}, \tag{5.5.1}$$

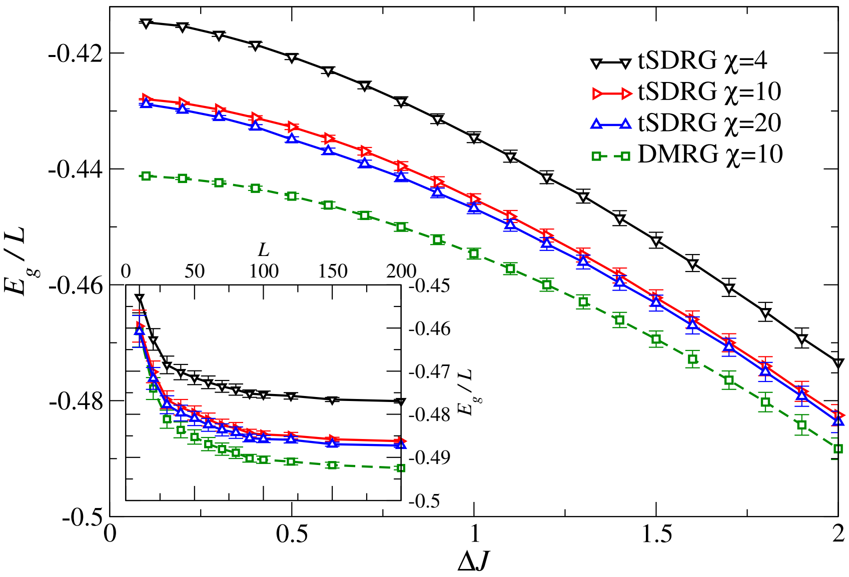

Figure 5.10: Ground state energy per site E_g/L as a function of disorder ΔJ for system size $L = 100$ for tSDRG (solid lines) and variational MPS (dashed). The error bars correspond to the standard error on the mean obtained from averaging over 200 different disorder configurations and various values of χ. Lines are guides to the eye. Inset: System size dependence of E_g/L for $\Delta J = 2^-$. Sizes $L = 10$–80 have been averaged over 500 disorder configuration, 90, 100 and 120 over 1000, 150 and 200 over 2000 configurations, respectively.

where $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle$ is understood to be the disorder-averaged expectation value of the two point spin-spin correlation function. The r^{-2} scaling of the correlation is a feature of the disorder in the system [120] and is different to that of the asymptotic behavior of the clean open Heisenberg XXX model [128]

$$\langle \vec{s}_r \cdot \vec{s}_0 \rangle \sim \frac{(-1)^r (\ln r)^{1/2}}{r}. \tag{5.5.2}$$

As discussed in chapter 4, correlation functions in tensor networks scale as $e^{-\alpha D(x_1,x_2)}$, where $D(x_1, x_2)$ is the number of tensors that connect site x_1 to x_2 [44]. tSDRG has a holographic geometry based on a random TTN, with path length $D_{\text{TTN}} \approx \log |x_2 - x_1|$, i.e. scaling logarithmically with distance when averaged. This

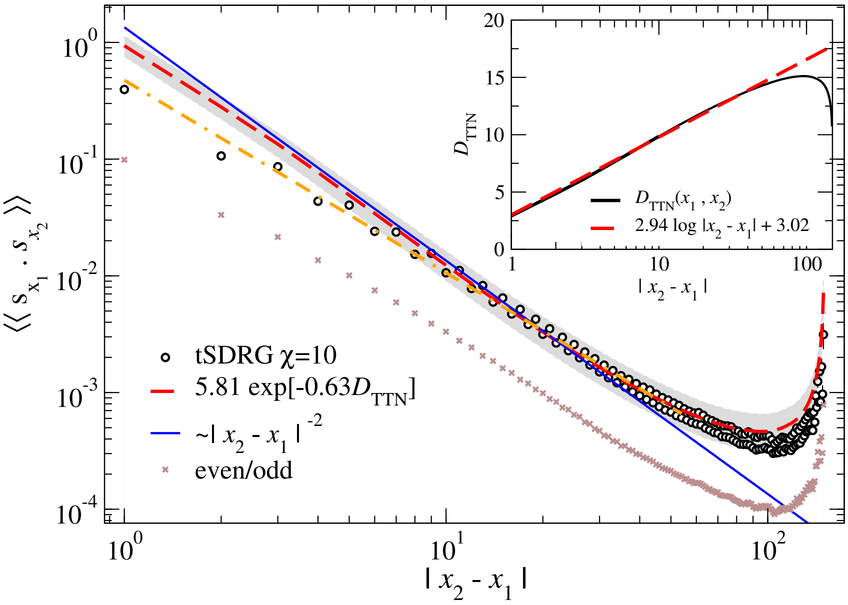

Figure 5.11: Correlation function for $L = 150$ and $\Delta J = 2^-$ averaged over 2000 samples for the direct calculation of $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle$ (black circles) and also via the holographic approach (5.5.3) using D_{TTN} (dashed red line with error of mean indicated by the grey shading) such that $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle \approx (5.81 \pm 0.93)\exp[-(0.62 \pm 0.02) D_{\mathrm{TTN}}]$. The expected thermodynamic scaling $|x_2 - x_1|^{-2}$ is also shown (solid blue line) while the dashed orange line denotes a power-law fit up to $|x_2 - x_1| = 50$ with slope 1.64. The (brown) crosses show $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle/4$ (for clarity) with all values for even distances $|x_2 - x_1|$ multiplied by 1.25. Inset: The holographic path length D_{TTN} connecting sites x_1 and x_2 averaged over the 2000 TTNs (black) and a fit in the logarithmic regime (red).

makes it much more suited to capture the desired power law decay

$$\begin{aligned} \langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle &\sim e^{-\alpha \langle D_{\mathrm{TTN}}(x_1,x_2) \rangle} \\ &\sim e^{-\alpha \log |x_2 - x_1|} \sim \ |x_2 - x_1|^{-a}. \end{aligned} \tag{5.5.3}$$

In fig. 5.11, we show the behaviour of $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle$ computed directly as well as its holographic estimate based on (5.5.3). We find that the behaviour for $|x_2 - x_1| \gg 1$ and $|x_2 - x_1| < L/2$ is indeed very similar for both approaches. The best fit value for α is 0.62 ± 0.02 where the error is the standard error. The fitting was performed using the *lsqcurvefit* function in MATLAB version 2013a. The function is based on a trust-region algorithm [129] with weights to take into account the accuracy of the data. We find that in the indicated distance regime, both measures of $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle$

are consistent with the expected r^{-2} behaviour. For $|x_2 - x_1| \gtrsim L/2$ we see that the boundaries lead to an upturn on the behaviour of $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle$ for both direct and holographic estimates. This upturn is a result of boundary effects and can easily be understood in terms of the holographic TNN: for $|x_2 - x_1| \geq L/2$, the average path length in the tree decreases (cp. fig. 5.9). This is also consistent with periodic systems where we expect correlation functions to be equal for $|x_2 - x_1| = r$ and $L - r$. In the inset of fig. 5.11 we show the distance dependence of D_{TTN} with $\chi = 10$. For $|x_2 - x_1| < L/2$, the data can be described by as linear behaviour in $\log|x_2 - x_1|$ with slope 2.94 ± 0.02. Note that this slope along with the value of $\alpha = 0.62 \pm 0.01$ gives an estimate of power-law exponent $a = (0.62 \pm 0.01) \times (2.94 \pm 0.01) = 1.84 \pm 0.04$ for fixed $L = 150$. Figure 5.12 shows that as L increases, the resulting value of the scaling power a also increases towards the expected value of 2 for larger systems upon increasing L. We have also checked that the differences between $\chi = 10$ and 20 remain within the error bars and hence we use $\chi = 10$ for calculations of $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle$ in fig. 5.11. We further note that fig. 5.11 shows a clear difference in the correlation function between even and odd distances, due to the fact that singlets can only form with nearest neighbours on the current coarse graining scale. The difference in magnitude is found to be 1/4 as predicted previously [130].

In addition to the power law scaling of mean correlations, it is expected [120] that the *typical* correlations scale as

$$\langle \log|\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle| \rangle \sim -|x_2 - x_1|^{1/2}, \qquad (5.5.4)$$

where the left hand side of (5.5.4) is the disorder-averaged mean of the log of the spin correlation function, i.e. the log of the geometric mean of the correlation function. Figure 5.13 shows that this typical correlation function indeed scales as $|x_2 - x_1|^{1/2}$ and the quality of the fit increases upon increasing χ and system size. For $L = 150$, as χ is increased from 4 to 50, the agreement with (5.5.4) improves up to approximately half the system size, at which point boundary effects become important as in fig. 5.11. The typical/geometric mean of the path lengths does not allow to reproduce the typical correlation behaviour (5.5.4), but rather continues to retain a logarithmic scaling behaviour. This suggests that the TTN constructed by our tSDRG selects those path lengths corresponding to mean correlation. Clearly, eq. (5.5.3) ignores correlation information stored in the isometry tensors and we expect that its inclusion will recover also the typical correlation behaviour. Indeed, the need to increase χ in fig. 5.13 in order to reproduce (5.5.4) already confirms that the tensor content is very important here.

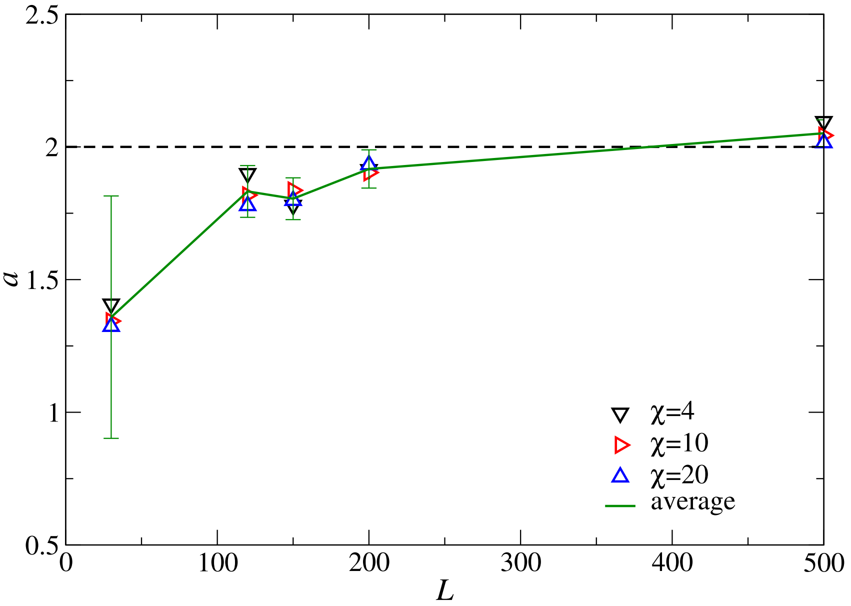

Figure 5.12: The scaling parameter a from eq. (5.5.3) as a function of system size L for different values of χ at $\Delta J = 2^{-}$. The solid lines are guides to the eye only. The asymptotic value of $a = 2$ is indicated by the horizontal dashed line.

While eq. (5.5.3) neatly describes the power law behaviour of the data, a more accurate ansatz should be

$$\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle \simeq A \langle e^{-D_{\mathrm{TTN}}(x_1,x_2)} \rangle^{a}. \qquad (5.5.5)$$

We plot the fit of this along with the correlation data rescaled to remove the even-odd discrepancy discussed above in fig. 5.14. This shows that $\langle\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle\rangle \approx (1.67 \pm 0.10)\langle \exp[-D_{\mathrm{TTN}}] \rangle^{(0.69\pm0.01)}$ is a remarkably accurate fit to the data for all length scales. This implies that the majority of the correlation information is stored in the structure of the TTN rather than the contents of the tensors.

5.5.3 Entanglement Entropy

In general, the entanglement entropy $S_{\mathrm{A|B}}$ is difficult to compute as the size of the reduced density matrix ρ_{A} scales exponentially with the size of block A. While for special cases, such as the XX model [131], $S_{\mathrm{A|B}}$ can be computed more easily, the general strategy involves finding the eigen- or singular values of ρ_{A} [30].

The TTN representation of tSDRG gives an alternative means of calculating

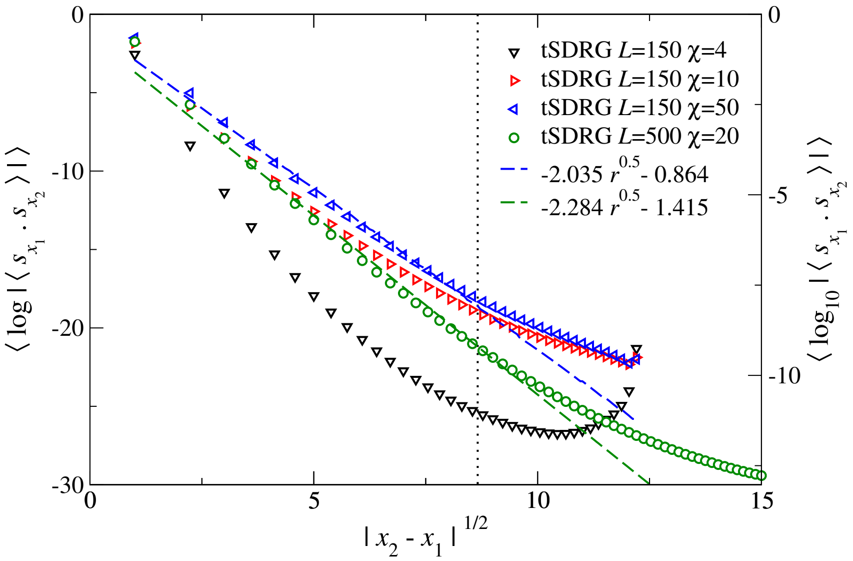

Figure 5.13: The *typical* spin correlation function averaged over 2000 samples for $L = 500$ (green circles) and $L = 150$ (triangles) and χ values as given in the legend. Error bars are within symbol size throughout. The dashed lines are fits to the linear regimes for $L = 150$, $\chi = 50$ (blue) and $L = 500$, $\chi = 20$ (green). The vertical dotted line indicates half the system size for $L = 150$.

$S_{\mathrm{A|B}}$ for any bipartitions A and B of the system. In a similar manner to the correlation functions, the geometry of the tensor network is related to its ability to capture $S_{\mathrm{A|B}}$. As mentioned in chapter 4, $S_{\mathrm{A|B}}$ is proportional to the minimum number of indices, n_A, that one would have to cut to separate a block A of spins from the rest B of the chain (cp. fig. 5.9) [44]. For the TTN the position of the block in the chain alters the number of indices that have to be cut to separate it from the rest of the system. This suggests that there are spatial regions in the chain that are more and less entangled, which is likely to be true for a strongly disordered spin chain. The concept is hence similar to discussing the entanglement in the MDH implementation [118] of SDRG, where the entanglement entropy is related to the number of singlets that have to be broken to separate a region from the rest [122].

In fig. 5.15 we show that the average value of $S_{\mathrm{A|B}}$ remains approximately constant upon increasing the disorder, while the average of the maximal $S_{\mathrm{A|B}}$ shows a pronounced increase. This indicates that the full distribution of $S_{\mathrm{A|B}}$ develops long tails with large $S_{\mathrm{A|B}}$ values when increasing ΔJ. For strong disorders $\Delta J \gtrsim 1.5$ we find that tSDRG captures more entanglement than DMRG. The DMRG estimates of $S_{\mathrm{A|B}}$ are consistently below the values obtained by the tSDRG, but when increasing

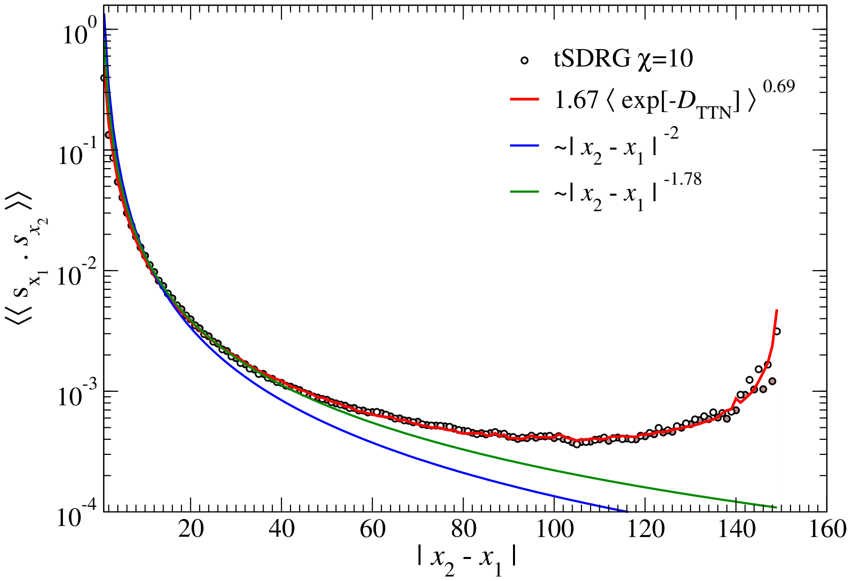

Figure 5.14: Rescaled correlation function to remove odd-even effects as discussed in text on a semi-log plot. Odd points are open circles and even are shaded. The expected thermodynamic scaling is shown as a solid blue line. The fitted scaling factor from fig. 5.12 plotted as a solid green line. The red line indicates the alternative holographic fitting with $\langle\langle\vec{s}_{x_1}\cdot\vec{s}_{x_2}\rangle\rangle \approx (1.67 \pm 0.10)\langle\exp[-D_{\mathrm{TTN}}]\rangle^{(0.69\pm0.01)}$.

χ the deviation is reduced. This behaviour is most pronounced for the average of the maximal $S_{\mathrm{A|B}}$ values. For example, with $\chi = 20$, the $S_{\mathrm{A|B}}$ values obtained for DMRG deviate from the tSDRG results around $\Delta J \approx 1.2$. Hence we see that an increase in $S_{\mathrm{A|B}}$ requires a considerable increase in χ for DMRG to accurately capture the entanglement. On the other hand, for weak disorders $\Delta J \lesssim 0.5$, DMRG gives consistent results already for small $\chi = 10$. The values obtained for $S_{\mathrm{A|B}}$ from tSDRG are much higher in this regime. We believe this to be an overestimation of $S_{\mathrm{A|B}}$ by the tSDRG because, as discussed before, tSDRG selects most strongly the singlet pairs in the disordered system, which of course become less prevalent for low disorder.

Figure 5.16 shows that when L is increased for $\Delta J = 2$, both the average and average peak values of $S_{\mathrm{A|B}}$ increase logarithmically in L. This again implies that as L is increased, the χ value for DMRG needs to be increased also to be able to capture the entanglement. On the other hand, the holographic nature of the TTN means that the minimal surface in the network increases with system size and thus describes this entanglement without the need to increase χ. Although

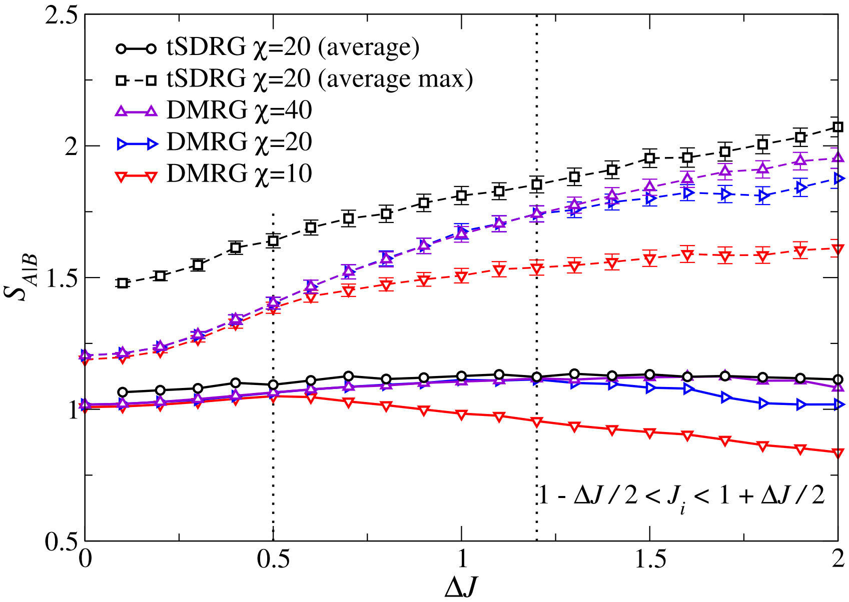

Figure 5.15: Entanglement entropy $S_{\mathrm{A|B}}$ for all possible bipartitions (cp. fig. 5.9) for $L = 30$ as a function of ΔJ averaged over 100 disorder configuration using DMRG and tSDRG. Solid lines indicate the arithmetic mean over disorder configurations while dashed lines denote the mean of the maximal $S_{\mathrm{A|B}}$ values at the chosen ΔJ. Lines connecting symbols are guides to the eye only. Error bars denote standard error of the mean when larger than symbol size. The two vertical dotted lines highlight $\Delta J = 0.5$ and 1.2 as discussed in the text.

$S_{\mathrm{A|B}}$ is therefore captured well by the network, contracting ρ_{A} for larger L becomes increasingly difficult, even with the simplifications suggested in section 5.5.3, since the size of the matrices scales as $O(\chi^{n_{\mathrm{A}}})$. We therefore have to restrict ourselves to smaller χ and L values than in sections 5.5.1 and 5.5.2.

In refs. [122, 132], Refael and Moore calculate a *block* entanglement $S_{\mathrm{A,B}}$ in the random singlet phase and show that it scales as

$$S_{\mathrm{A,B}} \sim \frac{\log 2}{3} \log_2 L_{\mathrm{B}} \approx 0.231 \ldots \log_2 L_{\mathrm{B}}, \tag{5.5.6}$$

where region B is a block of extent L_{B} in the centre of the spin chain. Note that this implies an effective central charge [122] of $\tilde{c} = 1 \cdot \log 2$. This is different from the bipartition entanglement $S_{\mathrm{A|B}}$ that we considered before. We show the resulting $S_{\mathrm{A,B}}$ in fig. 5.17. The figure clearly indicates that finite size effects become prevalent for large L_{B}, so we fit for $L_{\mathrm{B}} \leq L/2$ only. The resulting scaling behaviour $S_{\mathrm{A,B}} \approx$

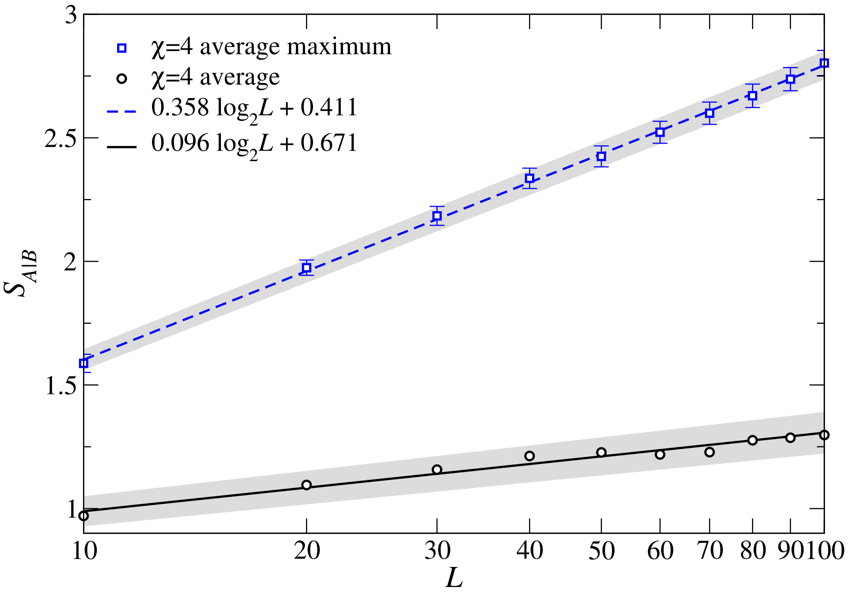

Figure 5.16: Entanglement entropy $S_{\mathrm{A|B}}$ as a function of L averaged over 100 samples and all possible bipartitions (as in fig. 5.15) for $\chi = 4$ and $\Delta J = 2^-$. The dashed blue line is the fit $(0.358 \pm 0.005)\log_2 L + (0.41 \pm 0.03)$, the solid black line is $(0.096 \pm 0.008)\log_2 L + (0.67 \pm 0.04)$. Error bars denote the standard error of the mean for the $S_{\mathrm{A|B}}$ values when larger than symbol size while grey shaded regions show the standard error of the indicated fits.

$(0.22 \pm 0.02)\log_2 L_{\mathrm{B}}$ is consistent with eq. (5.5.6). We note, however, that finite size corrections might still be present at the system size available to us here; ideally one should aim for much larger system sizes [133].

We finally also examine the entanglement entropy per bond, S/n_{A}, of a TTN for both bipartitions A|B and blocks A, B with $\chi = 10$ when averaging over 500 disorder configurations with $L = 50$. Figure 5.18 shows that away from the boundaries S/n_{A} saturates to the same constant 0.47 ± 0.02 for bipartitions and blocks. Note that for $L_{\mathrm{B}} \sim L/2$, we find that up to 20% of our samples for $\chi = 10$ lead to calculations of $S_{\mathrm{A,B}}$ consuming memory beyond 100GB and therefore fail to complete. Larger memory calculations are currently out of reach for us and we disregard the configurations. Nevertheless, we believe that this will not greatly change the average values of $S_{\mathrm{A,B}}/n_{\mathrm{A}}$ reported here as the higher failure rate is for block sizes where boundary conditions become influential, which is supported by the calculations for smaller χ. For $\chi = 4$ we find 0.42 ± 0.02 for both blocks and bipartitions with a much lower failure rate ($< 1\%$) due to the smaller size of the density matrices. This might conceivably suggest that $S/n_{\mathrm{A}} = 0.5$ is a limiting value

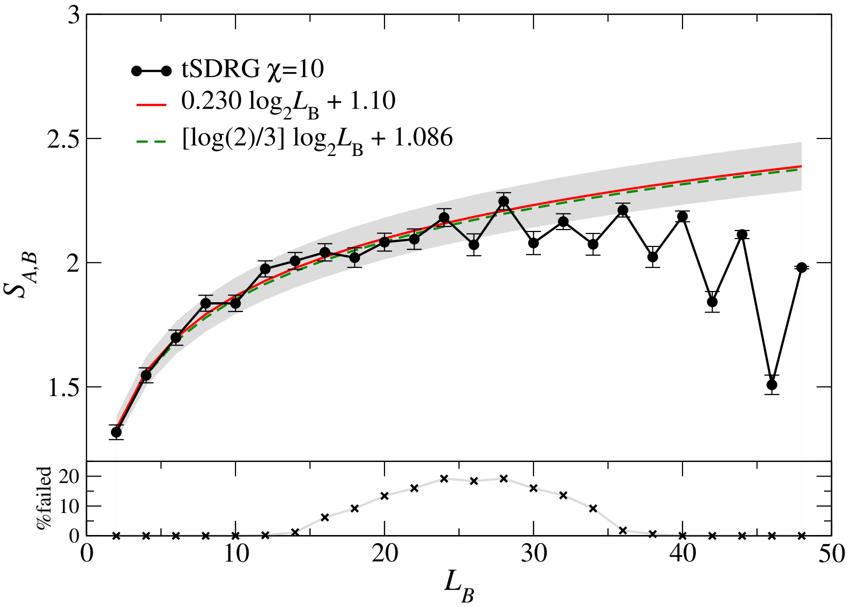

Figure 5.17: The entanglement entropy $S_{\rm A,B}$ (black) averaged over 500 samples as a function of the size of a *block* $L_{\rm B}$ placed in the middle of a chain with $L = 50$ for $\chi = 10$ and $\Delta J = 2^{-}$. The fitting (red, solid line) gives $S_{\rm A|B} = (0.22 \pm 0.02)\log_2 L_{\rm B} + (1.12 \pm 0.05)$ for $L_{\rm B} \leq 25$, above which finite size effects dominate. The grey shaded region indicates the accuracy of the fit. The (green) dashed line shows the entanglement scaling (5.5.6) from ref. [132] with the vertical position fitted to the point $L_{\rm B} = 2$. The straight black lines are a guide to the eye only. At the bottom, we show the failure rate in percent (crosses) for different $L_{\rm B}$.

for larger χ and L. In turn, this would imply $n_{\rm A} = 2 \log L_{\rm B}/3$. This is consistent with ref. [44] and implies that the entanglement entropy is proportional to the length of the holographic minimal surface that connects the two blocks.

5.6 Conclusion

In this chapter, we demonstrate the validity and usefulness of a suitably adaptive tensor network approach to locally disordered one-dimensional quantum many-body systems. In contrast to traditional DMRG approaches to disordered systems, where the initial geometry of the MPS ignores the disorder and only takes it into account at the stage of variational sweeps [19], our approach incorporates the disorder into the fabric of its tensor network. We believe this strategy to be inherently more suited to disordered systems — the results presented here show that the accuracy

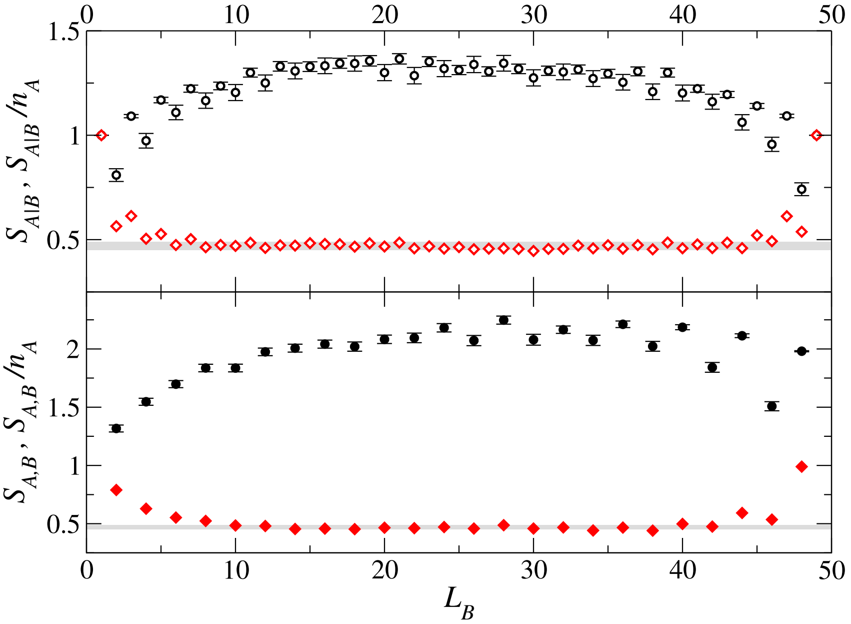

Figure 5.18: Entanglement entropy S (black circles) and entanglement entropy per bond S/n_{A} (red diamonds) for bipartitions A|B (top, open symbols) and blocks A, B (bottom, filled symbols) with $\chi = 10$ and $\Delta J = 2^{-}$. The entanglement per bond saturates to 0.47 ± 0.02 for bipartitions and 0.48 ± 0.02 for blocks (grey shaded regions).

of tSDRG is already comparable to DMRG without including any additional variational updates. This advantage is particularly evident for long-ranged correlations and an entanglement entropy that violates the area law.

Our results furthermore show that, when disorder-averaged, a random AFM spin 1/2 system is well characterised by an effective CFT on the boundary of a discretised holographic bulk. We believe, to the best of our knowledge, that we have thus shown the *quantitative* validity of holography for the first time here. In particular, our spin-spin correlation function, fig. 5.11, as well as the block and bipartition entanglement entropies, fig. 5.18 show excellent qualitative and numerical agreement with their holographic counterparts. Such an agreement also reconfirms that the self-assembly of the TTN produces the necessary tensor network geometry.

Whilst here we concentrated on the disordered Heisenberg model, the method should be straightforwardly applicable to the XX and XXZ models as studied by Fisher [120]. Similarly it should work for the Jordan-Wigner-transformed equivalent fermionic models with a disordered hopping parameter [134]. It should also be permissible to implement different forms of disorder, such as aperiodic sequences

[135] as long as the singlet approximation is valid throughout the renormalisation procedure. We have checked that tSDRG, just as the SDRG of Hikihara [124], is also able to model random FM/AFM couplings that create large effective spins as the renormalisation progresses. As such it may be possible to use our approach to study higher spin systems given a suitably high χ. It should also be fairly simple to extend the tSDRG method to periodic systems by introducing a bond between the first and last MPO tensor, which is effectively taking a trace over the MPO. We note that implementation of *on-site disorder*, such as in the random transverse field Ising model [121], does not appear to have a natural implementation using the local RG outlined in section 5.2. Here it may be possible to implement a tensor network with a different structure, but at the moment it is not clear to us how this would be performed.

The tensor network approach makes finding other expectation values, i.e. in addition to those studied here, straightforward as they are simply the contraction of the set of isometries with a matrix operator. An example is the *string order parameter* [136] that is used to find a hidden topological order in the ground state [137]. If the entanglement entropy can be found, so too can the entanglement spectrum, which has become a popular means of characterising many-body wave functions [73, 75, 77, 138, 139, 140, 141], for better or for worse [142]. Excited states can be found by diagonalising the top tensor and instead of keeping the lowest energy eigenvector, keeping a suitable set of higher energy eigenvectors. This will only be accurate for low energy excitations as at each step of the renormalisation process only the low energy components are kept while information about higher energy modes is discarded. Furthermore, it is possible that when moving far away from the ground state the geometry of the network is no longer appropriate.

Our local RG procedure selects spin pairs based on energy gaps. It is tempting to reformulate this based on the local entanglement content of such pairs. However, it is not straightforward to find such a local measure that captures energies and wave functions well simultaneously. In particular, we do not find a convenient local entanglement measure that would have a simple relation to the local values of J_i. More promising might be the implementation of a variational TTN [43]. Our initial results suggest that this does indeed improve the energy values, but at considerably increased efforts in implementation and computation — every disorder configuration of course necessitating its own variationally updated tree structure.

Chapter 6

Leaf-to-Leaf Path Lengths in Complete Tree Graphs

6.1 Introduction

As discussed in section 4.3 the geometry of a tensor network plays a large role in defining the entanglement and correlation properties that the network can encode. When the networks are simple, such as an MPS, it is straightforward to find analytic expressions for the minimal surface and path length that define these properties; when networks get more complicated it is not so clear. In chapter 5 we find that the tensor network is an inhomogeneous binary tree where the structure is defined by the coupling strengths in the Hamiltonian [12]. To analyse the form of the correlation functions we used numerics to ascertain the average path length (see inset of fig. 5.11) and minimal surface as analytic expressions are not available. The aim of this chapter is to start the discussion of the structure of TTNs in terms of graph theory.

The study of graphs and trees, i.e. points (or *vertices*) with pairwise relations (or *edges*) between them, has a long and distinguished history inside and outside of science. In computer science, graphs, trees and their study are closely connected, e.g. with sorting and search algorithms [143]; in chemistry the Wiener number is a topological index intimately correlated with, e.g., chemical and physical properties of alkane molecules [144]. In physics, graphs are equally ubiquitous, not least because of their immediate usefulness for systematic perturbation calculations in quantum field theories [145]. In mathematics, graph theory is in itself an accepted branch of mainstream research and graphs are a central part of the field of discrete mathematics [146]. Away from science, trees and graphs are perhaps most recognisable in the form of a *family tree*, which is used in mapping family genealogy. The connection to family history will become apparent in the nomenclature of tree graph theory.

An important concept that appears in all these fields is the *distance* in a graph, i.e. the number of edges connecting two vertices [147, 148, 149]. For tree graphs, defined as undirected graphs in which any two vertices are connected by only one path, various results exist ([150, 151, 152] for example) that compute the distance from the top of the tree (the *root*) to the bottom (its *leaves*). In a binary tree such as shown in Fig. 6.1 this distance might correspond, e.g. to the number of yes/no decisions one performs when searching for information.

In this chapter we take the example of a regular, complete binary tree as the simplest case within the set of all possible trees that can be generated using tSDRG. These have the same structure as regular TTNs [153, 154]. We let the *leaves* of the tree be surrogate lattice sites and therefore fix their position and order. The lengths that we concentrate on are leaf-to-leaf path lengths across the ordered tree, which are related to the two-point correlation functions discussed in chapters 4 and 5. We derive the average leaf-to-leaf path length for varying leaf separation with leaves ordered in a one-dimensional line as shown e.g. in Fig. 6.1 for a binary tree. The method is then generalised to m-ary trees and the moments of the leaf-to-leaf distances. Explicit analytical results are derived for finite and infinite trees as well as the case of PBCs.

Throughout the rest of this chapter the term *path length* will be used to mean the number to vertices that separate two leaves. This is in order to be consistent with the tensor network literature (e.g. [12, 44]) and the rest of the thesis. In most graph theory literature, including [13], distance is calculated by summing the edges rather than vertices. Converting between the two is simple as the number of edges between two leaves is 1 greater than the number of vertices. We also note that in graph theory the term *path length* usually refers to the sum of the levels of each of the vertices in the tree [143].

6.2 Average Leaf-to-Leaf Path Length in Complete Binary Trees

6.2.1 Recursive Formulation

Let us start by considering the complete binary tree shown in fig. 6.1. It is a connected graph where each vertex is 3-valent and there are no loops. The *root node* is the vertex with just two degrees at the top of fig. 6.1. The rest of the

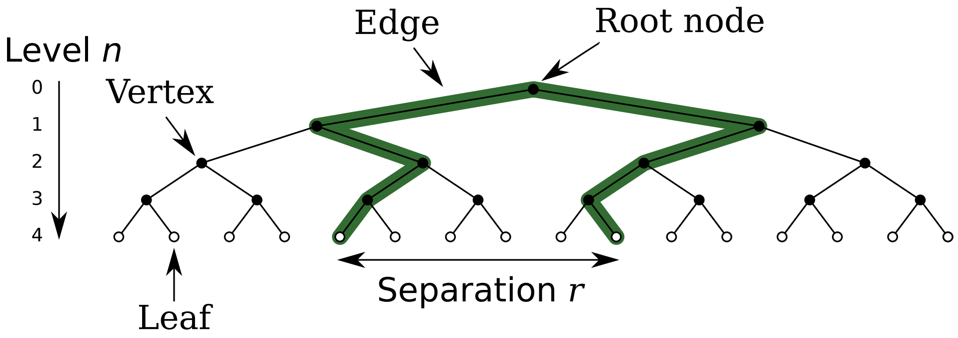

Figure 6.1: A complete binary tree with various definitions discussed in main text labelled. Circles (•, ∘) denote vertices while lines indicate edges between the vertices of different depth. The tree as shown has a depth of 4 and $L = 16$ leaves (∘). The indicated separation is $r = 5$ while the associated leaf-to-leaf path length equals $\ell = 7$ as indicated by the thick line.

vertices each have two *child* nodes and one *parent*. A *leaf node* has no children. The *depth* of the tree denotes the number of vertices from the root node with the root node at depth zero. With these definitions, a binary tree is *complete* [155] or *perfect* [156] if all of the leaf nodes are at the same depth and all the levels are completely filled. We now denote by the *level*, n, a complete set of vertices that have the same depth. These are enumerated with the root level as 0. We will refer to a *level n tree* as a complete tree where the leaves are at level n. The *leaf-to-leaf path length*, ℓ, is the number of vertices that are passed to go from one leaf node to another (cp. fig. 6.1).

Let us now impose an *order* on the tree of fig. 6.1 such that the leaves are enumerated from left to right to indicate position values, x_i, for leaf i. Then we can define a leaf *separation* $r = |x_i - x_j|$ for any pair of leaves i and j. This is equivalent to the notion of distance on a one-dimensional physical lattice. Let the *length* L be the length of the lattice, i.e. number of leaf nodes. Then for such a complete binary tree, we have $L = 2^n$. Clearly, there are many pairs of leaves separated by r from each other (cp. fig. 6.1). Let $\{\ell_n(r)\}$ denote the set of all corresponding path lengths. We now want to calculate the average path length $\mathcal{L}_n(r)$ from the set $\{\ell_n(r)\}$. We first note that for a level n tree the number of possible paths with separation r is $2^n - r$. In fig. 6.2, we see that any complete level n tree can be decomposed into two level $(n-1)$ sub-trees each of which contains 2^{n-1} leaves. Let $\mathcal{S}_n(r)$ denote the sum of all possible path lengths encoded in the set $\{\ell_n(r)\}$.

The structure of the decomposition in fig. 6.2 suggests that we need to distin-

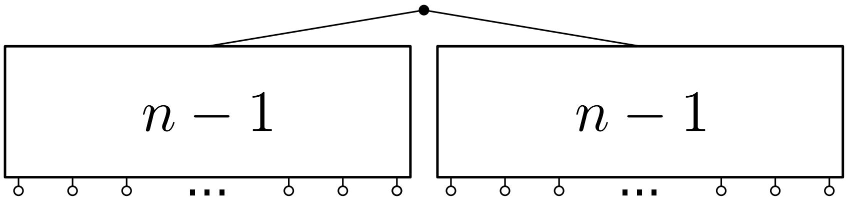

Figure 6.2: Schematic decomposition of a level n tree with root node (●) and leaves (○) into two level $n-1$ trees (rectangles) each of which has 2^{n-1} leaves.

guish two classes of separations r. First, for $r < 2^{n-1}$, paths are either completely contained within each of the two level $(n-1)$ trees or they bridge from the left level $(n-1)$ tree to the right level $(n-1)$ tree. Those which are completely contained sum to $2\mathcal{S}_{n-1}(r)$. For those paths with separation r that bridge across the two level $(n-1)$ trees, there are r of such paths and each path has length $\ell_{n-1} = 2n-1$. Next, for $r \geq 2^{n-1}$, paths no longer fit into a level $(n-1)$ tree and always bridge from left to right. Again, each such path is $(2n-1)$ long and there are $L-r = 2^n - r$ such paths. Putting it all together, we find that

$$\mathcal{S}_n(r) = \begin{cases} 2\mathcal{S}_{n-1}(r) + (2n-1)r, & r < 2^{n-1}, \\ (2n-1)(2^n - r), & r \geq 2^{n-1}. \end{cases} \tag{6.2.1}$$

for $n > 1$ and with $\mathcal{S}_1(r) = 1$. Dividing by the total number of possible paths with separation r then gives the desired average path length

$$\mathcal{L}_n(r) \equiv \frac{\mathcal{S}_n(r)}{2^n - r}. \tag{6.2.2}$$

6.2.2 An Explicit Expression

As long as $r < 2^{n-1}$, eq. (6.2.1) can be recursively expanded, i.e.

$$\begin{aligned} \mathcal{S}_n(r) &= 2\mathcal{S}_{n-1}(r) + (2n-1)r \\ &= 2\left[2\mathcal{S}_{n-2}(r) + (2(n-1)-1)r\right] + 2nr \\ &= \ldots \end{aligned} \tag{6.2.3}$$

After ν such expansions, we arrive at

$$\mathcal{S}_n(r) = 2^{\nu}\mathcal{S}_{n-\nu}(r) + \sum_{k=0}^{\nu-1} 2^k(2(n-k)-1)r. \tag{6.2.4}$$

The expansion can continue while $r < 2^{n-\nu-1}$. It terminates when $n-\nu$ becomes so small that the leaf separation r is no longer contained within the level-$(n-\nu)$ tree. From eq. (6.2.1) it is clear that the recursion only continues when $r < 2^{n-1}$ or

$$n > \log_2 r + 1. \tag{6.2.5}$$

The critical value n_c is the largest integer value that does not satisfy eq. (6.2.5), or equivalently 1 less than the smallest integer that does satisfy this condition. Hence

$$\log_2 r + 1 < n_c + 1 \leq \log_2 r + 2 \quad ; \quad n_c \in \mathbb{Z}, \tag{6.2.6}$$

which can be expressed succinctly as

$$n_c(r) = \lfloor \log_2 r \rfloor + 1, \tag{6.2.7}$$

where $\lfloor \cdot \rfloor$ denotes the floor function. For simplicity, we will suppress the r dependence, i.e. we write $n_c \equiv n_c(r)$ in the following. Continuing with the expansion of $\mathcal{S}_n(r)$ up to the n_c term, we find

$$\mathcal{S}_n(r) = 2^{n-n_c}\mathcal{S}_{n_c}(r) + \sum_{k=0}^{n-n_c-1} 2^k \left[2(n-k)-1\right] r. \tag{6.2.8}$$

When the final bracket is expanded, it is simply the sum of three geometric series

$$\sum_{k=0}^{n-n_c-1} 2^k \left[2(n-k)-1\right] r = rn \sum_{k=1}^{n-n_c} 2^k - r \sum_{k=1}^{n-n_c-1} k2^{k+1} - r \sum_{k=0}^{n-n_c-1} 2^k. \tag{6.2.9}$$

The first part can be simplified using

$$\sum_{k=1}^{l} x^k = \frac{x(1-x^l)}{1-x}, \tag{6.2.10}$$

the second part using

$$\sum_{k=1}^{l} kx^{k+1} = \frac{x(1-x^{l+1})}{(1-x)^2} - \frac{x+lx^{l+2}}{1-x}, \tag{6.2.11}$$

and the final part using

$$\sum_{k=0}^{l} x^k = \frac{1-x^{l+1}}{1-x}. \tag{6.2.12}$$

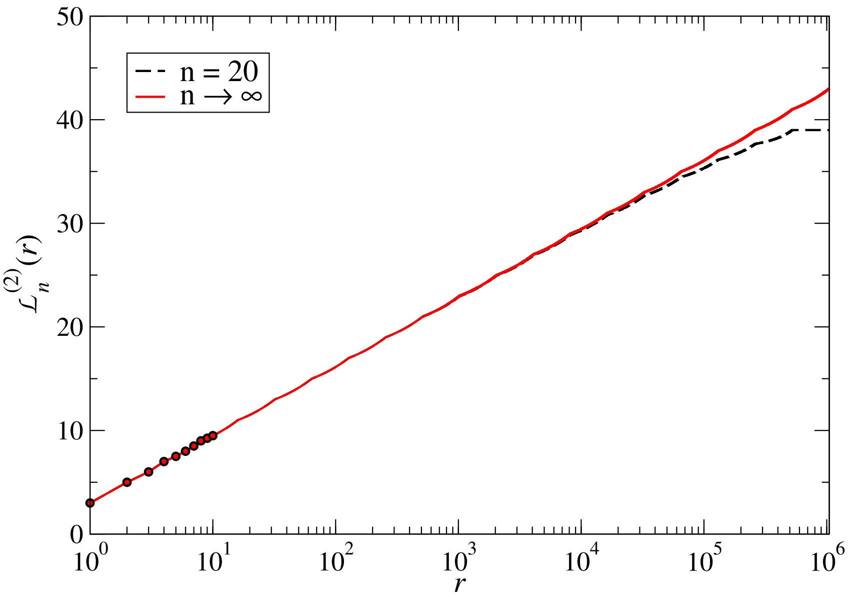

Figure 6.3: The average leaf-to-leaf path length $\mathcal{L}_n(r)$ versus leaf separation r for a complete binary tree of $n = 20$ (dashed), i.e. length $L = 2^{20} = 1,048,576$, and also for $n \to \infty$ (solid). The first 10 values are indicated by circles.

When all put together eq. (6.2.9) is

$$\mathcal{S}_n(r) = 2^{n-n_c}\mathcal{S}_{n_c}(r) + r\left[2^{n-n_c}(2n_c+3) - (3+2n)\right]. \tag{6.2.13}$$

From eq. (6.2.1), we have $\mathcal{S}_{n_c}(r) = (2n_c - 1)(2^{n_c} - r)$. Thus eq. (6.2.13) becomes

$$\mathcal{S}_n(r) = 2^n(2n_c - 1 + 2^{2-n_c}r) - (2n+3)r. \tag{6.2.14}$$

Hence the average leaf-to-leaf distances are given by

$$\mathcal{L}_n(r) = \frac{1}{2^n - r}\left[2^n(2n_c - 1 + 2^{2-n_c}r) - (2n+3)r\right]. \tag{6.2.15}$$

In the limit of $n \to \infty$ for fixed r, we have

$$\lim_{n\to\infty} \mathcal{L}_n(r) \equiv \mathcal{L}_\infty(r) = 2n_c - 1 + 2^{2-n_c}r. \tag{6.2.16}$$

We emphasise that $\mathcal{L}_\infty(r) < \infty \; \forall r < \infty$.

In fig. 6.3 we show finite and infinite path lengths $\mathcal{L}_n(r)$. We see that whenever $r = 2^i$, $i \in \mathbb{N}$, we have a cusp in the $\mathcal{L}_n(r)$ curves. Between these points, the floor function ($\lfloor\cdot\rfloor$) enhances deviations from the leading $\log_2 r$ behavior. This behaviour is from the self-similar structure of the tree. Consider a sub-tree with ν levels; the largest separation that can occur in that sub-tree is $r = 2^\nu$, which has average length $2\nu - 1$. When r becomes larger than the sub-tree size the path length can no longer be $2\nu - 1$ but always larger, so there is a cusp where this distance is removed from the possibilities. The constant average distance when $r \geq \frac{L}{2}$ is because there is only one possible path length that connects the two primary sub-trees, which is clear from (6.2.1).

6.3 Generalization to Complete m-ary Trees

6.3.1 Average Leaf-to-Leaf Path Length in Complete Ternary Trees

Ternary trees are those where each node has *three* children. Let us denote by $\mathcal{S}_n^{(3)}(r)$ and $\mathcal{L}_n^{(3)}(r)$ the sum and average, respectively, of all possible leaf-to-leaf path lengths $\{\ell_n^{(3)}(r)\}$ for given r in analogy to the binary case discussed before. Furthermore, $L = 3^n$. Following the arguments which led to eq. (6.2.1), we have

$$\mathcal{S}_n^{(3)}(r) = \begin{cases} 3\mathcal{S}_{n-1}^{(3)}(r) + 2(2n-1)r, & r < 3^{n-1}, \\ (2n-1)(3^n - r), & r \geq 3^{n-1}. \end{cases} \tag{6.3.1}$$

This recursive expression can again be understood readily when looking at the structure of a ternary tree. Clearly, $\mathcal{S}_n^{(3)}(r)$ will now consist of the sum of leaf-to-leaf distances for three level n trees, plus the sum of all paths that connect the nodes across the three trees of level n. The distances of these paths is solely determined by n irrespective of the number of children and hence remains $2n - 1$. As before, we need to distinguish between the case when r fits within a level $n - 1$ tree, i.e. $r < 3^{n-1}$, and when it connects different level $n - 1$ trees, $r \geq 3^{n-1}$. For $r < 3^{n-1}$, there are now $2r$ such paths, i.e., r between the left and centre level $n-1$ trees and r the centre and right level $n-1$ trees. For $r \geq 3^{n-1}$ there are $L - r = 3^n - r$ paths.

We again expand the recursion (6.3.1) and find, with $n_c^{(3)} = \lfloor \log_3 r \rfloor + 1$ in analogy to (6.2.7), that

$$\mathcal{S}_n^{(3)}(r) = 3^n \left[2n_c^{(3)} - 1 + 3^{1-n_c^{(3)}} r\right] - 2(n+1)r \tag{6.3.2}$$

and

$$\mathcal{L}_n^{(3)}(r) = \frac{S_n^{(3)}(r)}{3^n - r}, \tag{6.3.3}$$

$$\mathcal{L}_\infty^{(3)}(r) = 2n_c^{(3)} - 1 + 3^{1-n_c^{(3)}} r. \tag{6.3.4}$$

6.3.2 Average Leaf-to-Leaf Path Length in Complete m-ary Trees

The methodology and discussion of the binary and ternary trees can be generalised to trees of $m > 1$ children, known as *m-ary* trees. The maximal path length for any tree is independent of m and determined entirely by the geometry of the tree. Each leaf node is at depth n, a maximal path has the root node as the lowest common ancestor, therefore the maximal path is $(2n - 1)$.

A recursive function can be obtained using similar logic to before. For a given n, there are m subgraphs with the structure of a tree with $n-1$ levels. When r is less than the size of each subgraph ($r < m^{n-1}$), the sum of the paths is therefore the sum of m copies of the subgraph along with the paths that connect neighbouring pairs. When r is larger than the size of the subgraph ($r \geq m^{n-1}$), the paths are all maximal. When all this is taken into account the recursive function is

$$\mathcal{S}_n^{(m)}(r) = \begin{cases} m\mathcal{S}_{n-1}^{(m)}(r) + (m-1)(2n-1)r, & r < m^{n-1}, \\ (2n-1)(m^n - r), & r \geq m^{n-1}. \end{cases} \tag{6.3.5}$$

This can be solved in the same way as the binary case to obtain an expression for the sum of the paths for a given m, n and r

$$\mathcal{S}_n^{(m)}(r) = m^n \left[2n_c^{(m)} - 1 + \frac{2m^{1-n_c^{(m)}} r}{(m-1)}\right] - \left(2n + \frac{m+1}{m-1}\right) r, \tag{6.3.6}$$

The average path length is then

$$\mathcal{L}_n^{(m)}(r) = \frac{\mathcal{S}_n^{(m)}(r)}{m^n - r}. \tag{6.3.7}$$

and

$$\mathcal{L}_\infty^{(m)}(r) = \left(2n_c^{(m)} - 1 + \frac{2m^{1-n_c^{(m)}} r}{(m-1)}\right). \tag{6.3.8}$$

We note that in analogy with eq. (6.2.7), we have used

$$n_c^{(m)} = \lfloor \log_m r \rfloor + 1 \tag{6.3.9}$$

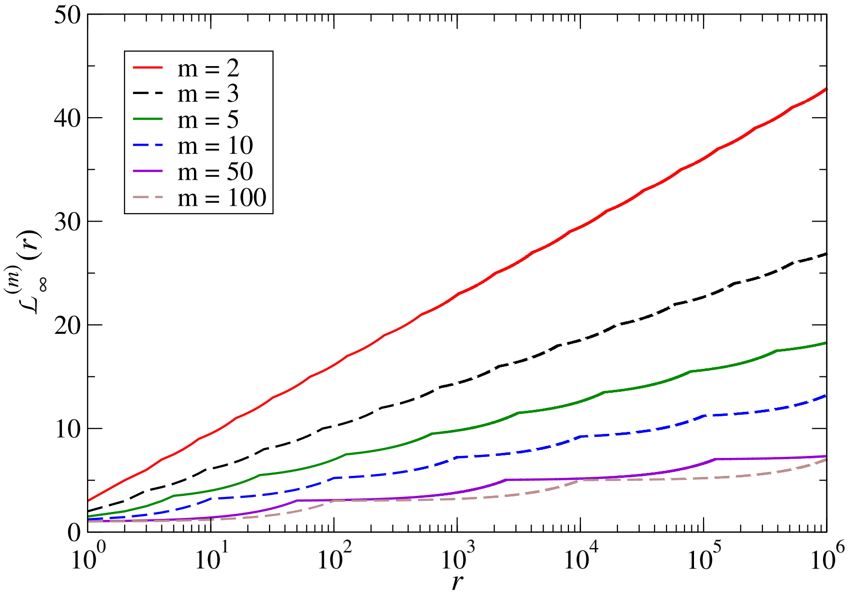

Figure 6.4: Average leaf-to-leaf distance $\mathcal{L}_\infty^{(m)}(r)$ for m-ary trees of various m. The curves for $m = 2, 5, 50$ are shown as solid lines, while those for $m = 3, 10$ and 100 have been indicated as dashed lines for clarity.

in deriving these expressions. Figure 6.4 shows the resulting path lengths in the $n \to \infty$ limit for various values of m.

6.4 Moments of the Leaf-to-Leaf Path Length Distribution in Complete m-ary Trees

6.4.1 Variance of Leaf-to-Leaf Path Lengths in Complete m-ary Trees

In addition to the average path length $\mathcal{L}_n^{(m)}(r)$, it is also of interest to ascertain its variance

$$\mathrm{var}[\mathcal{L}_n^{(m)}](r) = \langle\{\ell_n^{(m)}(r)^2\}\rangle - [\mathcal{L}_n^{(m)}(r)]^2. \qquad (6.4.1)$$

Here $\langle\{\ell_n^{(m)}(r)^2\}\rangle$ denotes the average over the set of all squared paths for given r in an m-ary tree. In order to obtain the variance, we need to obtain an expression for the sum of the squares of path lengths. This can again be done recursively, i.e. with $\mathcal{Q}_n^{(m)}(r)$ denoting this sum of squared leaf-to-leaf path length for an m-ary tree of leaf separation r. Similarly to eq. (6.3.5) we have

$$\mathcal{Q}_n^{(m)}(r) = \begin{cases} m\mathcal{Q}_{n-1}^{(m)}(r) + (m-1)(2n-1)^2 r, & r < m^{n-1}, \\ (2n-1)^2(m^n - r), & r \geq m^{n-1}. \end{cases} \tag{6.4.2}$$

Here, the difference to eq. (6.3.5) is that we have squared the path length terms $(2n-1)$. As before, expanding down to n_c gives a term containing $\mathcal{Q}_{n_c}^{(m)}(r)$,

$$\begin{aligned} \mathcal{Q}_n^{(m)}(r) &= m^{n-n_c}\mathcal{Q}_{n_c}^{(m)}(r) + \sum_{k=0}^{n-n_c-1} m^k r(m-1)[2(n-k)-1]^2 \\ &= m^{n-n_c}\mathcal{Q}_{n_c}^{(m)}(r) + r(m-1) \\ &\quad \times \sum_{k=0}^{n-n_c-1} \left[(4n^2-4n+1)m^k + (4-8n)km^k + 4k^2m^k\right], \end{aligned} \tag{6.4.3}$$

where here and in the remainder of the chapter, we suppress the (m) superscript of $n_c^{(m)}$ for simplicity. As, before the sum can be simplified using equations for geometric series. The first part is a simple geometric series given by eq. (6.2.12). The second part is an arithmetico-geometric series similar to eq. (6.2.11)

$$\sum_{k=0}^{l} kx^k = \frac{x(1-x^l)}{(1-x)^2} - \frac{lx^{l+1}}{1-x}. \tag{6.4.4}$$

The final part is another also an arithmetico-geometric series and has the following form [157]

$$\sum_{k=0}^{l-1} k^2x^k = \frac{1}{(1-x)^3}\left[(-l^2+2l-1)x^{l+2} + (2l^2-2l-1)x^{l+1} - l^2x^l + x^2 + x\right]. \tag{6.4.5}$$

Putting this all together gives

$$\begin{aligned} \mathcal{Q}_n^{(m)}(r) = \frac{1}{(m-1)^2}\Big\{&8rm^{n-n_c+1}\left[n_c(m-1)+1\right] + m^n(m-1)^2(2n_c-1)^2 \\ &-r\left[(2n-1)^2 + (2mn+m)^2 + m(8n^2-6)\right]\Big\}. \end{aligned} \tag{6.4.6}$$

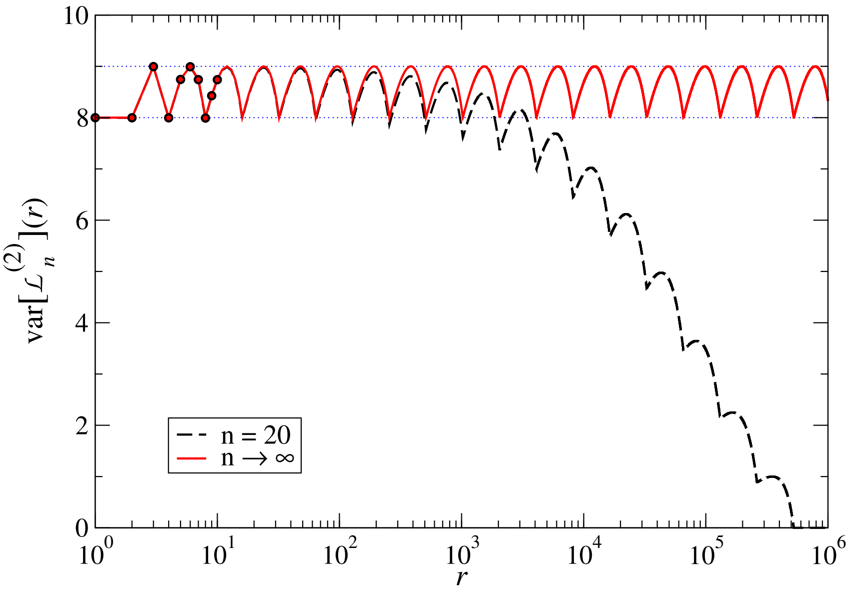

Figure 6.5: Variance $\mathrm{var}[\mathcal{L}_n^{(2)}](r)$ of the path lengths for binary trees. The two lines compare a finite tree ($n = 20$, dashed line) to an infinite tree (solid line). The circles indicate the first 10 $\mathrm{var}[\mathcal{L}_n^{(2)}]$ values similar to fig. 6.3. The two dotted horizontal lines correspond to $\mathrm{var}[\mathcal{L}_n^{(m)}] = 8$ and 9.

We can therefore write for the variance

$$\begin{aligned} \mathrm{var}[\mathcal{L}_n^{(m)}](r) &= \frac{\mathcal{Q}_n^{(m)}(r)}{m^n - r} - \left[\mathcal{L}_n^{(m)}(r)\right]^2 \\ &= \frac{\mathcal{Q}_n^{(m)}(r)}{m^n - r} - \left[\frac{\mathcal{S}_n^{(m)}(r)}{m^n - r}\right]^2 . \end{aligned} \tag{6.4.7}$$

Using eqs. (6.4.6), (6.3.7) and (6.3.6), we then have explicitly

$$\begin{aligned} \mathrm{var}[\mathcal{L}_n^{(m)}](r) = \frac{4r}{m^{2n_c-2}(m^n - r)^2(m-1)^2} \Big(& m^{2n}\Big[m^{n_c-1}(m+1) - r\Big] + m^{2n_c-1}r \\ & - m^n\Big\{m^{n_c-1}(2n - 2n_c + 1)(m-1)r \\ & - m^{2n_c-2}(n_c - n)^2 + m^{2n_c}(n - n_c + 1)^2 \\ & - m^{2n_c-1}\Big[2n^2 - n(4n_c - 2) + 2n_c(n_c - 1) - 1\Big]\Big\}\Big). \end{aligned} \tag{6.4.8}$$

An example for the case of binary trees is given in fig. 6.5. As before we can find

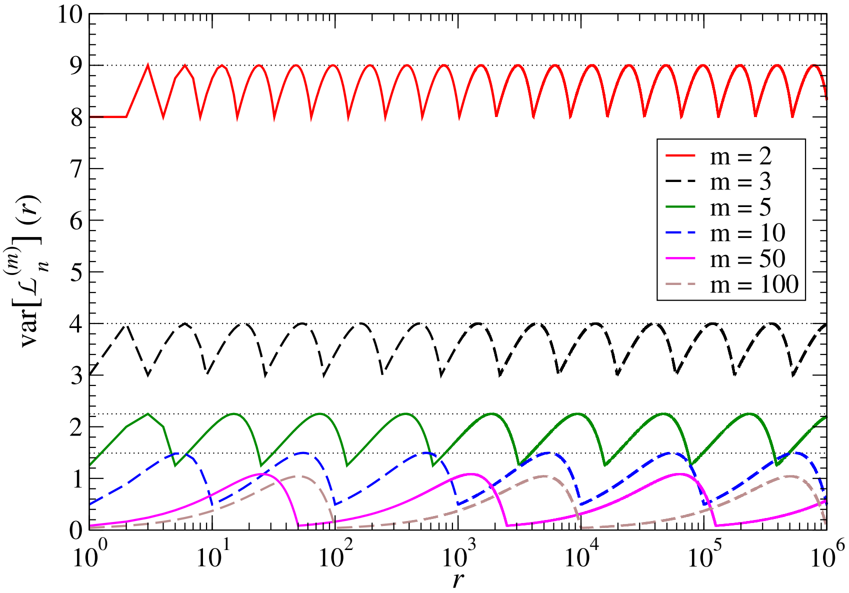

Figure 6.6: Variance in the limit of $n \to \infty$ for various m-ary trees indicates by lines as in fig. 6.4. The 4 dotted horizontal lines correspond to $\mathrm{var}[\mathcal{L}_\infty^{(m)}] = 9, 4, 2.25, 1.49$.

the result in the limit of $n \to \infty$

$$\mathrm{var}[\mathcal{L}_\infty^{(m)}](r) = \frac{4r\left[m^{n_c-1}(m+1) - r\right]}{m^{2n_c-2}(m-1)^2}. \qquad (6.4.9)$$

$\mathrm{var}[\mathcal{L}_\infty^{(m)}](r)$ for various values of m are plotted in fig. 6.6. When $r = m^i$, $i \in \mathbb{N}^0$, then $\mathrm{var}[\mathcal{L}_\infty^{(m)}]$ has local minima with values

$$\mathrm{var}[\mathcal{L}_\infty^{(m)}](m^i) = \frac{4m}{(m-1)^2}. \qquad (6.4.10)$$

Similarly, it can be shown that the local maxima are at $r = \frac{1}{2}m^i(m+1)$, then

$$\mathrm{var}[\mathcal{L}_\infty^{(m)}]\left(\frac{1}{2}m^i(m+1)\right) = \frac{4m}{(m-1)^2} + 1. \qquad (6.4.11)$$

These values are indicated in fig. 6.6 for selected m.

6.4.2 General Moments of Leaf-to-Leaf Path Lengths in Complete m-ary Trees

The derivation in section 6.4.1 suggests that any q-th raw moment of path lengths can be calculated using the same process as for eq. (6.4.2). Indeed, let us define $\mathcal{M}_{q,n}^{(m)}(r)$ as the q-th moment of the distribution of path lengths in an m-ary tree of level n with leaf separation r. Then $\mathcal{M}_{1,n}^{(m)}(r) = \mathcal{L}_n^{(m)}(r)$, $\mathcal{M}_{2,n}^{(m)}(r) = \mathcal{Q}_n^{(m)}(r)$ and

$$\mathrm{var}[\mathcal{L}_n^{(m)}](r) = \frac{\mathcal{M}_{2,n}^{(m)}(r)}{m^n - r} - \left[\frac{\mathcal{M}_{1,n}^{(m)}(r)}{(m^n - r)}\right]^2. \qquad (6.4.12)$$

Following eq. (6.4.2), we find

$$\mathcal{M}_{q,n}^{(m)}(r) = \begin{cases} m\mathcal{M}_{q,n-1}^{(m)}(r) + (2n-1)^q(m-1)r, & r < m^{n-1}, \\ (2n-1)^q(m^n - r), & r \geq m^{n-1}. \end{cases} \qquad (6.4.13)$$

By expanding, this gives

$$\mathcal{M}_{q,n}^{(m)}(r) = m^{n-n_c}\mathcal{M}_{q,n_c}^{(m)}(r) + \sum_{k=0}^{n-n_c-1} m^k(m-1)\left[2(n-k)-1\right]^q r. \qquad (6.4.14)$$

As before, n_c corresponds to the first n value where, for given r, we have to use the second part of the expansion as in eq. (6.4.13). Hence we can substitute the second part of (6.4.14) for $\mathcal{M}_{q,n_c-1}^{(m)}(r)$ giving

$$\mathcal{M}_{q,n}^{(m)}(r) = m^{n-n_c}(2n_c-1)^q(m^{n_c}-r) + \sum_{k=0}^{n-n_c-1} m^k(m-1)\left[2(n-k)-1\right]^q r. \qquad (6.4.15)$$

In order to derive an explicit expression for this similar to section 6.2.2, we need again to study the final sum of eq. (6.4.15). We write

$$\begin{aligned} &\sum_{k=0}^{n-n_c-1} m^k(m-1)\left[2(n-k)-1\right]^q r \\ &\quad = r(m-1)(-2)^q\left[\sum_{k=0}^{\infty} m^k\left(k-n+\frac{1}{2}\right)^q - \sum_{k=n-n_c}^{\infty} m^k\left(k-n+\frac{1}{2}\right)^q\right] \\ &\quad = r(m-1)(-2)^q\left[\sum_{k=0}^{\infty} m^k\left(k-n+\frac{1}{2}\right)^q - m^{n-n_c}\sum_{k=0}^{\infty} m^k\left(k-n_c+\frac{1}{2}\right)^q\right] \\ &\quad = r(m-1)(-2)^q\left[\Phi\left(m,-q,\frac{1}{2}-n\right) - m^{n-n_c}\Phi\left(m,-q,\frac{1}{2}-n_c\right)\right], \end{aligned} \qquad (6.4.16)$$

where in the last step we have introduced the *Hurwitz-Lerch Zeta function* Φ [158, 159] (also referred to as the *Lerch transcendent* [160] or the *Hurwitz-Lerch Transcendent* [161]). It is defined as the sum

$$\Phi(z,s,u) = \sum_{k=0}^{\infty} \frac{z^k}{(k+u)^s}, \quad z \in \mathbb{C}. \tag{6.4.17}$$

The properties of $\Phi(z,s,u)$ are [160]

$$\Phi(z,s,u+1) = \frac{1}{z}\left(\Phi(z,s,u) - \frac{1}{u^s}\right), \tag{6.4.18}$$

$$\Phi(z,s-1,u) = \left(u + z\frac{\partial}{\partial z}\right)\Phi(z,s,u), \tag{6.4.19}$$

$$\Phi(z,s+1,u) = -\frac{1}{s}\frac{\partial \Phi}{\partial u}(z,s,u). \tag{6.4.20}$$

Hence we can write

$$\mathcal{M}^{(m)}_{q,n}(r) = m^{n-n_c}(2n_c-1)^q(m^{n_c}-r)+ \tag{6.4.21}$$
$$r(m-1)(-2)^q\left[\Phi\left(m,-q,\frac{1}{2}-n\right) - m^{n-n_c}\Phi\left(m,-q,\frac{1}{2}-n_c\right)\right].$$

Averages of $\mathcal{M}^{(m)}_{,n}(r)$ can be defined as previously via

$$\mathcal{A}^{(m)}_{q,n}(r) = \frac{\mathcal{M}^{(m)}_{q,n}(r)}{m^n - r} \tag{6.4.22}$$

such that $\mathcal{L}^{(m)}_n(r) = \mathcal{A}^{(m)}_{1,n}(r)$ and $\mathrm{var}[\mathcal{L}^{(m)}_n](r) = \mathcal{A}^{(m)}_{2,n}(r) - \left[\mathcal{A}^{(m)}_{1,n}(r)\right]^2$.

In order to look at the $n \to \infty$ limit of $\mathcal{A}^{(m)}_{q,n}(r)$ it is necessary to see how Φ behaves in this limit. From eq. (6.4.19)

$$\Phi\left(m,-(q+1),\left(\frac{1}{2}-n\right)\right) = \left[\left(\frac{1}{2}-n\right) + m\frac{\partial}{\partial m}\right]\Phi\left(m,-q,\left(\frac{1}{2}-n\right)\right), \tag{6.4.23}$$

and the sum of a geometric series [157]

$$\Phi\left(m,0,\left(\frac{1}{2}-n\right)\right) = \frac{1}{1-m}, \tag{6.4.24}$$

we conclude that for finite q

$$\Phi\left(m,-q,\left(\frac{1}{2}-n\right)\right) \sim \mathcal{O}(n^q). \tag{6.4.25}$$

Therefore

$$\lim_{n\to\infty} \frac{r(m-1)(-2)^q \Phi\left(m,-q,\frac{1}{2}-n\right)}{m^n - r} = 0, \tag{6.4.26}$$

In the $n \to \infty$ limit, we find

$$\begin{aligned}\lim_{n\to\infty} \mathcal{A}^{(m)}_{q,n}(r) &\equiv \mathcal{A}^{(m)}_{q,\infty}(r)\\ &= m^{-n_c}\left[(2n_c-1)^q(m^{n_c}-r) - r(m-1)(-2)^q\Phi\left(m,-q,\frac{1}{2}-n_c\right)\right].\end{aligned} \tag{6.4.27}$$

6.5 Complete m-ary Trees with Periodicity

Up to now we have always dealt with trees in which the maximum separation r was set by the number of leaves, i.e. $r \leq m^n$. This represents *open* boundary in terms of physical systems. A *periodic* boundary can be realised by having the leaves of the tree form a circle as depicted in fig. 6.7 for a binary tree. For such a tree, only separations $r \leq L/2$ are relevant since all cases with $r > L/2$ can be reduced to smaller $r = \mathrm{mod}(r, L/2)$ values by going around the periodic tree in the opposite direction. Therefore we can write

$$\mathcal{M}^{(m,\circ)}_{1,n}(r) = \mathcal{M}^{(m)}_{1,n}(r) + \mathcal{M}^{(m)}_{1,n}(m^n - r), \tag{6.5.1}$$

where $r < L/2$ and the subscript $\circ$ denotes the periodic case. Note that the case where $r = L/2$ the clockwise and anti-clockwise paths are the same so only need to be counted once. In the simple binary tree case we can expand this via (6.2.14) as in section 6.2.2 and find

$$\begin{aligned}\mathcal{M}^{(2,\circ)}_{1,n}(r) &\equiv \mathcal{S}^{(2,\circ)}_{n}(r)\\ &= 2^n\left[2n_c + 2\tilde{n}_c - 2n - 5 + 2^{2-n_c}r + 2^{2-\tilde{n}_c}(2^n - r)\right],\end{aligned} \tag{6.5.2}$$

with n_c as in eq. (6.2.7) and $\tilde{n}_c = \lfloor \log_2(2^n - r)\rfloor + 1$. For every r, we have 2^n possible starting leaf positions on a periodic binary tree and hence the average leaf-to-leaf path length can be written as

$$\begin{aligned}\mathcal{A}^{(2,\circ)}_{1,n}(r) \equiv \mathcal{L}^{(2,\circ)}_{n}(r) &= \frac{\mathcal{S}^{(2,\circ)}_{n}(r)}{2^n}\\ &= 2n_c + 2\tilde{n}_c - 2n - 5 + 2^{2-n_c}r + 2^{2-\tilde{n}_c}(2^n - r).\end{aligned} \tag{6.5.3}$$

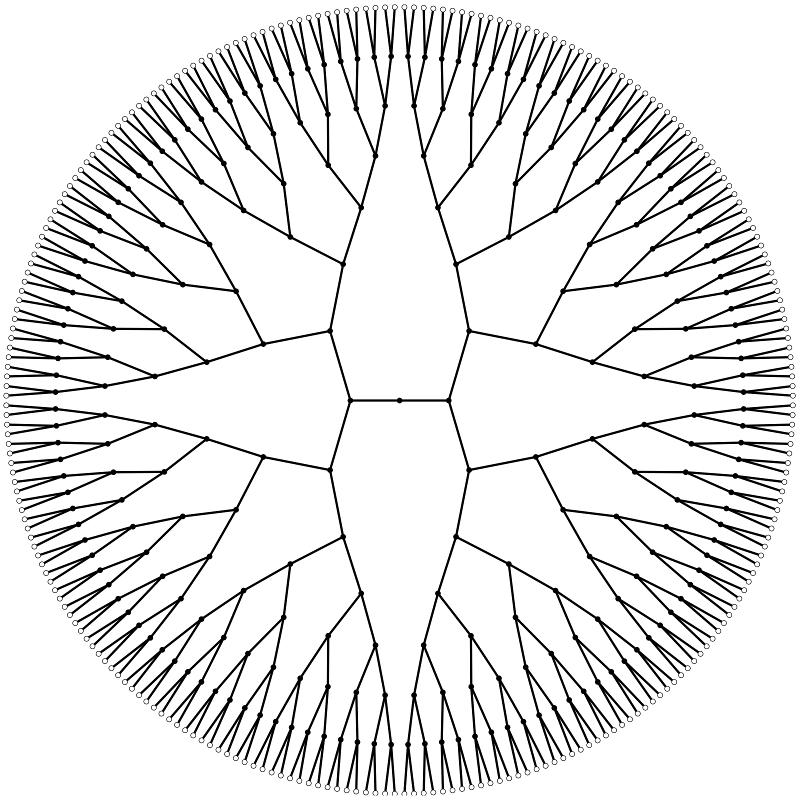

Figure 6.7: A periodic, complete, binary tree with $n = 8$ levels. Circles and lines as in fig. 6.1.

This expression is the periodic analogue to eq. (6.2.15). Generalizing to m-ary trees, with $\tilde{n}_c = \lfloor \log_m(m^n - r) \rfloor + 1$, we find

$$\begin{aligned}
\mathcal{M}_{1,n}^{(m,\circ)}(r) &= \mathcal{M}_{1,n}^{(m)}(r) + \mathcal{M}_{1,n}^{(m)}(m^n - r) \\
&= m^n \left[2n_c + 2\tilde{n}_c - 2n - 1 + \frac{2}{m-1} \left(m^{1-n_c} r + m^{1-\tilde{n}_c}(m^n - r) - m \right) \right].
\end{aligned} \tag{6.5.4}$$

The average path length for m-ary periodic trees is then given as

$$\begin{aligned}
\mathcal{A}_{1,n}^{(m,\circ)}(r) &= \frac{\mathcal{M}_{1,n}^{(m,\circ)}(r)}{m^n} \\
&= \left[2n_c + 2\tilde{n}_c - 2n - 1 + \frac{2}{m-1} \left(m^{1-n_c} r + m^{1-\tilde{n}_c}(m^n - r) - m \right) \right].
\end{aligned} \tag{6.5.5}$$

To again study the case of $n \to \infty$, it is necessary to observe how $\tilde{n}_c$ behaves for large n and fixed m, r. When $n \gg r$, we have $r < m^{n-1}$ and hence

$$\lim_{n\to\infty} \lfloor \log_m(m^n - r) \rfloor = n - 1. \tag{6.5.6}$$

This enables us to simply take the limits of eq. (6.5.5) to give

$$\lim_{n\to\infty} \mathcal{A}^{(m,\circ)}_{1,n}(r) \equiv \mathcal{A}^{(m,\circ)}_{1,\infty}(r) = 2n_c - 1 + \frac{2m^{1-n_c}r}{(m-1)}, \tag{6.5.7}$$

which is the same as the OBC case (6.3.8). This is to be expected as a small region of a large circle can be approximated by a straight line.

Lastly, the q-moments can be expressed similarly to eq. (6.4.21) via the Lerch transcendent as

$$\begin{aligned}\mathcal{M}^{(m,\circ)}_{q,n}(r) &= \mathcal{M}^{(m)}_{q,n}(r) + \mathcal{M}^{(m)}_{q,n}(m^n - r),\\ &= m^{n-n_c}(2n_c-1)^q(m^{n_c} - r) + m^{n-\tilde{n}_c}(2\tilde{n}_c - 1)^q(m^{\tilde{n}_c} - m^n + r)\\ &\quad + (m-1)(-2)^q\left[m^n\Phi\left(m, -q, \frac{1}{2} - n\right) - rm^{n-n_c}\Phi\left(m, -q, \frac{1}{2} - n_c\right)\right.\\ &\quad \left. -(m^n - r)m^{n-\tilde{n}_c}\Phi\left(m, -q, \frac{1}{2} - \tilde{n}_c\right)\right],\end{aligned} \tag{6.5.8}$$

The average q-moments in full are therefore

$$\mathcal{A}^{(m,\circ)}_{q,n}(r) = \frac{\mathcal{M}^{(m,\circ)}_{q,n}(r)}{m^n} \tag{6.5.9}$$

for a complete, periodic, m-ary tree. To take the limit $n \to \infty$ notice that $\tilde{n}_c = n$ when $r < m^{n-1}$ for large n. Just like with eq. (6.5.7), this results in $\mathcal{A}^{(m,\circ)}_{q,\infty}(r) = \mathcal{A}^{(m)}_{q,\infty}(r)$.

6.6 Asymptotic Scaling of the Correlation for a Homogeneous Tree Tensor Network

To illustrate where the properties calculated for complete trees may arise in tensor network simulations we aim to construct a tensor network wavefunction that has this tree structure and calculate its correlation functions. A tensor network that has the structure of a complete tree graph is known as a TTN and is often used to model critical one dimensional quantum lattice systems due to the fact that they can be efficiently updated [43, 126]. It is, however, not clear if there is a Hamiltonian that

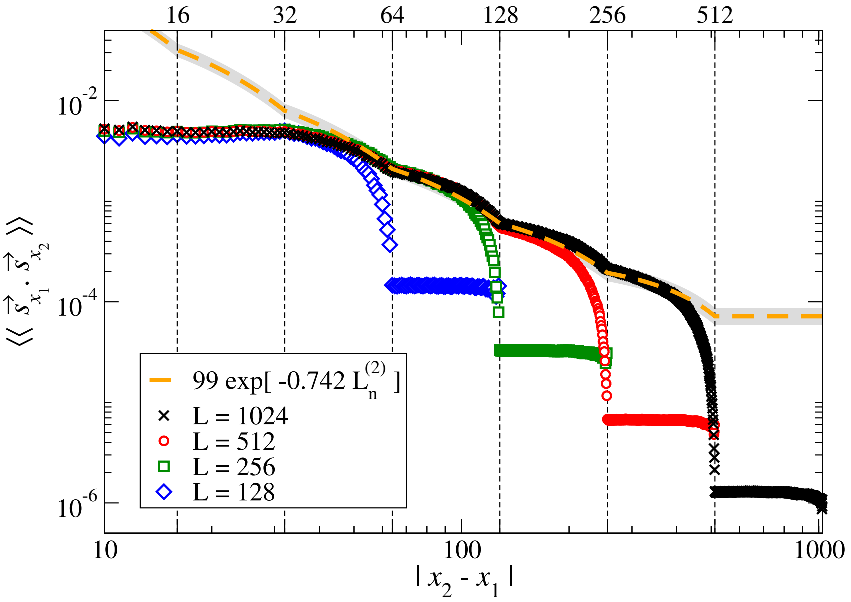

Figure 6.8: Two point correlation function for TTNs with $\chi = 4$ averaged over all pairs of sites separated by $|x_2 - x_1|$ as discussed in text. The TTNs have $L = 128$ (blue diamonds), 256 (green squares), 512 (red circles), 1024 (black crosses) corresponding to n = 7, 8, 9, 10 levels respectively. The vertical dashed lines highlight $|x_2 - x_1|$ = 16, 32, 64, 128, 256, 512. The orange dashed line corresponds to a fit of $A\ \exp[-\alpha\mathcal{L}_n^{(2)}(r)]$ with $A = 99 \pm 9$ and $\alpha = 0.742 \pm 0.006$. The grey shaded region is the standard error on the fit.

admits a wavefunction with a TTN structure as its ground state. In principle it is possible to start from a tensor network wavefunction and derive a *parent Hamiltonian* for which the wavefunction is a ground state [101, 162]. In the case of homogeneous TTNs the procedure to create such a parent Hamiltonian seems likely to be highly non-trivial and not unique. Here we build such a TTN from the *binary* tree structure shown in fig. 6.1. At each internal vertex we place an isometric tensor [12, 34] with initially random entries and so-called bond dimension $\chi = 4$. Using as proxy a spin-1/2 Heisenberg model, given by eq. (2.6.22), we perform energy minimisation [34, 43] at a bulk site. After each minimization, we replicate the bulk tensor to all other tensors such that every isometry is kept identical [30]. The process is then repeated until convergence (in energy).

A two-point correlation function $\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle$ is calculated [12, 43] for all pairs of sites and averaged for all points separated by $|x_2 - x_1|$. The results are given in fig. 6.8. As discussed in chapter 4 the two point correlation function is expected to scale as

$$C(x_1, x_2) \sim \exp[-\alpha D_{TN}(x_1, x_2)], \tag{6.6.1}$$

where α is a constant and $D_{TN}(x_1, x_2)$ is the number of tensor connecting sites x_1 and x_2. Hence we expect the asymptotic correlation function to scale as $\sim \exp\left[-\alpha \mathcal{L}_n^{(2)}(r)\right]$. Figure 6.8 shows that, away from small separations (e.g. $|x_2 - x_1| > 32$ for $L = 1024$), the content of the tensors no longer dominates the structural contribution and $\langle \vec{s}_{x_1} \cdot \vec{s}_{x_2} \rangle$ exhibits many of the properties we find in fig. 6.3. The overall form of the long range correlations is a power law but there are also the characteristic fluctuations from the self-similar structure of the tree with cusps at $|x_2 - x_1| = 2^i$ for integer $i \geq 5$ (corresponding to $|x_2 - x_1| > 32$). When reaching the finite-size dominated regime $|x_2 - x_1| \geq \frac{L}{2}$, we find an approximate constant average correlation. This is smaller than expected from eq. (6.2.16) because the top tensor of the TTN only has $\chi = 1$ and contributes less to the correlation function than the other tensors. We emphasize that we have chosen a low bond dimension $\chi = 4$ so that we can study the asymptotic form of the correlation functions for smaller system sizes.

The form of the correlations expressed in fig. 6.8 corresponds to those of a suitable parent Hamiltonian, i.e. one that has a ground state implied by this holographic tree structure. In addition, the results may also be useful for those building TTNs as a variational method for the study of critical systems. The appearance of this form of the correlation for models that do not have a natural tree structure in the wavefunction, such as the Heisenberg model, is an indicator that the chosen χ is too small to capture the physics of the model. This is similar to the erroneous exponential decay of correlation functions found by DMRG for critical systems with power-law correlations in case of small χ [30], as illustrated by fig. 4.2(b). In these situations the structure of the network dominates the value of the correlation rather than the information in the tensors.

6.7 Conclusions

We have calculated an analytic form for the average path length between two leaves with a given separation — ordered according to the physical distance long a line — in a complete binary tree graph. This result is then generalised to a complete tree where each vertex has any finite number of children. In addition to the mean leaf-to-leaf path length, it is found that the raw moments of the distribution of path lengths have an analytic form that can be expressed in a concise way in terms of the Hurwitz-Lerch Zeta function. These findings are calculated for open trees, where the leaves form an open line, periodic trees, where the leaves form a circle, and infinite trees, which is the limit where the number of levels, n, goes to infinity. Each of these results has a concise form and characteristic features due to the self-similarity of the trees. We believe that these results provide a useful insight into the structure of the regular tree graphs that are relevant for the field of tensor networks [153, 154]. We also note that path lengths computed here are qualitatively similar, but quantitatively different from those for the random-spin chains [12] described in chapter 5. This points to a subtle, yet physically relevant, difference in their Hilbert space properties.

Chapter 7

Leaf-to-Leaf Path Lengths in Full Binary Trees

7.1 Introduction

In chapter 6 [13] we analysed the leaf-to-leaf path lengths in complete trees as an introduction to the study of the structure of the tensor networks formed by the tSDRG algorithm of chapter 5. However the trees constructed by tSDRG [12] are very different. In fig. 7.1 we show an example of binary tree created by tSDRG. Here the leaves do not all appear at the same level, but rather each node can become a leaf node according to how the couplings are renormalised.

In this chapter we will extend the analysis of tree graphs to full binary trees in the form of the *Catalan Trees* and randomly generated full binary trees. In this context a *full binary tree* is a tree graph where every internal vertex has exactly two children [155] and a *leaf* or a *leg* is a terminal vertex with no children. We define n as the number of internal vertices in the graph, which we shall refer to *nodes* or simply *vertices*. The *root* is the top vertex of the tree, which is unique as it is not the child of any vertex. An *edge* is the line connecting any two vertices. The *leaf-to-leaf path length* as before is the number of vertices that connect two leaves. These definitions are shown pictorially in fig. 7.1.

7.2 Introduction to Catalan Trees

To analytically analyse trees that more closely match those found in chapter 5 we concentrate on the set of all *unique* full binary trees. This set is known as the set of *Catalan trees* [143] for reasons that will become clear as we progress. Given a

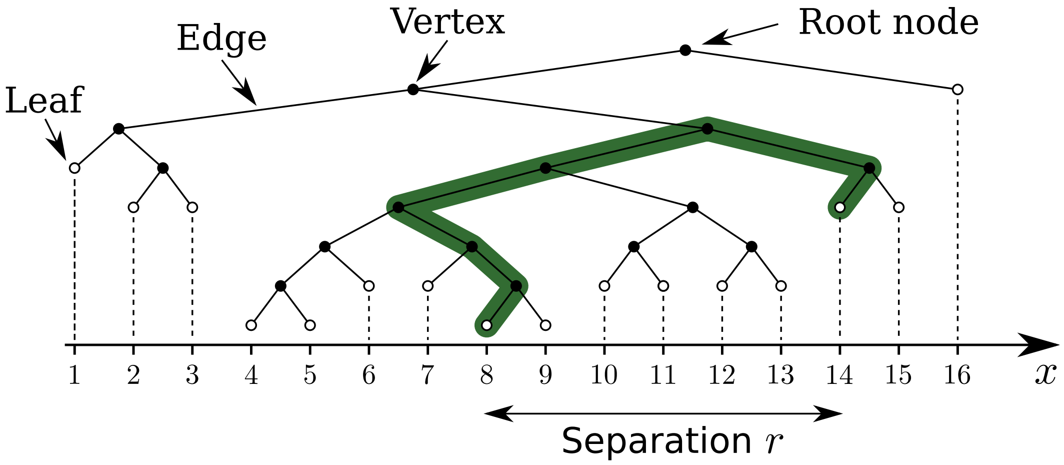

Figure 7.1: A random binary tree created by the tSDRG algorithm of chapter 5. As before the filled (open) circles are internal vertices (leaves) and the connecting lines are edges. The tree as shown has $n = 15$ and 16 leaves ($\circ$). The indicated separation is $r = 6$ while the associated leaf-to-leaf path length equals $\ell = 6$ as indicated by the thick line. The thick green line denotes the path.

specific number of internal vertices n, there are a finite number of unique trees as shown in fig. 7.2 for 1, 2 and 3 vertices. This set of trees can be decomposed in terms of *primary sub-trees* with p and q vertices where $p + q + 1 = n$ as shown by the dashed boxes in fig. 7.2. Here a *sub-tree* rooted at vertex v is the set of vertices and leaves that descend from v, including v itself. A *primary sub-tree* is then a sub-tree rooted at one of the children of the root node. The generalisation of this observation allows us to use a convenient diagrammatic decomposition as given by fig. 7.3. We emphasise that unlike in chapter 6, n is the number of internal vertices, not the number of levels. The number of leaves in a full binary tree, and therefore the length of the effective lattice, can be shown [143] to be $L = n + 1$.

From the diagrammatic decomposition of a tree of n vertices as in fig. 7.3 it is possible to count the number of unique binary trees. The number of trees in each subset is the number in the left subtree C_α times the number in the right sub-tree $C_{n-\alpha-1}$. When summed over all of the subsets we obtain

$$C_n = \sum_{\alpha=0}^{n-1} C_\alpha C_{n-\alpha-1}, \tag{7.2.1}$$

where $C_0 = 1$ by definition. This is *Segner's recurrence relation* and can be used as

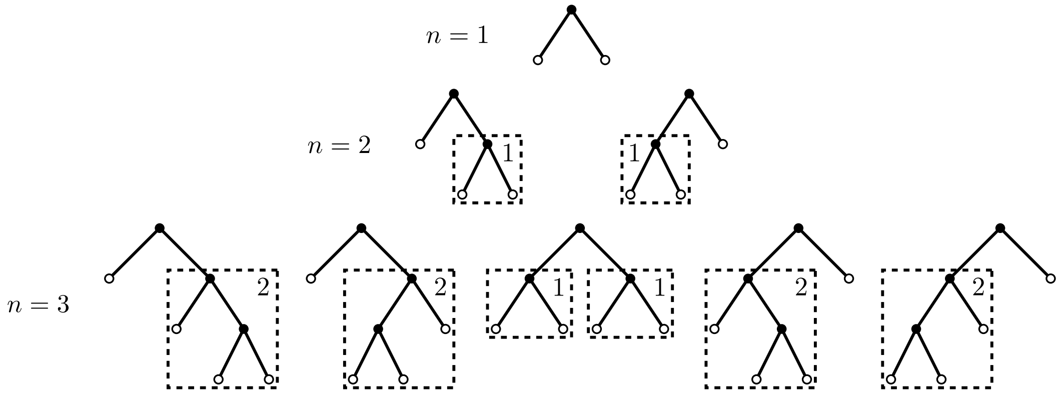

Figure 7.2: The set of all unique full binary trees with 1, 2 and 3 vertices. The dashed boxes denote the sub-trees that make up the full set. Lines and symbols are as in fig. 7.1.

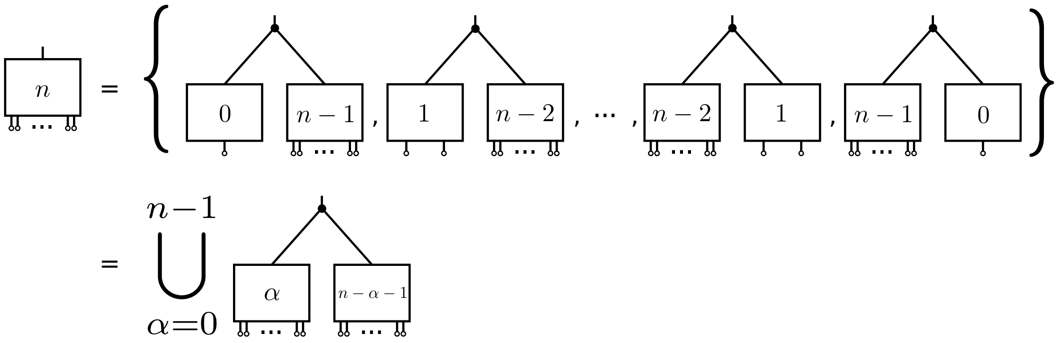

Figure 7.3: Diagrammatic decomposition of the set of trees with n vertices in terms of primary sub-trees whose vertices add up to $n-1$.

the definition of a *Catalan number* C_n [163]. Therefore the number of unique trees with n vertices is given by the *Catalan number* C_n, hence the name *Catalan trees.*

7.3 Properties of Catalan Numbers

Before we begin the analysis of the Catalan trees it will prove useful to introduce some properties of the *Catalan numbers.* The Catalan numbers are described by Koshy [163] to be “like the North Star in the evening sky, ... a beautiful and bright light in the mathematical heavens” due to their “ubiquitousness [and] tendency to appear in quite unexpected and unrelated places”. They first appeared in China in the early 18th century in works of Ming in deriving series expansions for the sine function [164]. In the West the Catalan numbers originate in Euler's *polygon triangulation problem* that he sent to other mathematicians in around 1751 [163, 165]. The first publication was by von Segner and Euler in the same edition of Novi Commentarii Academiae Scientiarum Imperialis Petropolitanae in 1761 [165, 166, 167]. The Catalan numbers became a named series following the publication of an article by Catalan in 1838 [165, 168]. Since then the Catalan numbers have appeared in a surprisingly large number of problems, many of which are found in ref. [163].

The first few Catalan numbers are $\{1, 1, 2, 5, 14, 42, 132, \dots\}$. There are several general ways that they can be expressed, for example eq. (7.2.1) and

$$C_n = \frac{1}{n+1}\binom{2n}{n} = \frac{(2n)!}{(n+1)!n!}. \tag{7.3.1}$$

They are also recursive and follow the rule

$$C_n = \frac{4n-2}{n+1}C_{n-1}. \tag{7.3.2}$$

In the remainder of the chapter we make extensive use of *generating function* methods [169]. These are a convenient means of viewing and manipulating number series. A generating function is a function such that the n-th number in the series is the n-th coefficient in a Taylor expansion of the function

$$a(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \cdots = \sum_{n=0}^{\infty} a_n x^n. \tag{7.3.3}$$

The generating function for the Catalan numbers is

$$\mathcal{C}(x) = \sum_{n=0}^{\infty} C_n x^n. \tag{7.3.4}$$

It is possible to find an explicit functional form for this generating function as follows. Squaring eq. (7.3.4) gives a form comparable to Segner's relation (7.2.1)

$$\begin{aligned}\mathcal{C}^2(x) &= C_0^2 + (C_0C_1 + C_1C_0)x + \cdots + (C_0C_n + C_1C_{n-1} + \cdots + C_nC_0)x^n + \dots \\ &= C_1 + C_2x + C_3x^2 + \cdots + C_{n+1}x^n + \dots \\ &= \frac{(\mathcal{C}(x) - C_0)}{x}. \end{aligned} \tag{7.3.5}$$

Multiplying by x gives a quadratic eq. that can easily be solved

$$x\mathcal{C}^2(x) - \mathcal{C}(x) + 1 = 0. \tag{7.3.6}$$

Hence

$$\mathcal{C}(x) = \frac{1 \pm \sqrt{1-4x}}{2x}. \tag{7.3.7}$$

We choose the minus to make $C(0)$ finite and the rest of the Catalan numbers positive. Expanding the square root using [170]

$$\sqrt{1+x} = \sum_{n=0}^{\infty} \binom{1/2}{n} x^n, \tag{7.3.8}$$

where [171]

$$\binom{1/2}{k} = \binom{2k-2}{k-1} \frac{(-1)^{k-1}}{k2^{2k-1}}. \tag{7.3.9}$$

This can be rearranged to give

$$\begin{aligned} \sqrt{1+x} &= \sum_{n=0}^{\infty} \frac{(-1)^n (2n)!}{(1-2n)(n!)^2(4^n)} x^n \\ &= 1 - 2\sum_{n=1}^{\infty} \binom{2n-2}{n-1} \left(\frac{-1}{4}\right)^n \frac{x^n}{n}, \end{aligned} \tag{7.3.10}$$

Hence the generating function (7.3.7) can be written as

$$\begin{aligned} \frac{1-\sqrt{1-4x}}{2x} &= \frac{1}{2x}\left[1 - 1 + 2\sum_{n=1}^{\infty} \frac{1}{n}\binom{2n-2}{n-1} x^n\right] \\ &= \sum_{n=1}^{\infty} \frac{1}{n}\binom{2n-2}{n-1} x^{n-1} \\ &= \sum_{n=0}^{\infty} \frac{1}{n+1}\binom{2n}{n} x^n. \end{aligned} \tag{7.3.11}$$

Comparison with eq. (7.3.4) yields the explicit binomial form shown in eq. (7.3.1). Expansion of the binomial coefficient in terms of factorials

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}, \tag{7.3.12}$$

then obtains the factorial form of eq. (7.3.1).

7.4 Leg Depths

To calculate leaf-to-leaf path lengths using a recursive approach similar to that in chapter 6 it is necessary to calculate the average depth of each leaf. This is trivial for the complete trees of chapter 6 as all of the leaves have the same depth by definition. For the case of the Catalan trees this is not so obvious as each individual tree is different. We define m as the horizontal position of a leaf in a tree or sub-tree, with $m = 1$ being the left-most leaf. We emphasise that this is not necessarily the same as the *lattice* position (x) as that is a global position for the whole tree, and does not apply for sub-trees. Also all daughters of the left sub-tree are always to the left of daughters of the right sub-tree. This is true at all levels and for all sub trees. Define the *depth* of a leg as the number of internal vertices that connect the leg to the root node (including the root node). The *depth function* $D_{m,n}$ is then defined as the total number of vertices that connect the leg m to the root when summing over all possible diagrams of n vertices, or

$$D_{m,n} = \sum_{t \in \tau_n} \text{depth of } m\text{th leg}, \tag{7.4.1}$$

where t is a Catalan tree and τ_n is the set of all Catalan trees with n vertices.

7.4.1 Depth of the First Leg

Due to the fact that the set of trees with n vertices can be decomposed in terms of trees with fewer vertices (fig. 7.4), the sum of the depths of the first leg can be expressed in terms of the sub-trees. $D_{1,n}$ is equal to the sum of the first leg depths of the left hand sub-tree multiplied by the degeneracy of the right sub-tree, plus the number of vertices that connect the sub-trees

$$\begin{aligned} D_{1,n} &= \sum_{k=0}^{n-1} \left[D_{1,n-1-k} C_k + C_{n-1-k} C_k \right] \\ &= \sum_{k=0}^{n-1} D_{1,n-1-k} C_k + C_n, \end{aligned} \tag{7.4.2}$$

where in the last line we have used Segner's relation (7.2.1).

In a similar manner to the Catalan numbers (7.3.5), we can define a generating function for the depth

$$\mathcal{D}_1(x) = \sum_{n=0}^{\infty} D_{1,n} x^n. \tag{7.4.3}$$

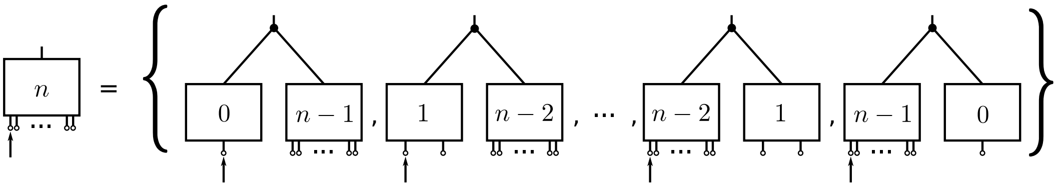

Figure 7.4: Schematic representation of the contribution to the first leaf depth from each of the sub-trees of the decomposition in fig. 7.3.

Multiplying by $\mathcal{C}(x)$ gives us an expression of the form of (7.4.2)

$$\begin{aligned}\mathcal{D}_1(x)\mathcal{C}(x) &= D_{1,0}C_0 + (D_{1,0}C_1 + D_{1,1}C_0)x + \dots \\ &\qquad \dots + (D_{1,0}C_n + D_{1,1}C_{n-1} + \dots + D_{1,n}C_0)x^n + \dots \\ &= (D_{1,1} - C_1) + (D_{1,2} - C_2)x + \dots + (D_{1,n+1} - C_{n+1})x^n + \dots \\ &= \sum_{n=0}^{\infty} D_{1,n+1}x^n - \sum_{n=0}^{\infty} C_{n+1}x^n \\ &= \frac{(\mathcal{D}_1(x) - D_{1,0}) - (\mathcal{C}(x) - C_0)}{x}. \end{aligned} \tag{7.4.4}$$

Collecting the $\mathcal{D}_1(x)$ terms

$$\mathcal{D}_1(x) = \frac{C_0 - D_{1,0} - \mathcal{C}(x)}{x\mathcal{C}(x) - 1} \tag{7.4.5}$$

and using $\mathcal{C}(x) = \frac{1}{1 - x\mathcal{C}(x)}$ (see (A.2.1) derived in the appendix) gives

$$\begin{aligned}\mathcal{D}_1(x) &= \mathcal{C}^2(x) + (D_{1,0} - C_0)\mathcal{C}(x) \\ &= \mathcal{C}^2(x) - \mathcal{C}(x),\end{aligned} \tag{7.4.6}$$

where we have used $C_0 = 1$ and $D_{1,0} = 0$. This can be put back into the form of a generating function using $\mathcal{C}^2(x) = \sum_{n=0}^{\infty} C_{n+1}x^n$ (see (A.2.5))

$$\mathcal{D}_1(x) = \sum_{n=0}^{\infty} \left[C_{n+1} - C_n\right] x^n, \tag{7.4.7}$$

which implies our result

$$D_{1,n} = C_{n+1} - C_n. \tag{7.4.8}$$

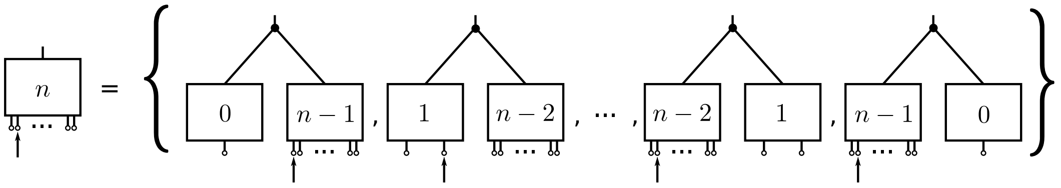

Figure 7.5: Schematic representation of the contribution to the depth of the second leaf from each of the sub-trees. The second leg is highlighted showing that in the $(0, n-1)$ subset, the second leg is in the right hand sub-tree.

There are C_n graphs, thus the average depth is

$$d_{1,n} = \frac{C_{n+1} - C_n}{C_n} = \frac{3n}{n+2}, \tag{7.4.9}$$

where we have used the recursion relation (7.3.2). In the limit of $n \to \infty$

$$d_{1.\infty} \equiv \lim_{n\to\infty} d_{1,n} = 3. \tag{7.4.10}$$

7.4.2 Depth of the Second Leg

Figure 7.5 shows that when the n vertex tree is decomposed in terms of the primary sub-trees, the second leg on the $(0, n-1)$ subset is no longer in the left hand sub-tree. Therefore the recursion relation for the second leg depth is

$$D_{2,n} = \sum_{k=1}^{n-1} D_{2,k} C_{n-1-k} + D_{1,n-1} + C_n. \tag{7.4.11}$$

As before, we create a generating function for the depths, but here the sum starts at 1 to reflect the fact that the second leg depth is not defined for a tree with one leg

$$\mathcal{D}_2(x) = \sum_{n=1}^{\infty} D_{2,n} x^n. \tag{7.4.12}$$

As with the first leg, we multiply by $\mathcal{C}(x)$ and compare with recursion relation (7.4.11)

$$\begin{aligned}\mathcal{D}_2(x)\mathcal{C}(x) &= D_{2,1}C_0x + (D_{2,1}C_1 + D_{2,2}C_0)x^2 + \dots \\ &\qquad\qquad \dots + (D_{2,1}C_{n-1} + \dots + D_{2,n}C_0)x^n + \dots \\ &= (D_{2,2} - D_{1,1} - C_2)x + \dots + (D_{2,n+1} - D_{1,n} - C_{n+1})x^n + \dots \\ &= \sum_{n=1}^{\infty} D_{2,n+1}x^n - \sum_{n=1}^{\infty} D_{1,n}x^n - \sum_{n=1}^{\infty} C_{n+1}x^n \\ &= \left(\frac{\mathcal{D}_2(x) - D_{2,1}x}{x}\right) - (\mathcal{D}_1(x) - D_{1,0}) - \left(\frac{\mathcal{C}(x) - C_1x - C_0}{x}\right).\end{aligned} \tag{7.4.13}$$

Using eq. (A.2.1) and the fact that $C_0 = 1$, $C_1 = 1$, $D_{1,0} = 0$ and $D_{2,1} = 1$, we can write

$$\mathcal{D}_2(x) = x\mathcal{C}(x)\mathcal{D}_1(x) + \mathcal{C}^2(x) - \mathcal{C}(x). \tag{7.4.14}$$

Then from the previous results: $\mathcal{D}_1(x) = \mathcal{C}^2(x) - \mathcal{C}(x)$ and $x\mathcal{C}^2(x) = \mathcal{C}(x) - 1$ we obtain

$$x\mathcal{D}_1(x) = \mathcal{C}(x) - x\mathcal{C}(x) - 1. \tag{7.4.15}$$

Applying this to eq. (7.4.13) gives

$$\mathcal{D}_2(x) = 2\mathcal{C}^2(x) - 2\mathcal{C}(x) - x\mathcal{C}^2(x). \tag{7.4.16}$$

Writing this as a summation

$$\begin{aligned}\mathcal{D}_2(x) &= \sum_{n=0}^{\infty} \left[2C_{n+1} - 2C_n\right] x^n - \sum_{n=1}^{\infty} C_nx^n \\ &= \sum_{n=1}^{\infty} \left[2C_{n+1} - 3C_n\right] x^n,\end{aligned} \tag{7.4.17}$$

implies the result

$$D_{2,n} = 2C_{n+1} - 3C_n. \tag{7.4.18}$$

The average depth is

$$d_{2,n} = \frac{2C_{n+1} - 3C_n}{C_n} = \frac{5n-2}{n+2}, \tag{7.4.19}$$

again using the recursion relation (7.3.2). In the limit of $n \to \infty$

$$d_{2.\infty} \equiv \lim_{n\to\infty} d_{2,n} = 5. \tag{7.4.20}$$

7.4.3 A General Equation for the Depth Function

We begin by making a recursion relation in a similar manner to the previous examples. In eq. (7.4.11) we noted that there was one diagram in the decomposition where the 2nd leg misses the left hand primary sub-tree. For general m there will be $m-1$ such cases. When the left sub-tree has $k = 0$ to $m-2$ vertices the contribution to the depth is from the leg $m-k-1$ of the right sub-tree because the left sub-tree has $k+1$ legs. The degeneracy of this term is given by the number of trees in the left block C_k. When $k > m-2$ the contribution is from the mth leg of the left block as before. Hence, in full the recursion relation can be written as

$$D_{m,n} = C_n + \sum_{k=m-1}^{n-1} D_{m,k}C_{n-1-k} + \sum_{k=0}^{m-2} D_{m-k-1,n-k-1}C_k. \tag{7.4.21}$$

The depth function is left-right symmetric in that the depth of the mth leg from the left is the same as the depth of the mth leg from the right (proved in appendix A.3), i.e. $D_{m,n} = D_{n+2-m,n}$. This is perhaps obvious when looking at the decomposition, but will be very useful in deriving a closed formula for the depth function.

Just as we saw for $m = 2$, the generating function for general m needs to start at $m-1$

$$\mathcal{D}_m(x) = \sum_{n=m-1}^{\infty} D_{m,n}x^n = \sum_{n=0}^{\infty} D_{m,n+m-1}x^{n+m-1}, \tag{7.4.22}$$

where in the final part the index on the sum starts at zero ($n \to n+m-1$). We start with eq. (7.4.21) for $D_{m,n+m}$

$$D_{m,n+m} = C_{n+m} + \sum_{k=m-1}^{n+m-1} D_{m,k}C_{n+m-k-1} + \sum_{k=0}^{m-2} D_{m-k-1,n+m-k-1}C_k. \tag{7.4.23}$$

Changing the index on the first sum so that it sums from zero ($k \to k+m-1$) and introducing a step function ($H_p = 1$ if $p \geq 0$, 0 if $p < 0$) so that the upper limit can go to infinity we obtain

$$D_{m,n+m} = C_{n+m} + \sum_{k=0}^{\infty} D_{m,k+m-1}C_{n-k}H_{n-k} + \sum_{k=0}^{m-2} D_{m-k-1,n+m-k-1}C_k. \tag{7.4.24}$$

We multiply both sides by x^{m+n} and sum from $n=0$ to ∞ to match the generating function (7.4.22)

$$\begin{aligned}\sum_{n=0}^{\infty} D_{m,n+m} x^{m+n} = \sum_{n=0}^{\infty} C_{n+m} x^{n+m} + \sum_{n=0}^{\infty}\sum_{k=0}^{\infty} D_{m,k+m-1} C_{n-k} H_{n-k} x^{n+m} \\ + \sum_{n=0}^{\infty}\sum_{k=0}^{m-2} D_{m-k-1,n+m-k-1} C_k x^{n+m}. \qquad (7.4.25)\end{aligned}$$

The left hand side can clearly be expressed in terms of $\mathcal{D}_m(x)$

$$\begin{aligned}\sum_{n=0}^{\infty} D_{m,n+m} x^{m+n} &= \sum_{n=1}^{\infty} D_{m,n+m-1} x^{n+m-1} \\ &= \sum_{n=0}^{\infty} D_{m,n+m-1} x^{n+m-1} - D_{m,m-1} x^{m-1} \\ &= \mathcal{D}_m(x) - D_{m,m-1} x^{m-1}. \qquad (7.4.26)\end{aligned}$$

The first term on the right can be written in terms of $\mathcal{C}(x)$ as

$$\begin{aligned}\sum_{n=0}^{\infty} C_{n+m} x^{n+m} &= \sum_{n=m}^{\infty} C_n x^n \\ &= \sum_{n=0}^{\infty} C_n x^n - \sum_{n=0}^{m-1} C_n x^n \\ &= \mathcal{C}(x) - \sum_{n=0}^{m-1} C_n x^n. \qquad (7.4.27)\end{aligned}$$

The second term on the right can also be written in terms of generating functions

$$\begin{aligned}\sum_{n=0}^{\infty}\sum_{k=0}^{\infty} D_{m,k+m-1} C_{n-k} H_{n-k} x^{n+m} &= \sum_{k=0}^{\infty} D_{m,k+m-1} x^{m+k} \sum_{n=0} C_{n-k} H_{n-k} x^{n-k} \\ &= x \sum_{k=0}^{\infty} D_{m,k+m-1} x^{k+m-1} \sum_{l=-k}^{\infty} C_l H_l x^l \\ &= x \mathcal{D}_m(x) \mathcal{C}(x), \qquad (7.4.28)\end{aligned}$$

where we have used $l = n - k$ on the second line. Similarly, the final term is

$$\begin{aligned}\sum_{n=0}^{\infty}\sum_{k=0}^{m-2} D_{m-k-1,n+m-k-1}C_k x^{n+m} &= \sum_{k=0}^{m-2} C_k x^{k+1} \sum_{n=0}^{\infty} D_{m-k-1,n+m-k-1} x^{n+m-k-1} \\ &= \sum_{k=0}^{m-2} C_k x^{k+1} \left[\sum_{n=0}^{\infty} D_{m-k-1,n+m-k-2} x^{n+m-k-2}\right. \\ &\qquad \left. - D_{m-k-1,m-k-2} x^{m-k-2}\right] \\ &= \sum_{k=0}^{m-2} C_k x^{k+1} \left[\mathcal{D}_{m-k-1}(x) - D_{m-k-1,m-k-2} x^{m-k-2}\right]. \end{aligned} \tag{7.4.29}$$

When put together eq. (7.4.25) becomes

$$\begin{aligned}\mathcal{D}_m(x) - D_{m,m-1}x^{m-1} = \mathcal{C}(x) &- \sum_{n=0}^{m-1} C_n x^n + x\mathcal{D}_m(x)\mathcal{C}(x) \\ &+ \sum_{k=0}^{m-2} C_k x^{k+1} \left[\mathcal{D}_{m-k-1}(x) - D_{m-k-1,m-k-2} x^{m-k-2}\right]. \end{aligned} \tag{7.4.30}$$

Collecting the $\mathcal{D}_m(x)$ on the left then using eq. (A.2.1) and $D_{m,n} = D_{n+2-m,n}$ yields

$$\begin{aligned}\mathcal{D}_m(x) = \mathcal{C}^2(x) - \mathcal{C}(x) &\sum_{n=0}^{m-1} C_n x^n + \mathcal{C}(x) D_{1,m-1} x^{m-1} \\ &+ \mathcal{C}(x) \sum_{k=0}^{m-2} C_k x^{k+1} \left[\mathcal{D}_{m-k-1}(x) - D_{1,m-k-2} x^{m-k-2}\right]. \end{aligned} \tag{7.4.31}$$

This means that we only need to know the depth of the first leg to create the generating functions of the rest. We know from section 7.4.1 that $D_{1,n} = C_{n+1} - C_n$ so

$$\begin{aligned}\mathcal{D}_m(x) = \mathcal{C}^2(x) - \mathcal{C}(x) &\sum_{n=0}^{m-1} C_n x^n + \mathcal{C}(x)(C_m - C_{m-1}) x^{m-1} \\ &+ \mathcal{C}(x) \sum_{k=0}^{m-2} C_k x^{k+1} \left[\mathcal{D}_{m-k-1}(x) - (C_{m-k-1} - C_{m-k-2}) x^{m-k-2}\right]. \end{aligned} \tag{7.4.32}$$

Using Segner's relation (7.2.1) we notice that for $m \geq 2$

$$\sum_{k=0}^{m-2} C_k C_{m-k-2} = C_{m-1} \tag{7.4.33}$$

and

$$\sum_{k=0}^{m-2} C_k C_{m-k-1} = \sum_{k=0}^{m-1} C_k C_{m-k-1} - C_{m-1}C_0 = C_m - C_{m-1}. \tag{7.4.34}$$

This enables us to simplify eq. (7.4.32) to

$$\begin{aligned}
\mathcal{D}_m(x) &= \mathcal{C}^2(x) - \mathcal{C}(x) \sum_{n=0}^{m-1} C_n x^n + \mathcal{C}(x) C_{m-1} x^{m-1} + \mathcal{C}(x) \sum_{k=0}^{m-2} C_k x^{k+1} \mathcal{D}_{m-k-1}(x) \\
&= \mathcal{C}^2(x) - \mathcal{C}(x) \sum_{n=0}^{m-2} C_n x^n - \mathcal{C}(x) C_{m-1} x^{m-1} \\
&\qquad + \mathcal{C}(x) C_{m-1} x^{m-1} + \mathcal{C}(x) \sum_{k=0}^{m-2} C_k x^{k+1} \mathcal{D}_{m-k-1}(x) \\
&= \mathcal{C}^2(x) + \mathcal{C}(x) \sum_{n=0}^{m-2} C_n x^n \left[x \mathcal{D}_{m-n-1}(x) - 1 \right]. \qquad (7.4.35)
\end{aligned}$$

Here we introduce the ansatz that will take us to the solution. We will show that the generating function $\mathcal{D}_m(x)$ decomposes as

$$\mathcal{D}_m(x) = f_m(x) \mathcal{C}(x) + g_m(x), \tag{7.4.36}$$

where $f_m(x)$ and $g_m(x)$ are polynomial in x of order $m-2$. This is motivated by the fact that for $m \geq 2$, $D_{m,n}$ is composed of a sum of Catalan numbers with indices from $n+1$ down to $n-(m-2)$. The Catalan numbers come out of the polynomial due to the fact that

$$x^m \mathcal{C}(x) = \sum_{n=0}^{\infty} C_n x^{n+m} = \sum_{n=m}^{\infty} C_{n-m} x^n. \tag{7.4.37}$$

The form of $f_m(x)\mathcal{C}(x)$ will give the expression for $d_{m,n}$ but when putting it in the form of (7.4.22) there will be a slew of terms that will be cancelled by $g_m(x)$.

Inserting the ansatz (7.4.36) into eq. (7.4.35) gives

$$f_m(x)\mathcal{C}(x) + g_m(x) = \mathcal{C}^2(x) + \mathcal{C}(x) \sum_{n=0}^{m-2} C_n x^n \left[x f_{m-n-1}(x)\mathcal{C}(x) + x g_{m-n-1}(x) - 1 \right]. \tag{7.4.38}$$

In the spirit of the ansatz we use eq. (7.3.5) to remove the $\mathcal{C}^2(x)$ terms and make the expression dependent on just $\mathcal{C}(x)$

$$\begin{aligned} f_m(x)\mathcal{C}(x) + g_m(x) = \frac{\mathcal{C}(x)}{x} - \frac{1}{x} + \mathcal{C}(x) \sum_{n=0}^{m-2} C_n x^n \left[x g_{m-n-1}(x) - 1 \right] \\ + \left[\mathcal{C}(x) - 1\right] \sum_{n=0}^{m-2} C_n x^n f_{m-n-1}(x)\mathcal{C}(x). \end{aligned} \tag{7.4.39}$$

Collecting coefficients of $\mathcal{C}(x)$ gives

$$f_m(x) = \frac{1}{x} + \sum_{n=0}^{m-2} C_n x^n \left[f_{m-n-1}(x) + x g_{m-n-1}(x) - 1 \right], \tag{7.4.40}$$

leaving

$$g_m(x) = -\frac{1}{x} - \sum_{n=0}^{m-2} C_n x^n f_{m-n-1}(x). \tag{7.4.41}$$

We can now write out $f_m(x)$ and $g_m(x)$ for the first few m. Recall that the above expressions are only valid for $m \geq 2$, we therefore obtain $f_1(x)$ and $g_1(x)$ from (7.4.6) to seed the other terms. The first seven $f_m(x)$ are

$$f_1(x) = \frac{1}{x} - 1 \tag{7.4.42}$$

$$f_2(x) = \frac{2}{x} - 3 \tag{7.4.43}$$

$$f_3(x) = \frac{3}{x} - 5 - 2x \tag{7.4.44}$$

$$f_4(x) = \frac{4}{x} - 7 - 4x - 4x^2 \tag{7.4.45}$$

$$f_5(x) = \frac{5}{x} - 9 - 6x - 8x^2 - 10x^3 \tag{7.4.46}$$

$$f_6(x) = \frac{6}{x} - 11 - 8x - 12x^2 - 20x^3 - 28x^4 \tag{7.4.47}$$

$$f_7(x) = \frac{7}{x} - 13 - 10x - 16x^2 - 30x^3 - 56x^4 - 84x^5 \tag{7.4.48}$$

and $g_m(x)$

$$g_1(x) = -\frac{1}{x} \tag{7.4.49}$$

$$g_2(x) = -\frac{2}{x} + 1 \tag{7.4.50}$$

$$g_3(x) = -\frac{3}{x} + 2 + x \tag{7.4.51}$$

$$g_4(x) = -\frac{4}{x} + 3 + 3x + 2x^2 \tag{7.4.52}$$

$$g_5(x) = -\frac{5}{x} + 4 + 5x + 7x^2 + 5x^3 \tag{7.4.53}$$

$$g_6(x) = -\frac{6}{x} + 5 + 7x + 12x^2 + 19x^3 + 14x^4 \tag{7.4.54}$$

$$g_7(x) = -\frac{7}{x} + 6 + 9x + 17x^2 + 33x^3 + 56x^4 + 42x^5. \tag{7.4.55}$$

It is quite simple to spot patterns in these equations and write a general formula for each

$$\begin{aligned} f_m(x) &= \frac{m}{x} - 1 - 2H_{m-2}C_0(m-1) - 2H_{m-3}C_1(m-2)x - 2H_{m-4}C_2(m-3)x^2 \\ &\quad - 2H_{m-5}C_3(m-4)x^3 - \ldots \\ &= \frac{m}{x} - 1 - 2\sum_{k=0}^{m-2}(m-k-1)C_k x^k \end{aligned} \tag{7.4.56}$$

$$\begin{aligned} g_m(x) &= -\frac{m}{x} + H_{m-2}(C_1(m-2) + C_0) + H_{m-3}(C_2(m-3) + C_1)x \\ &\quad + H_{m-4}(C_3(m-4) + C_2)x^2 + H_{m-5}(C_4(m-5) + C_3)x^3 + \ldots \\ &= -\frac{m}{x} + \sum_{k=0}^{m-2}\left[(m-k-2)C_{k+1} + C_k\right]x^k. \end{aligned} \tag{7.4.57}$$

In the first lines of each a step function H_p has been introduced to cut the expression so that it gives only the terms allowed for that m. This is neatly included within the summations in the final forms. Multiplying (7.4.56) by $\mathcal{C}(x)$ using (7.4.37) yields

$$\begin{aligned} f_m(x)\mathcal{C}(x) &= \sum_{n=0}^{\infty}\left[\frac{m}{x} - 1 - 2\sum_{k=0}^{m-2}(m-k-1)C_k x^k\right] C_n x^n \\ &= \sum_{n=m-1}^{\infty}\left[mC_{n+1} - C_n - 2\sum_{k=0}^{m-2}(m-k-1)C_kC_{n-k}\right] x^n \\ &\quad + m\sum_{n=-1}^{m-2} C_{n+1}x^n - \sum_{n=0}^{m-2} C_n x^n - 2\sum_{k=0}^{m-2}\sum_{n=k}^{m-2}(m-k-1)C_kC_{n-k}x^n. \end{aligned} \tag{7.4.58}$$

This implies that

$$D_{m,n} = mC_{n+1} - C_n - 2\sum_{k=0}^{m-2}(m-k-1)C_kC_{n-k} \tag{7.4.59}$$

and we have our result. We now just have to show that the rest of the terms are taken care of by $g(x)$. The final term in eq. (7.4.58) can be simplified using the double sum identity

$$\sum_{a=0}^{c}\sum_{b=a}^{c} M_{a,b} = \sum_{b=0}^{c}\sum_{a=0}^{b} M_{a,b} \tag{7.4.60}$$

where $a, b, c \in \mathbb{Z}_+$, which is found from changing the order of summation of a and b to c with the condition that $a \leq b$.

$$\begin{aligned} \sum_{k=0}^{m-2}\sum_{n=k}^{m-2}(m-k-1)C_kC_{n-k}x^n &= \sum_{n=0}^{m-2}\sum_{k=0}^{n}(m-k-1)C_kC_{n-k}x^n \\ &= \sum_{n=0}^{m-2}\sum_{k=0}^{n}\left[mC_kC_{n-k} - (k+1)C_kC_{n-k}\right] x^n \\ &= \sum_{n=0}^{m-2}\left[mC_{n+1} - (2n+1)C_n\right] x^n, \end{aligned} \tag{7.4.61}$$

where in the final line we have used Segner's relation on the first term and (A.2.7) on the second. Thus (7.4.58) can be written as

$$f_m(x)C(x) = \sum_{n=m-1}^{\infty} D_{m,n}x^n + \frac{m}{x} - \sum_{n=0}^{m-2}\left[C_n - mC_{n+1} + 2mC_{n+1} - 2(2n+1)C_n\right] x^n. \tag{7.4.62}$$

Using the recursion relation for Catalan numbers (7.3.2)

$$\begin{aligned}
f_m(x)C(x) &= \sum_{n=m-1}^{\infty} D_{m,n}x^n + \frac{m}{x} - \sum_{n=0}^{m-2} \left[C_n + mC_{n+1} - (n+2)C_{n+1}\right] x^n \\
&= \sum_{n=m-1}^{\infty} D_{m,n}x^n + \frac{m}{x} - \sum_{n=0}^{m-2} \left[(m-n-2)C_{n+1} + C_n\right] x^n \\
&= \mathcal{D}_m(x) - g_m(x), \qquad (7.4.63)
\end{aligned}$$

which is simply the initial ansatz (7.4.36).

7.5 Path Lengths in Catalan Trees

Now that the leaf depths have been calculated it is possible to begin the analysis of the average leaf-to-leaf path distance $A_{r,n}$. In a similar way to the complete trees of chapter 6 and leaf depths in the previous section, we shall first calculate the *summed leaf-to-leaf path length* $S_{r,n}$, which is defined as the total number of vertices that connect two leaves of separation r when summing over all possible trees with n vertices. For a tree with n vertices there are $n+1$ leaves and C_n trees, therefore for a separation r there are $(n+1-r)C_n$ paths, hence

$$A_{r,n} = \frac{S_{r,n}}{(n+1-r)C_n}. \qquad (7.5.1)$$

7.5.1 Nearest Neighbours ($r = 1$)

To derive the basic recursion relation, consider splitting the set of n vertex trees into sub-trees with $n-\alpha-1$ and α vertices as shown in fig. 7.6. For the set of α vertexed trees, the summed path length is simply $S_{1,\alpha}$. Similarly for the set of $n-\alpha-1$ vertexed trees, the summed path length is $S_{1,n-\alpha-1}$. The path connecting the two trees, i.e. the $(n-\alpha)^{\text{th}}$ leaf of the $n-\alpha-1$ vertex tree connecting to the 1^{st} leaf of the α vertex tree, requires more discussion. The summed root-to-vertex depth of the first leaf of the α vertex tree is $D_{1,\alpha}$. We next consider just a single path, with depth δ_1, within the set of paths that connect leaf $n-\alpha$ to the root in the $n-\alpha-1$ vertex tree as shown by the dashed line of fig. 7.6. As there are C_α possible trees with α vertices, δ_1 will contribute C_α times in the sum of all connecting paths. Similarly the root will contribute $1 \times C_\alpha$. The total length for paths connecting to the first leg of the α tree containing δ_1 is $D_{1,\alpha} + C_\alpha + C_\alpha\delta_1$. To obtain the complete summed path length we now sum over all $C_{n-\alpha-1}$ paths that connect leaf $n-\alpha$ to

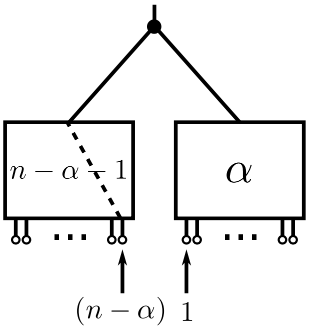

Figure 7.6: Schematic of splitting an n vertex tree into a single vertex, an $n - \alpha - 1$ vertex tree and an α vertex tree. A possible path from leaf $n - \alpha$ on the $n - \alpha - 1$ vertex tree to the 1st leaf on the α tree is indicated as dashed line.

the root in the $n - \alpha - 1$ tree $(\delta_1, \delta_2, \ldots, \delta_{C_{n-\alpha-1}})$

$$[D_{1,\alpha} + C_\alpha + C_\alpha \delta_1] + [D_{1,\alpha} + C_\alpha + C_\alpha \delta_2] + \cdots + [D_{1,\alpha} + C_\alpha + C_\alpha \delta_{n-\alpha-1}] \, . \quad (7.5.2)$$

As

$$\sum_{k=0}^{C_{n-\alpha-1}} \delta_k = D_{n-\alpha,n-\alpha-1}, \quad (7.5.3)$$

the summed path length becomes

$$C_\alpha D_{n-\alpha,n-\alpha-1} + C_{n-\alpha-1} D_{1,\alpha} + C_{n-\alpha-1} C_\alpha. \quad (7.5.4)$$

The inclusion of contributions from all α yields the equation

$$\begin{aligned} S_{1,n} = \sum_{\alpha=0}^{n-1} [&C_\alpha D_{n-\alpha,n-\alpha-1} + C_{n-\alpha-1} D_{1,\alpha} + C_{n-\alpha-1} C_\alpha \\ &+ C_\alpha S_{1,n-\alpha-1} + C_{n-\alpha-1} S_{1,\alpha}] \, . \end{aligned} \quad (7.5.5)$$

It is possible to simplify eq. (7.5.5) by utilising Segner's relation (7.2.1), the fact that the tree sets and therefore subsets are left-right symmetric ($D_{m,n} = D_{n+2-m,n}$) and that the order in which a sum is performed is irrelevant (discussed in appendix A.1) resulting in

$$\begin{aligned} S_{1,n} &= C_n + \sum_{\alpha=0}^{n-1} \left(C_\alpha D_{n-\alpha,n-\alpha-1} + C_{n-\alpha-1} D_{1,\alpha} + C_\alpha S_{1,n-\alpha-1} + C_{n-\alpha-1} S_{1,\alpha} \right) \\ &= C_n + \sum_{\alpha=0}^{n-1} \left(C_\alpha D_{1,n-\alpha-1} + C_{n-\alpha-1} D_{1,\alpha} + C_\alpha S_{1,n-\alpha-1} + C_{n-\alpha-1} S_{1,\alpha} \right) \\ &= C_n + 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1} D_{1,\alpha} + 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1} S_{1,\alpha}. \end{aligned} \tag{7.5.6}$$

Using the first leaf depth function (7.4.8) eq. (7.5.5) can be written as a recursion of just path lengths and Catalan numbers

$$\begin{aligned} S_{1,n} &= C_n + 2\sum_{\alpha=0}^{n-1} \left(C_{\alpha+1} - C_\alpha \right) C_{n-\alpha-1} + 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1} S_{1,\alpha} \\ &= C_n + 2\sum_{\alpha=0}^{n-1} C_{\alpha+1} C_{n-(\alpha+1)} - 2\sum_{\alpha=0}^{n-1} C_\alpha C_{n-\alpha-1} + 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1} S_{1,\alpha} \\ &= C_n + 2\sum_{\beta=0}^{n} C_\beta C_{n-\beta} - 2C_n - 2C_n + 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1} S_{1,\alpha} \\ &= 2C_{n+1} - 3C_n + 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1} S_{1,\alpha}, \end{aligned} \tag{7.5.7}$$

where in the third line we have changed the index of the sum to $\beta = \alpha + 1$ and subtracted $2C_0 C_{n-0}$ allowing the lower limit of the sum to be zero. On the final line we use Segner's relation (7.2.1) for $n + 1$. From this recursion relation it is now possible to use a generating function approach to find a result. The appropriate generating function is defined as

$$\mathcal{S}_1(x) = \sum_{n=0}^{\infty} S_{1,n} x^n = \sum_{n=0}^{\infty} S_{1,n+1} x^{n+1} \tag{7.5.8}$$

as $S_{1,0} = 0$. To match the generating function multiply $S_{1,n+1}$ by x^{n+1} and sum over n to obtain

$$\begin{aligned}\sum_{n=0}^{\infty} S_{1,n+1}x^{n+1} &= 2\sum_{n=0}^{\infty} C_{n+2}x^{n+1} - 3\sum_{n=0}^{\infty} C_{n+1}x^{n+1} + 2\sum_{n=0}^{\infty}\sum_{\alpha=0}^{n} C_{n-\alpha}S_{1,\alpha}x^{n+1} \\ &= \frac{2}{x}\left(\sum_{n=0}^{\infty} C_n x^n - C_0 - C_1 x\right) - 3\left(\sum_{n=0}^{\infty} C_n x^n - C_0\right) \\ &\qquad + 2x\sum_{n=0}^{\infty}\sum_{\alpha=0}^{\infty} H_{n-\alpha}C_{n-\alpha}S_{1,\alpha}x^{n-\alpha}x^{\alpha}, \qquad (7.5.9)\end{aligned}$$

where the first two terms on the right have been rearranged to match the generating function for the Catalan numbers (7.3.4). In the final term, similar to in eq. (7.4.24) in the depth function derivation, we introduce a step function $H_{n-\alpha}$ to allow the α index to go to infinity. Changing the n index on the final term to $q = n - \alpha$ allows the whole equation to be expressed in terms of generating functions

$$\begin{aligned}\mathcal{S}_1(x) &= \frac{2}{x}\left[\mathcal{C}(x) - 1\right] - 2 - 3\mathcal{C}(x) + 3 + 2x\sum_{\alpha=0}^{\infty}\sum_{q=-\alpha}^{\infty} H_q C_q x^q S_{1,\alpha}x^{\alpha} \\ &= \frac{2}{x}\left[\mathcal{C}(x) - 1\right] + 1 - 3\mathcal{C}(x) + 2x\sum_{\alpha=0}^{\infty}\sum_{q=0}^{\infty} C_q x^q S_{1,\alpha}x^{\alpha} \\ &= \frac{2}{x}\left[\mathcal{C}(x) - 1\right] + 1 - 3\mathcal{C}(x) + 2x\mathcal{C}(x)\mathcal{S}_1(x). \qquad (7.5.10)\end{aligned}$$

Solving for $\mathcal{S}_1(n)$ gives

$$\begin{aligned}\mathcal{S}_1(x) &= \frac{\frac{2}{x}\left[\mathcal{C}(x) - 1\right] + 1 - 3\mathcal{C}(x)}{1 - 2x\mathcal{C}(x)} \\ &= \frac{2\mathcal{C}(x)^2 - 3\mathcal{C}(x) + 1}{\sqrt{1-4x}} \\ &= \sum_{n=0}^{\infty}\left[2(n+1)C_{n+1} - 3(2n+1)C_n + (n+1)C_n\right]x^n, \qquad (7.5.11)\end{aligned}$$

where on the second line we have used $\mathcal{C}^2(x) = \frac{\mathcal{C}(x)-1}{x}$ from eq. (7.3.5). The final line is obtained using $\frac{1}{\sqrt{1-4x}} = \sum_{\alpha=0}^{\infty}(\alpha+1)C_\alpha x^\alpha$, $\frac{\mathcal{C}(x)}{\sqrt{1-4x}} = \sum_{n=0}^{\infty}(2n+1)C_n x^n$ and $\frac{\mathcal{C}^2(x)}{\sqrt{1-4x}} = \sum_{n=0}^{\infty}(n+1)C_{n+1}x^n$ derived in appendix A.2.

Equation (7.5.11) can be readily compared to the generating function (7.5.8) to get the result

$$S_{1,n} = 2(n+1)C_{n+1} - (5n+2)C_n. \qquad (7.5.12)$$

This can be simplified using the recursion rule for Catalan numbers (7.3.2)

$$\begin{aligned} S_{1,n} &= \frac{2(n+1)\left[(4n+2)C_n\right]}{n+2} - (5n+2)C_n \\ &= \frac{3n^2C_n}{n+2}. \end{aligned} \tag{7.5.13}$$

The total number of paths is nC_n so the average leaf-to-leaf path length ($A_{1,n}$) becomes

$$A_{1,n} = \frac{3n}{n+2}. \tag{7.5.14}$$

In the infinite n limit this is

$$A_{1,\infty} \equiv \lim_{n\to\infty} A_{1,n} = 3. \tag{7.5.15}$$

7.5.2 General Path Lengths

As in section 7.5.1, the discussion begins with the set of trees decomposed into an α and $n-\alpha-1$ vertex tree. For this discussion we will split the full expression into terms coming from paths within each of the sub-trees (the recursion term) and those paths that connect them.

The recursion term $[S_{r,n}]_{\mathrm{rec}}$ is simply the contribution to the summed path length from the two sub-trees $S_{r,n-\alpha-1}$ and $S_{r,\alpha}$ with appropriate degeneracies as before

$$\begin{aligned} [S_{r,n}]_{\mathrm{rec}} &= \sum_{\alpha=0}^{n-1}\left[C_\alpha S_{r,n-\alpha-1} + C_{n-\alpha-1}S_{r,\alpha}\right] \\ &= \sum_{\alpha=0}^{n-1}\left[C_{n-1-\alpha}S_{r,n-(n-1-\alpha)-1} + C_{n-\alpha-1}S_{r,\alpha}\right] \\ &= 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1}S_{r,\alpha}, \end{aligned} \tag{7.5.16}$$

where in the second line we are effectively summing from $n-1$ down to 0 in the first term.

The connecting paths are somewhat more complicated; in the case of nearest neighbours there is only one path that connects the two subtrees; for general paths there are r connecting paths. Figure 7.7(a) shows the three connecting paths for $r=3$. We observe that this is only true for certain values of α, near the edges of the tree the boundary prevents the existence of all r connecting paths. Figure 7.7(b) gives an example for $r=3$ again. We shall thus refer to the terms away from the

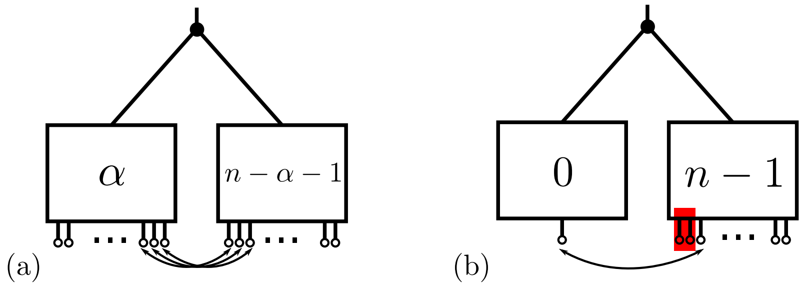

Figure 7.7: The connecting paths for $r = 3$ (a) away from the edges of the tree and (b) at the edge of the tree. The legs highlighted in red are those which have no connecting path due to the tree edge.

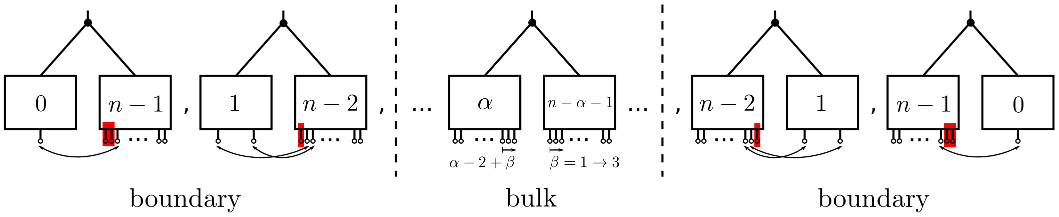

Figure 7.8: Diagrammatic decomposition showing the *bulk* and *boundary* contributions to the leaf-to-leaf path length. Again, the legs highlighted in red are those which have no connecting path due to the tree edge.

edges as the *bulk* and the terms affected by the edge as the *boundary*, both of which shall be treated separately.

The bulk terms are the simplest ones to create an expression for; they are just the generalisation of the nearest neighbour case. As before, the length of each connecting path is the addition of the depth of the leg in the left hand sub-tree, the root node and the depth of the leg in the right sub-tree. Here, we are again looking for the summed path lengths so need to use the depth functions $D_{m,n}$ for the depth of the legs in the left and right sub-trees, each has a degeneracy given by the number of trees in the opposing sub-tree and the root contributes 1 for each combination. We then need to sum over the r different positions that the connecting paths can start and end on labelled by β and also sum over the sub-tree combinations labelled by α, noting that the limits on the α sum are given by when this *bulk* behaviour is satisfied.

As a simple example take $r = 3$ again, shown schematically in fig. 7.8. The β index runs over the three paths connecting leg β in the right sub-tree and leg

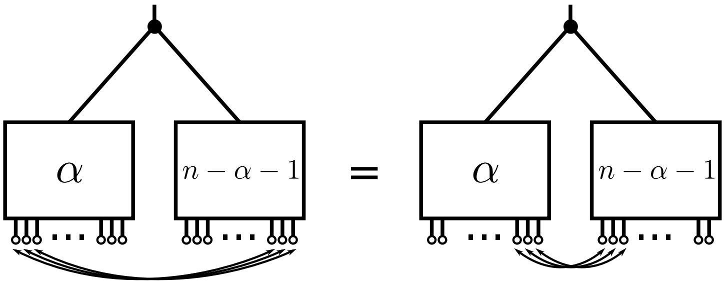

Figure 7.9: The connecting paths when $r = n - 2$. The long range paths are equivalent to short range paths due to the left-right symmetry of the sub-trees.

$\alpha - 2 + \beta$ in the left. The bulk behaviour is present for $\alpha = 2$ to $n - 3$ thus

$$[S_{3,n}]_{\mathrm{blk}} = \sum_{\alpha=2}^{n-3} \sum_{\beta=1}^{3} [C_{n-\alpha-1} D_{\alpha-2+\beta,\alpha} + C_\alpha D_{\beta,n-\alpha-1} + C_\alpha C_{n-\alpha-1}] \qquad (7.5.17)$$

This discussion generalises easily for $r \leq (n+1)/2$ after which there is no obvious bulk contribution as the lower limit on the α sum becomes larger than the upper limit.

$$\begin{aligned}
[S_{r,n}]_{\mathrm{blk}} &= \sum_{\alpha=r-1}^{n-r} \sum_{\beta=1}^{r} [C_{n-\alpha-1} D_{\alpha+1-r+\beta,\alpha} + C_\alpha D_{\beta,n-\alpha-1} + C_\alpha C_{n-\alpha-1}] \\
&= \sum_{\alpha=r-1}^{n-r} \sum_{\beta=1}^{r} [C_{n-\alpha-1} D_{1+r-\beta,\alpha} + C_\alpha D_{\beta,n-\alpha-1} + C_\alpha C_{n-\alpha-1}] \\
&= \sum_{\alpha=r-1}^{n-r} \sum_{\beta=1}^{r} \left[C_{n-\alpha-1} D_{\beta,\alpha} + C_{n-r+r-1-\alpha} D_{\beta,n-(n-r+r-1-\alpha)-1} + C_\alpha C_{n-\alpha-1}\right] \\
&= \sum_{\alpha=r-1}^{n-r} \sum_{\beta=1}^{r} [2C_{n-\alpha-1} D_{\beta,\alpha} + C_\alpha C_{n-\alpha-1}] \qquad ; \qquad r \leq \frac{n+1}{2} \qquad (7.5.18)
\end{aligned}$$

where we have used the left-right symmetry of the depth function ($D_{m,n} = D_{n+2-m,n}$) to get to line two. To get to line three we have summed the β index from the upper limit for the first term and summed the α index from the top on the second term.

The paths where $r > (n+1)/2$ have equivalent bulk terms due to the left-right symmetry of the sub-trees as shown in fig. 7.9. In this sense $r = n$ is the same as $r = 1$, $r = n - 1$ is equal to $r = 2$ and so on. The contribution to the bulk form these *long range* terms will have the same form as for $r \leq (n+1)/2$, but in the

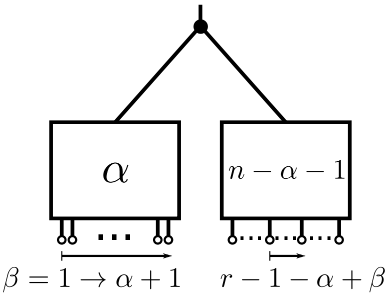

Figure 7.10: The legs that are involved in boundary paths when $\alpha < n - \alpha - 1$. All of the legs are involved in the left sub-tree, only the legs r away are involved in the right sub-tree labelled by $r-1-\alpha+\beta$.

limits on the sums r will be replaced by $n+1-r$

$$[S_{r,n}]_{\text{blk}} = \sum_{\alpha=n-r}^{r-1} \sum_{\beta=1}^{n-r+1} [2C_{n-\alpha-1}D_{\beta,\alpha} + C_\alpha C_{n-\alpha-1}] \quad ; \quad r > \frac{n+1}{2} \tag{7.5.19}$$

The boundary terms are limited by the fact that one of the subtrees has fewer than r legs, meaning that there cannot be all r connecting paths. In the boundary case for $r \leq (n+1)/2$ there are $\alpha+1$ contributions rather than r, so the β index is limited by $\alpha+1$. The limits on the α sum are given by the set of sub-trees with fewer than r legs and hence $r-1$ vertices. As shown in fig. 7.10 when $\alpha < n-\alpha-1$ all of the legs of the left sub-tree contribute but only those that are r legs away from a starting point do in the right sub-tree. This set starts with leg $r-\alpha-1$ and goes up with β. The $\alpha < n-\alpha-1$ refer to left hand boundary as shown in fig. 7.8. The right hand boundary is identical due to the left-right symmetry of sub-trees. The total expression is

$$\begin{aligned}
[S_{r,n}]_{\text{bnd}} &= 2\sum_{\alpha=0}^{r-2}\sum_{\beta=1}^{\alpha+1} [C_{n-\alpha-1}D_{\beta,\alpha} + C_\alpha D_{r-1-\alpha+\beta,n-\alpha-1} + C_\alpha C_{n-\alpha-1}] \\
&= 2\sum_{\alpha=0}^{r-2}\sum_{\beta=1}^{\alpha+1} [C_{n-\alpha-1}D_{\beta,\alpha} + C_\alpha D_{r-\beta+1,n-\alpha-1} + C_\alpha C_{n-\alpha-1}] \quad ; \; r \leq \frac{n+1}{2},
\end{aligned} \tag{7.5.20}$$

where to get to the final line we have summed the β index of the second term from top to bottom. As with the bulk terms the $r > (n+1)/2$ is identical except for replacing r with $n+1-r$

$$\begin{aligned}[S_{r,n}]_{\mathrm{bnd}} &= 2\sum_{\alpha=0}^{n-r-1}\sum_{\beta=1}^{\alpha+1}\left[C_{n-\alpha-1}D_{\beta,\alpha} + C_\alpha D_{n+2-r-\beta,n-\alpha-1} + C_\alpha C_{n-\alpha-1}\right] \\ &= 2\sum_{\alpha=0}^{n-r-1}\sum_{\beta=1}^{\alpha+1}\left[C_{n-\alpha-1}D_{\beta,\alpha} + C_\alpha D_{r-\beta+1,n-\alpha-1} + C_\alpha C_{n-\alpha-1}\right] \;\; ;\, r \leq \frac{n+1}{2},\end{aligned} \tag{7.5.21}$$

where we have again used $D_{m,n} = D_{n+2-m,n}$ and changed the order of the sum on the second term. When put together the full recursion relation is

$$[S_{r,n}]_{\mathrm{blk}} = \begin{cases} \displaystyle\sum_{\alpha=r-1}^{n-r}\sum_{\beta=0}^{r-1}\left(2C_{n-\alpha-1}D_{\beta+1,\alpha} + C_\alpha C_{n-\alpha-1}\right) & ; r \leq \dfrac{n+1}{2} \\ \displaystyle\sum_{\alpha=n-r}^{r-1}\sum_{\beta=0}^{n-r}\left(2C_{n-\alpha-1}D_{\beta+1,\alpha} + C_\alpha C_{n-\alpha-1}\right) & ; r > \dfrac{n+1}{2} \end{cases} \tag{7.5.22}$$

$$[S_{r,n}]_{\mathrm{bnd}} = \begin{cases} \displaystyle 2\sum_{\alpha=0}^{r-2}\sum_{\beta=0}^{\alpha}\left(C_{n-\alpha-1}D_{\beta+1,\alpha} + C_\alpha D_{r-\beta,n-\alpha-1} + C_\alpha C_{n-\alpha-1}\right) & ; r \leq \dfrac{n+1}{2} \\ \displaystyle 2\sum_{\alpha=0}^{n-r-1}\sum_{\beta=0}^{\alpha}\left(C_{n-\alpha-1}D_{\beta+1,\alpha} + C_\alpha D_{r-\beta,n-\alpha-1} + C_\alpha C_{n-\alpha-1}\right) & ; r > \dfrac{n+1}{2} \end{cases} \tag{7.5.23}$$

$$[S_{r,n}]_{\mathrm{rec}} = 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1}S_{r,\alpha}. \tag{7.5.24}$$

Note that the limits on the sums imply that

$$S_{r,n} = 0 \qquad \forall r > n. \tag{7.5.25}$$

We combine the bulk and boundary terms to create the *source* ($T_{r,n}$), which seeds the recursion and sets the boundary properties. Thus the full equation becomes

$$S_{r,n} = T_{r,n} + 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1}S_{r,\alpha}. \tag{7.5.26}$$

7.5.3 Next-to-Nearest Neighbours

Using the full recursion relation it is simple to obtain an expression for *next-to-nearest neighbours* $(r = 2)$

$$S_{2,n} = 6C_{n+1} - 14C_n + 2C_{n-1} + 2\sum_{\alpha=0}^{n-1} C_{n-\alpha-1}S_{2,\alpha}. \tag{7.5.27}$$

As in section 7.5.1 we can use a generating function to obtain a closed form for the path lengths. Let

$$\mathcal{S}_2(x) = \sum_{n=0}^{\infty} S_{2,n}x^n = \sum_{n=0}^{\infty} S_{2,n+1}x^{n+1}, \tag{7.5.28}$$

where $S_{2,0} = 0$ due to (7.5.25). As before we start with $S_{2,n+1}$, multiply eq. (7.5.27) by x^{n+1} and sum to get

$$\sum_{n=0}^{\infty} S_{2,n+1}x^{n+1} = \sum_{n=0}^{\infty}(6C_{n+2} - 14C_{n+1} + 2C_n)x^{n+1} + 2\sum_{n=0}^{\infty}\sum_{\alpha=0}^{n} C_{n-\alpha}S_{2,\alpha}x^{n+1}. \tag{7.5.29}$$

Putting in a step function and summing to infinity as in eq. (7.5.10) we obtain

$$\begin{aligned}
\mathcal{S}_2(x) &= \frac{6}{x}\sum_{n=0}^{\infty} C_{n+2}x^{n+2} - 14\sum_{n=0}^{\infty} C_{n+1}x^{n+1} + 2x\sum_{n=0}^{\infty} C_nx^n \\
&\qquad + 2x\sum_{n=0}^{\infty}\sum_{\alpha=0}^{\infty} H_{n-\alpha}C_{n-\alpha}S_{2,\alpha}x^n \\
&= \frac{6}{x}\left(\mathcal{C}(x) - x - 1\right) - 14\left(\mathcal{C}(x) - 1\right) + 2x\mathcal{C}(x) + 2x\sum_{\alpha=0}^{\infty}\sum_{q=-\alpha}^{\infty} H_qC_qx^qS_{2,\alpha}x^{\alpha} \\
&= \frac{6}{x}\left(\mathcal{C}(x) - 1\right) + 8 - 14\mathcal{C}(x) + 2x\mathcal{C}(x) + 2x\sum_{\alpha=0}^{\infty}\sum_{q=0}^{\infty} C_qx^qS_{2,\alpha}x^{\alpha} \\
&= 6\mathcal{C}^2(x) + 8 - 14\mathcal{C}(x) + 2x\mathcal{C}(x) + 2x\mathcal{C}(x)\mathcal{S}_2(x)
\end{aligned} \tag{7.5.30}$$

where we have used $q = n - \alpha$ on the second line and $x\mathcal{C}^2(x) = \mathcal{C}(x) - 1$ (from eq. (7.3.5)) on the final line. Collecting the $\mathcal{S}_2(x)$ terms on the right and applying $1 - 2x\mathcal{C}(x) = \sqrt{1-4x}$ (from eq. (7.3.7)) gives

$$\mathcal{S}_2(x) = \frac{6\mathcal{C}^2(x) + 8 - 14\mathcal{C}(x) + 2x\mathcal{C}(x)}{\sqrt{1-4x}}. \tag{7.5.31}$$

Then using eqs. (A.2.16), (A.2.18) and (A.2.22), as in the $r = 1$ case, the final result can be obtained

$$\begin{aligned}\mathcal{S}_2(x) &= \sum_{n=0}^{\infty} \left[6(n+1)C_{n+1} - 14(2n+1)C_n + 8(n+1)C_n + 2x(2n+1)C_n\right] x^n \\ &= \sum_{n=0}^{\infty} \left[6(n+1)C_{n+1} - (20n+6)C_n\right] x^n + 2\sum_{n=1}^{\infty}(2n+1)C_{n-1}x^n \\ &= \sum_{n=1}^{\infty} \left[6(n+1)C_{n+1} - (20n+6)C_n + 2(2n+1)C_{n-1}\right] x^n + 6C_1 - 6C_0 \\ &= \sum_{n=1}^{\infty} \left[6(n+1)C_{n+1} - (20n+6)C_n + 2(2n+1)C_{n-1}\right] x^n. \qquad (7.5.32)\end{aligned}$$

This now gives an expression for $S_{2,n}$, which can be tidied up with (7.3.2)

$$\begin{aligned}S_{2,n} &= 6(n+1)C_{n+1} - (20n+6)C_n + 2(2n+1)C_{n-1} \\ &= \frac{(n-1)(5n-2)}{n+2}C_n. \qquad (7.5.33)\end{aligned}$$

To obtain the average path length we need to divide by the number of possible paths, which is $(n-1)C_n$ to get

$$A_{2,n} = \frac{5n-2}{n+2}. \qquad (7.5.34)$$

In the infinite n limit $A_{2,\infty} = 5$.

7.5.4 Larger Separations

Using the same method as above it is possible to get an expression for $r = 3$

$$S_{3,n} = \frac{(n-2)(13n^2 - 18n + 2)C_n}{(n+2)(2n-1)}. \qquad (7.5.35)$$

Dividing through by $(n-2)C_n$ gives the average:

$$A_{3,n} = \frac{13n^2 - 18n + 2}{(n+2)(2n-1)}. \qquad (7.5.36)$$

The method can be generalised for any r given an appropriate source function $\mathcal{T}_r(x)$. Starting with (7.5.26) and using the generating functions

$$\mathcal{S}_r(x) = \sum_{n=r-1}^{\infty} S_{r,n} x^n, \tag{7.5.37}$$

$$\mathcal{T}_r(x) = \sum_{n=r-1}^{\infty} T_{r,n} x^n, \tag{7.5.38}$$

we proceed as before,

$$\begin{aligned}
\sum_{n=r-1}^{\infty} S_{r,n} x^n &= \sum_{n=r-1}^{\infty} T_{r,n} x^n + 2 \sum_{n=r-1}^{\infty} \sum_{\alpha=0}^{n-1} C_{n-\alpha-1} S_{r,\alpha} x^n \\
\mathcal{S}_r(x) &= \sum_{n=r-1}^{\infty} T_{r,n} x^n + 2 \sum_{n=r-1}^{\infty} \sum_{\alpha=0}^{\infty} H_{n-\alpha-1} C_{n-\alpha-1} S_{r,\alpha} x^n \\
&= \sum_{r-1}^{\infty} T_{r,n} x^n + 2x \sum_{\alpha=0}^{\infty} S_{r,\alpha} x^\alpha \sum_{q=r-\alpha-2}^{\infty} H_q C_q x^q \\
&= \sum_{r-1}^{\infty} T_{r,n} x^n + 2x \sum_{\alpha=r-1}^{\infty} S_{r,\alpha} x^\alpha \sum_{q=0}^{\infty} C_q x^q \\
&= \sum_{r-1}^{\infty} T_{r,n} x^n + 2x \sum_{\alpha=r-1}^{\infty} S_{r,\alpha} x^\alpha \mathcal{C}(x) \\
&= \mathcal{T}_r(x) + 2x\mathcal{C}(x)\mathcal{S}_r(x),
\end{aligned} \tag{7.5.39}$$

where on the third line we have made use of the step function H_q and recalled that $S_{r,n} = 0$ when $r < n$ so the lower limit of the α sum can be raised to match the rest. Collecting the $\mathcal{S}_r(x)$ terms and dividing by $[1 - 2x\mathcal{C}(x)]$ again gives

$$\mathcal{S}_r(x) = \frac{\mathcal{T}_r(x)}{1 - 2x\mathcal{C}(x)}. \tag{7.5.40}$$

The issue now is finding a general form for the source, which appears to be as hard as finding a general equation for the path lengths. It is possible to use the generalised recursion relation to get a single case, but the number of terms grows rapidly with r and it soon becomes impractical to obtain these by hand. Using algebraic mathematics software such as *Mathematica* widens the scope of practicality, but a general expression has still not been found using this method.

7.5.5 General Solution

In additional work [14] it was shown that the depth function $D_{m,n}$ and the summed path length $S_{r,n}$ are very closely related

$$S_{r,n} = (n + 1 - r)D_{r,n}. \tag{7.5.41}$$

This in fact means that the average path length for a separation r is equal to the average depth when $m = r$, i.e.

$$A_{r,n} = \frac{(n + 1 - r)D_{r,n}}{(n + 1 - r)C_n} = d_{r,n}. \tag{7.5.42}$$

This can be observed for $r = 1, 2$ by comparing eqs. (7.4.9) and (7.4.19) with (7.5.14) and (7.5.34). Reference [14] shows that the solution to the depth function (7.4.59) can be expressed explicitly without the summation as

$$D_{m,n} = \frac{2m(m + 1)(2n - 2m + 1)(2n - 2m + 3)}{(n + 1)(n + 2)} C_m C_{n-m} - C_n. \tag{7.5.43}$$

Hence

$$A_{r,n} = d_{r,n} = \frac{2r(r + 1)(2n - 2r + 1)(2n - 2r + 3)}{(n + 1)(n + 2)} \frac{C_r C_{n-r}}{C_n} - 1. \tag{7.5.44}$$

Large n behaviour can be accessed by using the factorial form of the Catalan numbers (7.3.1) along with Stirling's approximation [172]

$$n! \sim \sqrt{2\pi n} \left(\frac{n}{e}\right)^n. \tag{7.5.45}$$

For large n the Catalan numbers are therefore

$$C_n \sim \frac{1}{n\sqrt{n\pi}} 2^{2n}. \tag{7.5.46}$$

Using this in eq. (7.5.44) gives

$$A_{r,\infty} \equiv \lim_{n\to\infty} = \frac{8r(r + 1)}{4^r} C_r - 1. \tag{7.5.47}$$

For large r as well $(0 \ll r \ll n)$ we get

$$A_{r,\infty} \sim \sqrt{\frac{64r}{\pi}}, \tag{7.5.48}$$

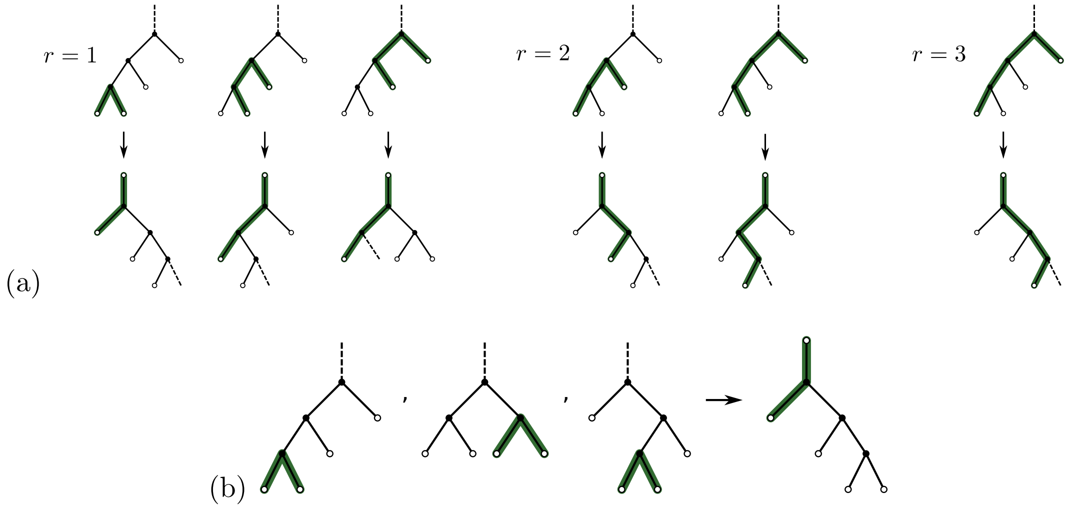

Figure 7.11: Graphical representation of the equivalence between leaf-to-leaf path length on a Catalan graph with $n = 3$ nodes and a rooted path connecting the same nodes created by deforming the tree. (a) All possible paths on a particular tree are shown, paths with separation r are mapped to depths in trees with $m = r$. The thick green line denotes the path. (b) Example of the degeneracy of rooted paths, showing the $(n - r + 1) = 3$ cases where $n = 3$ and $r = 1$ that map to the same rooted tree.

hence the asymptotic behaviour of the path lengths follow a square root in r.

There is a simple geometric interpretation of the result of (7.5.42). For example, take a leaf-to-leaf path and deform the tree such that the left hand leaf is above the root, making the first vertex the new root (a physical example is given in appendix A.4). Now it is clear that any path with separation r can be expressed as a leg depth with $m = r$, as shown in fig. 7.11(a). As the set of Catalan trees is *complete* in the sense that all possible binary trees are part of the set of Catalan trees, the newly deformed tree is also one of the set. By observation in, for example, fig. 7.11(b), one can see that $(n-r+1)$ paths map to each unique rooted tree depth. The degeneracy coming from the number of possible vertices to the right of the right hand leaf, which can be the original root, highlighted in fig. 7.11 by the dashed edge. The complete set of paths and their corresponding leaf depths are given in appendix A.4. Figure 7.12 plots $A_{r,n}$ against r for various values of n. The form of the plots are similar to those of tSDRG in chapter 5 in that they have a point beyond which the average path length gets shorter, which was not seen in the case of the complete trees of chapter 6. The asymptotic form (7.5.48) is however very different from that found for the complete trees exhibiting a square root form rather than logarithmic

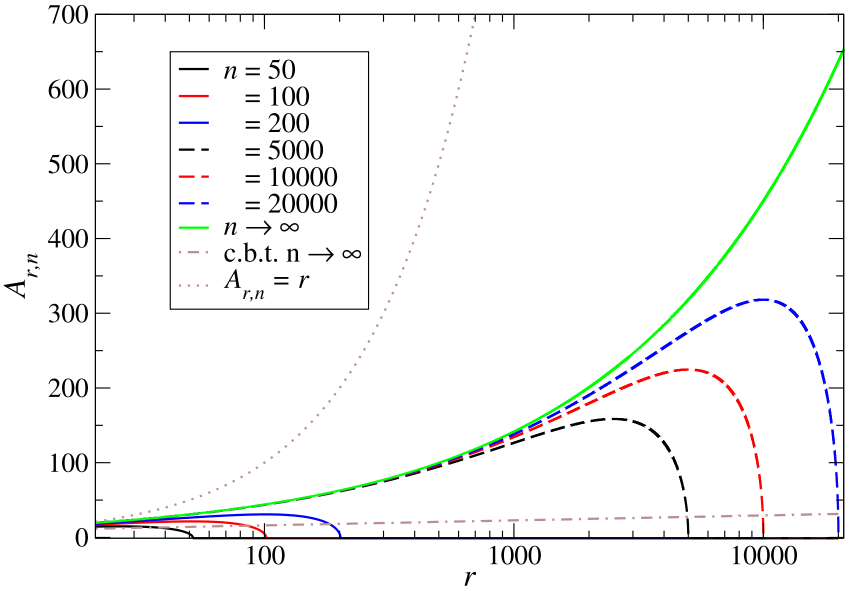

Figure 7.12: Average length of a leaf to leaf path $A_{r,n}$ versus separation r for $n = 50$ (solid black line), 100 (solid red), 200 (solid blue) and large $n = 5,000$ (dashed black line), $10,000$ (dashed red), $20,000$ (dashed blue). The solid green line denotes $A_{r,\infty}$, the dotted line corresponds to r while the dashed-dotted line shows the corresponding result of chapter 6 for complete binary trees (c.b.t.).

dependence on r. Furthermore the Catalan trees have a different path structure to those generated by tSDRG, which also have a logarithmic asymptotic scaling. This in turn means that when increasing r the average path length increases much more rapidly in the case of Catalan trees. This is shown in fig. 7.12 by comparing the green ($A_{r,\infty}$) line with the dashed-dotted line for complete binary trees. The reason for the much larger average path length is due to the fact that the complete binary is the most *compact* that a tree of n vertices can be. The Catalan trees have many more variants that are often more *extended*, even to the maximal case where a path can contain all n vertices.

7.6 Random Binary Trees

In this section we numerically calculate the average leaf-to-leaf distance $A^{\mathcal{R}}_{r,n}$ of all possible pairs of leaves of separation r in a *random* binary tree with n vertices. The random trees are constructed *from the ground up*, that is starting with $L = n + 1$ leaves and inserting vertices that join two neighbours until the root node. The

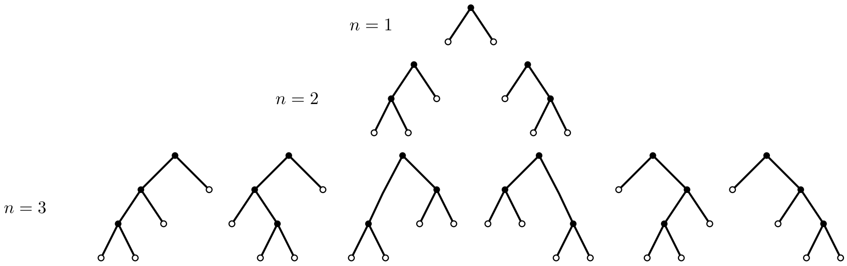

Figure 7.13: A complete set of random binary trees for n = 1,2 and 3 ($L = 2, 3, 4$). Circles and lines are as in fig. 7.1. Notice that unlike the Catalan trees in fig. 7.2 the centre two trees with $n = 3$ have the same structure and are therefore degenerate.

structure can then be defined by a vector that gives the order in which the vertices can be placed, in much the same way as the *order* vector described in chapter 5. The trees are *random* in the sense that the vector that defines their structure can be generated randomly from the $n!$ different permutations of the structure vector. This full set of trees is different to the Catalan trees in that there can be two vectors that create the same structure despite constructing the tree in a different order, as shown in fig. 7.13. This degeneracy changes the average path lengths in the trees.

For small n it is possible to generate all $n!$ permutations, shown in fig. 7.14(a), while for large n, fig. 7.14(b), we average over a finite number $N \ll n!$ chosen randomly. We see in fig. 7.14(a) that, similar to the complete binary trees considered in chapter 6, the path lengths increase approximately logarithmically with r until they reach a maximal value. Like the paths in the Catalan trees, the path length decreases rapidly beyond this maximal value. We also see that when we choose $10,000$ random binary trees from the $10! = 3,628,800$ possible trees at $n = 10$ the average leaf-to-leaf distance for each r is distinguishably different from the average over all possible permutations. This suggests that rare tree structures are quite important.

In fig. 7.14(b) we nevertheless show estimates of $A^{\mathcal{R}}_{r,n}$ for various n for 500 randomly chosen trees. As before, the shape of the curves for large n is similar to those for small n, with the asymptotic logarithmic scaling more apparent. For these larger values of n the path lengths are now longer than those of the complete tree, where they are shorter in fig. 7.14(a).

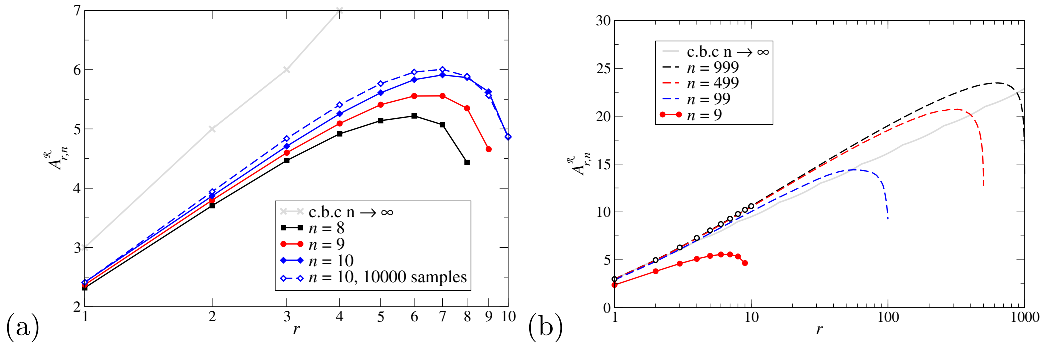

Figure 7.14: (a) Average leaf-to-leaf path length through a random binary tree connecting two leaves of separation r averaged over all possible trees for $n = 8, 9$ and 10 (solid symbols, lines are guide to the eye only). The open symbols (dashed line guide to the eye) refer to an average over 10000 randomly chosen trees from the 10! possibilities for $n = 10$. The grey crosses ($\times$) and line correspond to $\mathcal{L}^{(2)}_{\infty}(r)$ from the complete binary trees in chapter 6. (b) Average leaf-to-leaf distance constructed from 500 randomly chosen binary trees with $n = 99$, 499 and 999 (dashed lines). The open symbols ($\circ$) denote the first 10 data points. The closed symbols (red $\bullet$) and the solid line correspond to the $L = 10$ data from (a). The grey line corresponds to $\mathcal{L}^{(2)}_{\infty}(r)$ as in (a). Error bars have been omitted in (a) and (b) as they are within symbol size.

7.7 Conclusion

In this chapter we continue the analysis of the structure of TTNs using graph theoretic techniques. We find an analytic expression for the average path length between two leaves of a Catalan tree when averaged over all possible pairs of leaves and all possible trees. The average path length for separation r is found to be identical to the average depth of a leaf with $m = r$. Figure 7.15 shows that the form of the path lengths in Catalan trees is very different from those found in tSDRG (chapter 5) and complete binary trees (chapter 6), exhibiting a square root asymptotic behaviour rather than a logarithmic one. When constructing the trees according to a random vector as described in section 7.6, which allows degeneracy of the tree structures, the logarithmic asymptotic form is recovered, but is still not identical to the tSDRG results. Therefore it is clear that the set of trees under consideration plays a crucial role in the form of the average path length and can alter their asymptotic form. The selection process performed by the renormalisation group plays a critical role in choosing the trees that make up the set in tSDRG.

Upon completion of the work on Catalan trees and writing of this thesis

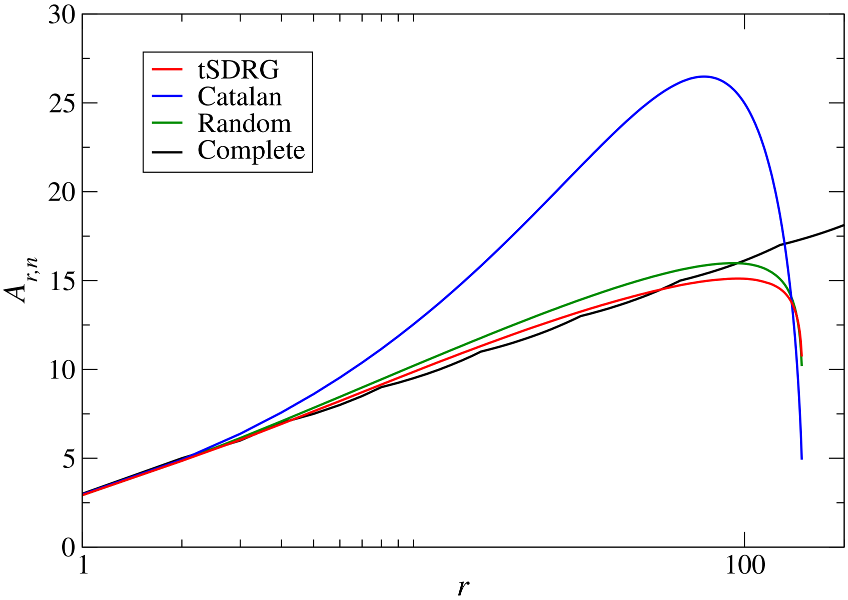

Figure 7.15: Comparing the leaf-to-leaf path lengths of the various trees studied thus far, all with $L = 150$ apart from the complete binary tree where we show the infinite limit. The red line gives path lengths from 2000 disorder realisations of tSDRG (from fig. 5.11). The blue line shows the average path length from Catalan trees using eq. (7.5.44). The green line is the average path length for 10000 random binary trees as discussed in section 7.6. The black line is $\mathcal{L}_{\infty}^{(2)}(r)$ from the complete binary trees in chapter 6.

chapter we learned of a proof of the average depth of a leaf of position m (eq. (7.5.43)) by Kirchenhofer [173, 174] and a generalisation to all statistical moments [175]. We have so far been unable to obtain refs. [173, 174], but the methods outlined in this chapter are distinct from those used in [175], whilst finding the same result. However, to the best of our knowledge, the discussion of the leaf-to-leaf path lengths is still novel.

Chapter 8

Summary and Outlook

8.1 Summary

To summarise, this thesis attempts to provide a pedagogical introduction to the field of tensor networks, particularly for use on disordered quantum many-body systems. The discussion of geometry and tensor networks is not new to this work, but we extend the discussion from regular, translationally invariant networks to those with more complex structures where the geometry is based on the random nature of the underlying model. We also provide examples of the effect of the geometry, which we believe are missing from the more conceptual papers on this subject.

The basic concepts of tensor networks were introduced in the context of an MPS-based DMRG algorithm. The diagrammatic notation that is common within most tensor network papers and talks was introduced and utilised throughout. The notion of contraction, reshaping and SVD, which are the fundamental actions that can be performed in tensor network algorithms, were discussed in the hope that the reader might use them to write their own routines should they wish. A general approach to the construction of MPO-based Hamiltonians was introduced with various examples given. The method by which DMRG is performed on MPSs is shown for the case of the spin-1/2 Heisenberg model. This method is at the heart of many modern DMRG algorithms for example the ITensor packages [79] used in chapter 3. We used the newly gained knowledge of DMRG to perform a detailed analysis of the phase diagram of the disordered Bose-Hubbard model. Because DMRG encodes the entanglement properties of the wavefunction, we use this as a general observable to obtain the phase throughout the entire phase space without having to perform multiple runs for every realisation.

We discuss the area law, which explains why DMRG works so well in certain situations and poorly in others [97]. The arguments are based on geometry, suggesting that DMRG is a special case within a wider range of tensor network states and that the structure of the network plays an important role in its success. The extension of the area law to critical systems highlights the connection between tensor networks and holographic field theories based on AdS/CFT [44]. In light of these holographic principles, we give examples of more appropriate tensor networks that can be used for critical and two dimensional systems.

In chapter 5 we reformulate the numerical SDRG of Hikihara [124] for the disordered spin-1/2 Heisenberg model in terms of a tensor network. This approach traces out a TTN where the position of the tensors is determined by the disorder in the Hamiltonian rather than placed by hand as in all other tensor network approaches. It is known for the disordered spin-1/2 Heisenberg model that upon disorder averaging an effective CFT emerges. We suggest that the logarithmic scaling of entanglement and power law decay of correlations are due to the holographic nature of the wavefunction.

For the regular and translationally invariant tensor networks it is clear what form the asymptotic average correlation function should take. For MPS it is simply the number of sites between the two points of the correlation function; for MERA there is a logarithmic scaling; for TTNs it is not so clear. The structure is not translationally invariant and there are holographic structures that mean that for a given separation there are many different path lengths that contribute to the average. We analyse the complete binary tree graph as it has the same structure as the regular TTN. We make a restriction that the leaves of the tree are fixed in an order that mimics a one dimensional lattice and calculate the average path length as a function of leaf separation. This result is generalised for m-ary splittings and all statistical moments. The discussion of the structure of TTNs continues with the Catalan trees which is the set of all unique binary trees with n internal vertices. We again make the restriction that the leaves form the one dimensional lattice. We find an analytic form for the depth of a lattice point m as well as the average path length for separation $1, 2, 3$. It was shown by Jon Fellows and co-authors in [14] that the general form for the average path length of separation r is identical to the average leaf depth at position $m = r$. The asymptotic form of the path lengths of complete trees, Catalan trees and randomly generated trees are compared with the results of the tSDRG trees to conclude that the choice of trees in the set plays a crucial role in the form of the resulting average correlation function.

8.2 Outlook

The use of tensor networks within the fields of condensed matter physics and quantum information theory is becoming ever more common. MPS-based DMRG is widely believed to be the most accurate method of numerically modelling one dimensional systems and it is being applied in increasingly complicated scenarios [2, 30]. Projected entangled pair states (PEPS) are being used both as a numerical method and as an analytic platform to uncover topological properties of matter [104, 105]. MERA and holographic tensor networks have become a useful tool within high energy physics and are being applied in linking entanglement with gravity in string theory [176].

There are many ways that tensor networks can aid the study of disordered systems. Although DMRG is in some ways imperfect for the modelling of disorder, it is so efficient that much can still be learned by applying it. Beyond the Heisenberg and Bose-Hubbard models discussed in the thesis, there are still a myriad of possible Hamiltonians that can be examined with DMRG. A current area of intense research is many-body localisation (MBL), the generalisation of Anderson localisation to interacting many-body systems [177]. It is believed that the area law holds for all excited states in systems with MBL up to some mobility energy, unlike gapped quantum systems where only the ground state is area law satisfying [178]. This in principle should allow for an efficient MPS representation, and therefore accurate DMRG simulation, of any state in a one-dimensional MBL spectrum. Strong disorder renormalisation techniques such as tSDRG can be used as high precision methods when disorder is strong. The method should be accurate for use with the FM/AFM disordered spin-1/2 Heisenberg model where large effective spins would be created as the renormalisation progresses [124]. Beyond spin-1/2 there have been exciting discoveries in disordered spin-3/2 Heisenberg systems, where the rich phase diagram contains topological phases as well as spin doublet and triplet phases [179]. It would be fascinating to uncover the optimal tensor network geometries in these situations.

More generally we would like to be able to construct an algorithm that can decide on the best network geometry for any system under consideration. Currently the geometry in most tensor network approaches is set by hand using prior knowledge of the model. In a network that can self optimise the structure, the final geometry can become a resource for learning about the properties of the wavefunction. Perhaps with these ideas, truly scalable two and three dimensional tensor network algorithms may be a possibility with and without disorder.

Appendix A

Proof of Catalan Number Equations

A.1 Changing the Order of the Sum

It is possible to change the order in which a sum is performed, i.e. instead of counting the indices upwards from some lower limit, count down from the upper limit. It is trivial to perform this change, but can be useful in simplifying certain sums. The general form is

$$\sum_{i=\alpha}^{\beta} x_i = \sum_{i'=\alpha}^{\beta} x_{\alpha+\beta-i'}. \tag{A.1.1}$$

A.2 Catalan Number Relations

The following expressions are frequently used throughout the chapter, this section gives proofs for them. The generating function for the Catalan numbers satisfies

$$\mathcal{C}(x) = \frac{1}{1 - x\mathcal{C}(x)} \tag{A.2.1}$$

Proof Starting with eq. (7.3.6) isolate one $\mathcal{C}(x)$

$$x\mathcal{C}^2(x) - \mathcal{C}(x) + 1 = 0 \tag{A.2.2}$$

$$\mathcal{C}(x)\left[1 - x\mathcal{C}(x)\right] = 1 \tag{A.2.3}$$

$$\mathcal{C}(x) = \frac{1}{1 - x\mathcal{C}(x)} \qquad \blacksquare \tag{A.2.4}$$

$\mathcal{C}^2(x)$ can be expressed as

$$\mathcal{C}^2(x) = \sum_{n=0}^{\infty} C_{n+1} x^n. \tag{A.2.5}$$

Proof Starting from 7.3.5

$$\begin{aligned}\mathcal{C}^2(x) &= \frac{\mathcal{C}(x) - 1}{x} \\ &= \sum_{n=0}^{\infty} C_n x^{n-1} - x^{-1} \\ &= \sum_{n=1}^{\infty} C_n x^{n-1} \\ &= \sum_{n=0}^{\infty} C_{n+1} x^n \qquad \blacksquare \end{aligned} \tag{A.2.6}$$

The following two expressions are extensions of Segner's relation (7.2.1) and are used in the derivation of the nearest neighbour ($r = 1$) path lengths

$$(n+1)C_{n+1} = \sum_{q=0}^{n} (2q+1) C_q C_{n-q}. \tag{A.2.7}$$

Proof Using Segner's relation, the right hand side can be written as a *square* of terms that takes into account the degeneracy given by $(2q + 1)$. This can be easily seen by first considering degeneracy $(q + 1)$

$$\begin{aligned}\sum_{q=0}^{n} (q+1) C_q C_{n-q} &= C_0 C_n + C_1 C_{n-1} + C_2 C_{n-2} + \cdots + C_n C_0 \\ &+ \;\; 0 \;\; + C_1 C_{n-1} + C_2 C_{n-2} + \cdots + C_n C_0 \\ &+ \;\; 0 \;\; + \;\; 0 \;\; + C_2 C_{n-2} + \cdots + C_n C_0 \ldots \end{aligned} \tag{A.2.8}$$

This multiplied by two completes this square but double counts the terms along the diagonal, which need to be removed. The complete square is clearly $(n + 1)$ copies of Segner's relation.

$$(n+1)C_{n+1} = 2\sum_{q=0}^{n} (q+1) C_q C_{n-q} - \sum_{q=0}^{n} C_q C_{n-q} \tag{A.2.9}$$

$$= \sum_{q=0}^{n} (2q+1) C_q C_{n-q} \qquad \blacksquare \tag{A.2.10}$$

A similar related expression is

$$(2n+1)C_n = \sum_{q=0}^{n}(q+1)C_qC_{n-q}. \tag{A.2.11}$$

Proof Starting with Eqn. (A.2.9), noticing that the final term is just Segner's relation for C_{n+1}

$$(n+1)C_{n+1} = 2\sum_{q=0}^{n}(q+1)C_qC_{n-q} - C_{n+1} \tag{A.2.12}$$

$$\frac{(n+2)C_{n+1}}{2} = \sum_{q=0}^{n}(q+1)C_qC_{n-q}. \tag{A.2.13}$$

Using the recursion relation (7.3.2) gives

$$\frac{(n+2)}{2}\frac{2(2n+1)}{n+2}C_n = \sum_{q=0}^{n}(q+1)C_qC_{n-q} \tag{A.2.14}$$

$$(2n+1)C_n = \sum_{q=0}^{n}(q+1)C_qC_{n-q} \qquad \blacksquare \tag{A.2.15}$$

In the derivation of the $r = 1$ path length (eq. (7.5.11)) it is necessary to simplify various expressions involving $1/(\sqrt{1-4x})$.

$$\frac{1}{\sqrt{1-4x}} = \sum_{\alpha=0}^{\infty}(\alpha+1)C_\alpha x^\alpha. \tag{A.2.16}$$

Proof

$$\begin{aligned}\sum_{\alpha=0}^{\infty}(\alpha+1)C_\alpha x^\alpha &= \mathcal{C}(x) + x\frac{\partial}{\partial x}\mathcal{C}(x) \\ &= \frac{1-\sqrt{1-4x}}{2x} + \left[\frac{1}{\sqrt{1-4x}} - \frac{1-\sqrt{1-4x}}{2x}\right] \\ &= \frac{1}{\sqrt{1-4x}} \qquad \blacksquare \end{aligned} \tag{A.2.17}$$

This result is then used for the other parts of eq. (7.5.13)

$$\frac{\mathcal{C}(x)}{\sqrt{1-4x}} = \sum_{n=0}^{\infty}(2n+1)C_nx^n. \tag{A.2.18}$$

Proof

$$
\begin{aligned}
\frac{\mathcal{C}(x)}{\sqrt{1-4x}} &= \sum_{\alpha=0}^{\infty} C_\alpha x^\alpha \sum_{\beta=0}^{\infty} (\beta+1) C_\beta x^\beta \\
&= \sum_{\alpha,\beta=0}^{\infty} (\beta+1) C_\alpha C_\beta x^{\alpha+\beta} \\
&= \sum_{n=0}^{\infty} \sum_{\beta=0}^{n} (\beta+1) C_{n-\beta} C_\beta x^n \qquad \text{(A.2.19)} \\
&= \sum_{n=0}^{\infty} (2n+1) C_n x^n \qquad \blacksquare \qquad \text{(A.2.20)}
\end{aligned}
$$

where in the last line we have used A.2.11. Finally

$$
\frac{\mathcal{C}^2(x)}{\sqrt{1-4x}} = \sum_{n=0}^{\infty} (n+1) C_{n+1} x^n. \qquad \text{(A.2.21)}
$$

Proof Starting with eq. (A.2.18)

$$
\begin{aligned}
\frac{\mathcal{C}^2(x)}{\sqrt{1-4x}} &= \mathcal{C}(x) \left(\frac{\mathcal{C}(x)}{\sqrt{1-4x}} \right) \\
&= \sum_{\alpha=0}^{\infty} C_\alpha x^\alpha \sum_{\beta=0}^{\infty} (2\beta+1) C_\beta x^\beta \\
&= \sum_{\alpha,\beta=0}^{\infty} (2\beta+1) C_\alpha C_\beta x^{\alpha+\beta} \\
&= \sum_{n=0}^{\infty} \sum_{\beta=0}^{n} (2\beta+1) C_{n-\beta} C_\beta x^n \\
&= \sum_{n=0}^{\infty} (n+1) C_{n+1} x^n \qquad \blacksquare \qquad \text{(A.2.22)}
\end{aligned}
$$

where in the last line we used A.2.7.

A.3 Left-Right Symmetry of the Depth function

As mentioned in section 7.4.3 the depth function (7.4.21) is left-right symmetric such that $D_{m,n} = D_{n+2-m,n}$. *Proof* In the same way as eq. (7.4.21), a recursion

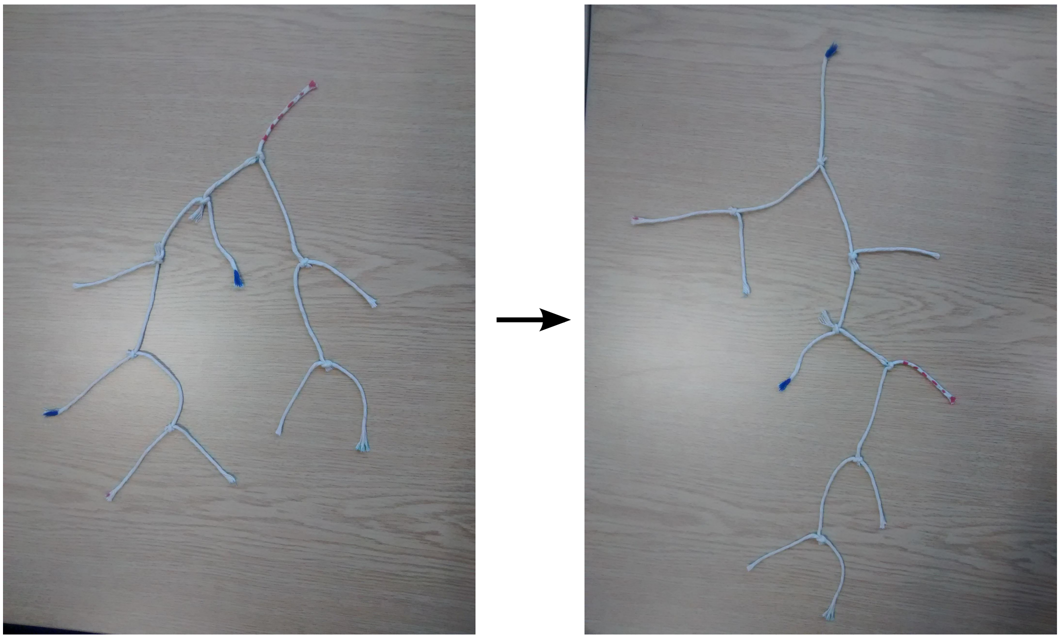

Figure A.1: A Catalan tree with $n = 7$ constructed out of string. The path with $r = 3$, highlighted by the blue leaves can be deformed so that one leaf sits above the root node and the other is at $m = 3$. The dashed red edge denotes the root node.

relation can be found for the pth leg from the right

$$D_{n+2-p,n} = C_n + \sum_{k=0}^{n-p} D_{n-k-p+1,n-k-1} C_k + \sum_{k=n-p+1}^{n-1} D_{n+2-p,k} C_{n-k-1}. \qquad \text{(A.3.1)}$$

The first sum is where the $(n+2-p)$th leg is $(n-k-p+1)$ in the right sub-tree. The second sum is where the left sub-tree is large enough to have $n+2-p$ legs. The degeneracies are given by the number of trees in the opposite block as before. Setting $m = n+2-p$ recovers eq. (7.4.21) as desired. ■

A.4 Relationship Between Path Length and Leaf Depth

A neat way of visualising the connection between path lengths and leaf depths is by constructing a physical tree, in fig. A.1 out of string, and deforming it. In this way all leaf-to-leaf paths can be mapped to leaf depths. Figure A.2 shows all possible leaf-to-leaf paths and their corresponding leg depths for the set of Catalan trees with $n = 3$.

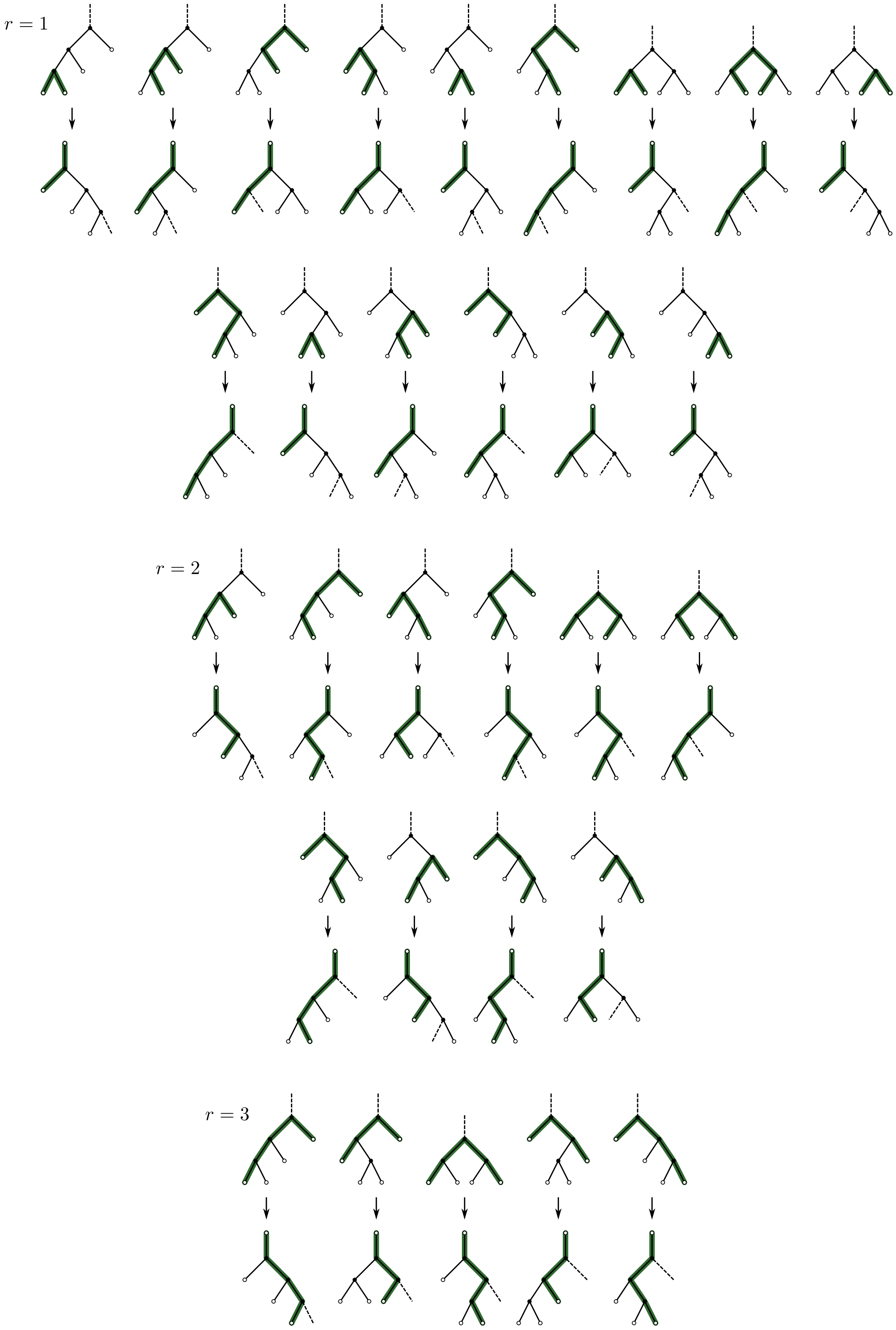

Figure A.2: All leaf-to-leaf paths and their corresponding leaf depths for all Catalan trees with $n = 3$. Symbols and lines as in fig. 7.1 apart from the dashed line that denotes the edge that would connect to the root from above.

Appendix B

Word Cloud

Figure B.1 is a diagrammatic representation of the frequency of use of words in this thesis, known as a *word cloud*, created using the *Wordle* applet [180]. It gives an interesting and fun way of summarising the key ideas within the text, but is no substitution for actually reading it!

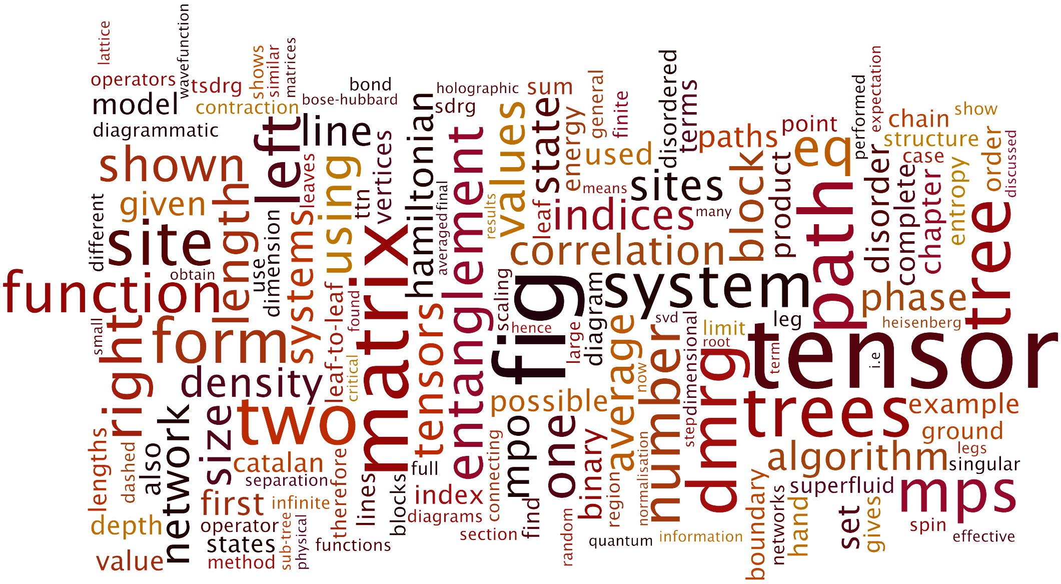

Figure B.1: A Word cloud illustrating the frequency of use of words in this thesis. The size of each word is determined by the number of times it appears, larger words appear more frequently.

Bibliography

[1] A. Altland and B. Simons. *Condensed Matter Field Theory.* Cambridge University Press, Cambridge, second edition, 2010.

[2] R. Orús. A practical introduction to tensor networks: Matrix product states and projected entangled pair states. *Ann. Phys.*, 349(0):117 – 158, June 2014.

[3] D. Poulin, A. Qarry, R. Somma, and F. Verstraete. Quantum simulation of time-dependent hamiltonians and the convenient illusion of Hilbert space. *Phys. Rev. Lett.*, 106:170501, April 2011.

[4] B. Swingle. *The Structure of Many-body Entanglement.* PhD thesis, Massachusetts Institute of Technology, September 2011.

[5] T. Pruschke. Dynamical mean-field approximation and cluster methods for correlated electron systems. In H. Fehske, R. Schneider, and A. Weie, editors, *Computational Many-Particle Physics*, volume 739 of *Lecture Notes in Physics*, pages 473–503. Springer Berlin Heidelberg, 2008.

[6] T. Tsuneda. *Density Functional Theory in Quantum Chemistry.* Springer Japan, 2014.

[7] K. G. Wilson. The renormalization group: Critical phenomena and the Kondo problem. *Rev. Mod. Phys*, 47(4):773–840, October 1975.

[8] M. Troyer and U.-J. Wiese. Computational complexity and fundamental limitations to fermionic quantum Monte Carlo simulations. *Phys. Rev. Lett.*, 94: 170201, May 2005.

[9] U. Schollwöck. The density-matrix renormalization group. *Rev. Mod. Phys.*, 77:259–315, April 2005.

[10] P. W. Anderson. Absence of Diffusion in Certain Random Lattices. *Phys. Rev.*, 109:1492–1505, March 1958.

[11] A. M. Goldsborough and R. A. Römer. Using entanglement spectra to discern phases in the disordered Bose-Hubbard model. *arXiv:1503.02973 [cond-mat.dis-nn]*, March 2015.

[12] A. M. Goldsborough and R. A. Römer. Self-assembling tensor networks and holography in disordered spin chains. *Phys. Rev. B*, 89:214203, June 2014.

[13] A. M. Goldsborough, S. A. Rautu, and R. A. Römer. Leaf-to-leaf distances and their moments in finite and infinite ordered m-ary tree graphs. *Phys. Rev. E*, 91:042133, April 2015.

[14] A. M. Goldsborough, J. M. Fellows, S. A. Rautu, M. Bates, G. Rowlands, and R. A. Römer. Leaf-to-leaf distances in ordered Catalan tree graphs. *arXiv:1502.07893 [math-ph]*, February 2015.

[15] S. R. White. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.*, 69:2863–2866, November 1992.

[16] R. M. Noack and S. R. White. The density matrix renormalization group. In I. Peschel, X. Wang, M. Kaulke, and K. Hallberg, editors, *Density Matrix Renormalization: A New Numerical Method in Physics*, pages 27–66. Springer Berlin Heidelberg, 1999.

[17] T. Costi. Wilson's numerical renormalization group. In I. Peschel, X. Wang, M. Kaulke, and K. Hallberg, editors, *Density Matrix Renormalization: A New Numerical Method in Physics*, pages 3–25. Springer Berlin Heidelberg, 1999.

[18] S. R. White. Density-matrix algorithms for quantum renormalization groups. *Phys. Rev. B*, 48:10345–10356, October 1993.

[19] K. A. Hallberg. New trends in density matrix renormalization. *Adv. Phys.*, 55(5-6):477–526, July 2006.

[20] S. Östlund and S. Rommer. Thermodynamic limit of density matrix renormalization. *Phys. Rev. Lett.*, 75:3537–3540, November 1995.

[21] G. Vidal. Efficient simulation of one-dimensional quantum many-body systems. *Phys. Rev. Lett.*, 93:040502, July 2004.

[22] A. J. Daley, C. Kollath, U. Schollwck, and G. Vidal. Time-dependent density-matrix renormalization-group using adaptive effective Hilbert spaces. *Journal of Statistical Mechanics: Theory and Experiment*, 2004(04):P04005, April 2004.

[23] S. R. White and A. E. Feiguin. Real-time evolution using the density matrix renormalization group. *Phys. Rev. Lett.*, 93:076401, August 2004.

[24] G. Vidal. Classical simulation of infinite-size quantum lattice systems in one spatial dimension. *Phys. Rev. Lett.*, 98:070201, February 2007.

[25] F. Verstraete, D. Porras, and J. I. Cirac. Density matrix renormalization group and periodic boundary conditions: A quantum information perspective. *Phys. Rev. Lett.*, 93:227205, November 2004.

[26] P. Pippan, S. White, and H. Evertz. Efficient matrix-product state method for periodic boundary conditions. *Phys. Rev. B*, 81:081103, February 2010.

[27] M. Weyrauch and M. V. Rakov. Efficient MPS algorithm for periodic boundary conditions and applications. *Ukr. J. Phys.*, 58(7):657–665, July 2013.

[28] I. Peschel and V. Eisler. The conceptual background of density-matrix renormalization. In H. Fehske, R. Schneider, and A. Weiße, editors, *Computational Many-Particle Physics*, volume 739 of *Lect. Notes Phys.*, pages 581–596. Springer Berlin Heidelberg, 2008.

[29] G. M. Crosswhite and D. Bacon. Finite automata for caching in matrix product algorithms. *Phys. Rev. A*, 78:012356, July 2008.

[30] U. Schollwöck. The density-matrix renormalization group in the age of matrix product states. *Ann. Phys.*, 326(1):96–192, January 2011.

[31] A. Milsted, J. Haegeman, T. J. Osborne, and F. Verstraete. Variational matrix product ansatz for nonuniform dynamics in the thermodynamic limit. *Phys. Rev. B*, 88:155116, October 2013.

[32] M. L. Wall and L. D. Carr. Out-of-equilibrium dynamics with matrix product states. *New J. Phys.*, 14(12):125015, December 2012.

[33] N. Nakatani and G. K.-L. Chan. Efficient tree tensor network states (TTNS) for quantum chemistry: Generalizations of the density matrix renormalization group algorithm. *J. Chem. Phys.*, 138(13):134113, April 2013.

[34] G. Evenbly and G. Vidal. Algorithms for entanglement renormalization. *Phys. Rev. B*, 79:144108, April 2009.

[35] J. I. Cirac and F. Verstraete. Renormalization and tensor product states in spin chains and lattices. *J. Phys. A: Math. Gen.*, 42(50):504004, December 2009.

[36] F. Fröwis, V. Nebendahl, and W. Dür. Tensor operators: Constructions and applications for long-range interaction systems. *Phys. Rev. A*, 81:062337, June 2010.

[37] S. Singh, R. N. C. Pfeifer, and G. Vidal. Tensor network states and algorithms in the presence of a global U(1) symmetry. *Phys. Rev. B*, 83:115125, March 2011.

[38] B. Pirvu, F. Verstraete, and G. Vidal. Exploiting translational invariance in matrix product state simulations of spin chains with periodic boundary conditions. *Phys. Rev. B*, 83:125104, March 2011.

[39] A. Weichselbaum. Non-abelian symmetries in tensor networks: A quantum symmetry space approach. *Ann. Phys.*, 327(12):2972 – 3047, July 2012.

[40] F. Pollmann and A. M. Turner. Detection of symmetry-protected topological phases in one dimension. *Phys. Rev. B*, 86:125441, September 2012.

[41] H. N. Phien, G. Vidal, and I. P. McCulloch. Dynamical windows for real-time evolution with matrix product states. *Phys. Rev. B*, 88:035103, July 2013.

[42] G. Vidal. Entanglement renormalization. *Phys. Rev. Lett.*, 99:220405, November 2007.

[43] L. Tagliacozzo, G. Evenbly, and G. Vidal. Simulation of two-dimensional quantum systems using a tree tensor network that exploits the entropic area law. *Phys. Rev. B*, 80:235127, December 2009.

[44] G. Evenbly and G. Vidal. Tensor network states and geometry. *J. Stat. Phys.*, 145(4):891–918, June 2011.

[45] J. Molina-Vilaplana. Holographic geometries of one-dimensional gapped quantum systems from tensor network states. *J. High Energ. Phys.*, 2013(5):1–25, May 2013.

[46] B. Pirvu, V. Murg, J. I. Cirac, and F. Verstraete. Matrix product operator representations. *New Journal of Physics*, 12(2):025012, February 2010.

[47] F. Verstraete, V. Murg, and J. I. Cirac. Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems. *Adv. Phys.*, 57(2):143–224, March-April 2008.

[48] I. P. McCulloch. From density-matrix renormalization group to matrix product states. *J. Stat. Mech. Theor. Exp.*, 2007(10):P10014, October 2007.

[49] F. H. L. Essler, H. Frahm, F. Göhmann, A. Klümper, and V. E. Korepin. *The One-Dimensional Hubbard Model*. Cambridge University Press, Cambridge, 2005.

[50] A. Juozapavičious. *Density-Matrix Renormalization-Group Analysis of Kondo and XY models*. PhD thesis, Royal Institute of Technology, 2001.

[51] D. Bohr. *The Density Matrix Renormalization Group Applied to mesoscopic structures*. PhD thesis, Technical University of Denmark, April 2004.

[52] W. Whiteley. Counting out to the flexibility of molecules. *Phys. Biol.*, 2: S116–S126, November 2005.

[53] M. P. A. Fisher, P. B. Weichman, G. Grinstein, and D. S. Fisher. Boson localization and the superfluid-insulator transition. *Phys. Rev. B*, 40:546–570, July 1989.

[54] H. S. J. van der Zant, W. J. Elion, L. J. Geerligs, and J. E. Mooij. Quantum phase transitions in two dimensions: Experiments in Josephson-junction arrays. *Phys. Rev. B*, 54:10081–10093, October 1996.

[55] D. B. Haviland, Y. Liu, and A. M. Goldman. Onset of superconductivity in the two-dimensional limit. *Phys. Rev. Lett.*, 62:2180–2183, May 1989.

[56] T. Pang. Universal critical normal sheet resistance in ultrathin films. *Phys. Rev. Lett.*, 62:2176–2179, May 1989.

[57] M. Greiner, O. Mandel, T. Esslinger, T. W. Hänsch, and I. Bloch. Quantum phase transition from a superfluid to a Mott insulator in a gas of ultracold atoms. *Nature*, 415:39–44, January 2002.

[58] U. Gavish and Y. Castin. Matter-wave localization in disordered cold atom lattices. *Phys. Rev. Lett.*, 95:020401, July 2005.

[59] P. Horak, J.-Y. Courtois, and G. Grynberg. Atom cooling and trapping by disorder. *Phys. Rev. A*, 58:3953–3962, November 1998.

[60] L. Sanchez-Palencia, D. Clément, P. Lugan, P. Bouyer, G. V. Shlyapnikov, and A. Aspect. Anderson localization of expanding Bose-Einstein condensates in random potentials. *Phys. Rev. Lett.*, 98:210401, May 2007.

[61] J. Billy, V. Josse, Z. Zuo, A. Bernard, B. Hambrecht, P. Lugan, D. Clement, L. Sanchez-Palencia, P. Bouyer, and A. Aspect. Direct observation of Anderson localization of matter waves in a controlled disorder. *Nature*, 453:891–894, June 2008.

[62] W. Krauth. Bethe ansatz for the one-dimensional boson Hubbard model. *Phys. Rev. B*, 44:9772–9775, November 1991.

[63] T. Giamarchi and H. Schulz. Localization and interaction in one-dimensional quantum fluids. *Europhys. Lett.*, 3:1287–1293, June 1987.

[64] Z. Ristivojevic, A. Petković, P. Le Doussal, and T. Giamarchi. Phase transition of interacting disordered bosons in one dimension. *Phys. Rev. Lett.*, 109: 026402, July 2012.

[65] R. T. Scalettar, G. G. Batrouni, and G. T. Zimanyi. Localization in interacting, disordered, Bose systems. *Phys. Rev. Lett.*, 66:3144–3147, June 1991.

[66] N. V. Prokof'ev and B. V. Svistunov. Comment on "one-dimensional disordered bosonic Hubbard model: A density-matrix renormalization group study". *Phys. Rev. Lett.*, 80:4355–4355, May 1998.

[67] W. Krauth, N. Trivedi, and D. Ceperley. Superfluid-insulator transition in disordered boson systems. *Phys. Rev. Lett.*, 67:2307–2310, October 1991.

[68] Ş. G. Söyler, M. Kiselev, N. V. Prokof'ev, and B. V. Svistunov. Phase diagram of the commensurate two-dimensional disordered Bose-Hubbard model. *Phys. Rev. Lett.*, 107:185301, October 2011.

[69] V. Gurarie, L. Pollet, N. V. Prokof'ev, B. V. Svistunov, and M. Troyer. Phase diagram of the disordered Bose-Hubbard model. *Phys. Rev. B*, 80:214519, December 2009.

[70] L. Pollet. A review of Monte Carlo simulations for the Bose-Hubbard model with diagonal disorder. *C. R. Phys.*, 14(8):712 – 724, September 2013. Disordered systems / Systèmes désordonnés.

[71] R. V. Pai, R. Pandit, H. R. Krishnamurthy, and S. Ramashesha. One-dimensional disordered bosonic Hubbard model: A density-matrix renormalization group study. *Phys. Rev. Lett.*, 76:2937–2940, April 1996.

[72] S. Rapsch, U. Schollwöck, and W. Zwerger. Density matrix renormalization group for disordered bosons in one dimension. *Europhys. Lett.*, 46(5):559, June 1999.

[73] H. Li and F. D. M. Haldane. Entanglement spectrum as a generalization of entanglement entropy: Identification of topological order in non-abelian fractional quantum Hall effect states. *Phys. Rev. Lett.*, 101:010504, July 2008.

[74] J. Eisert, M. Cramer, and M. B. Plenio. *Colloquium*: Area laws for the entanglement entropy. *Rev. Mod. Phys.*, 82:277–306, February 2010.

[75] F. Pollmann, A. M. Turner, E. Berg, and M. Oshikawa. Entanglement spectrum of a topological phase in one dimension. *Phys. Rev. B*, 81:064439, February 2010.

[76] X. Deng and L. Santos. Entanglement spectrum of one-dimensional extended Bose-Hubbard models. *Phys. Rev. B*, 84:085138, August 2011.

[77] X. Deng, R. Citro, E. Orignac, A. Minguzzi, and L. Santos. Bosonization and entanglement spectrum for one-dimensional polar bosons on disordered lattices. *New J. Phys.*, 15(4):045023, April 2013.

[78] J. A. Kjäll, J. H. Bardarson, and F. Pollmann. Many-body localization in a disordered quantum Ising chain. *Phys. Rev. Lett.*, 113:107204, September 2014.

[79] ITensor library. URL `http://itensor.org/`. Version: 0.2.3.

[80] T. D. Kühner, S. R. White, and H. Monien. One-dimensional Bose-Hubbard model with nearest-neighbor interaction. *Phys. Rev. B*, 61:12474–12489, May 2000.

[81] G. Roux, T. Barthel, I. P. McCulloch, C. Kollath, U. Schollwöck, and T. Giamarchi. Quasiperiodic Bose-Hubbard model and localization in one-dimensional cold atomic gases. *Phys. Rev. A*, 78:023628, August 2008.

[82] T. D. Kühner and H. Monien. Phases of the one-dimensional Bose-Hubbard model. *Phys. Rev. B*, 58:R14741–R14744, December 1998.

[83] L. Pollet, N. V. Prokof'ev, B. V. Svistunov, and M. Troyer. Absence of a direct superfluid to Mott insulator transition in disordered Bose systems. *Phys. Rev. Lett.*, 103:140402, September 2009.

[84] B. V. Svistunov. Superfluid-Bose-glass transition in weakly disordered commensurate one-dimensional system. *Phys. Rev. B*, 54:16131–16134, December 1996.

[85] E. Altman, Y. Kafri, A. Polkovnikov, and G. Refael. Superfluid-insulator transition of disordered bosons in one dimension. *Phys. Rev. B*, 81:174528, May 2010.

[86] S. Pielawa and E. Altman. Numerical evidence for strong randomness scaling at a superfluid-insulator transition of one-dimensional bosons. *Phys. Rev. B*, 88:224201, December 2013.

[87] I. Peschel, M. Kaulke, and O. Legeza. Density-matrix spectra for integrable models. *Ann. Physik (Leipzig)*, 8(2):153–164, February 1999.

[88] O. Legeza and G. Fáth. Accuracy of the density-matrix renormalization-group method. *Phys. Rev. B*, 53:14349–14358, June 1996.

[89] M.-C. Chung and I. Peschel. Density-matrix spectra of solvable fermionic systems. *Phys. Rev. B*, 64:064412, July 2001.

[90] S. Sachdev. *Quantum Phase Transitions*. Cambridge University Press, 2nd edition, April 2011.

[91] S. Liang and H. Pang. Approximate diagonalization using the density matrix renormalization-group method: A two-dimensional-systems perspective. *Phys. Rev. B*, 49:9214–9217, April 1994.

[92] T. Xiang. Density-matrix renormalization-group method in momentum space. *Phys. Rev. B*, 53:R10445–R10448, April 1996.

[93] S. R. White and D. J. Scalapino. Density matrix renormalization group study of the striped phase in the 2d $t - J$ model. *Phys. Rev. Lett.*, 80:1272–1275, February 1998.

[94] M.-C. Chung and I. Peschel. Density-matrix spectra for two-dimensional quantum systems. *Phys. Rev. B*, 62:4191–4193, August 2000.

[95] S. Depenbrock, I. P. McCulloch, and U. Schollwöck. Nature of the spin-liquid ground state of the S=1/2 Heisenberg model on the kagome lattice. *Phys. Rev. Lett.*, 109:067201, August 2012.

[96] F. Kolley, S. Depenbrock, I. P. McCulloch, U. Schollwöck, and V. Alba. Entanglement spectroscopy of SU(2)-broken phases in two dimensions. *Phys. Rev. B*, 88:144426, October 2013.

[97] M. B. Hastings. An area law for one-dimensional quantum systems. *Journal of Statistical Mechanics: Theory and Experiment*, 2007(08):P08024, August 2007.

[98] G. Vidal, J. I. Latorre, E. Rico, and A. Kitaev. Entanglement in quantum critical phenomena. *Phys. Rev. Lett.*, 90:227902, June 2003.

[99] J. I. Latorre, E. Rico, and G. Vidal. Ground state entanglement in quantum spin chains. *Quantum Info. Comput.*, 4(1):48–92, January 2004.

[100] F. Verstraete and J. I. Cirac. Renormalization algorithms for quantum-many body systems in two and higher dimensions. *arXiv:cond-mat/0407066v1*, July 2004.

[101] F. Verstraete, M. M. Wolf, D. Perez-Garcia, and J. I. Cirac. Criticality, the area law, and the computational power of projected entangled pair states. *Phys. Rev. Lett.*, 96:220601, June 2006.

[102] M. Lubasch, J. I. Cirac, and M.-C. Bañuls. Algorithms for finite projected entangled pair states. *Phys. Rev. B*, 90:064425, August 2014.

[103] N. Schuch, D. Poilblanc, J. I. Cirac, and D. Pérez-García. Resonating valence bond states in the PEPS formalism. *Phys. Rev. B*, 86:115108, September 2012.

[104] P. Corboz and F. Mila. Crystals of bound states in the magnetization plateaus of the Shastry-Sutherland model. *Phys. Rev. Lett.*, 112:147203, April 2014.

[105] J. Haegeman, K. Van Acoleyen, N. Schuch, J. I. Cirac, and F. Verstraete. Gauging quantum states: From global to local symmetries in many-body systems. *Phys. Rev. X*, 5:011024, February 2015.

[106] J. McGreevy. Holographic duality with a view toward many-body physics. *Adv. High Energy Phys.*, 2010:723105, May 2010.

[107] J. Maldacena. The large-N limit of superconformal field theories and supergravity. *Int. J. Theor. Phys.*, 38(4):1113–1133, April 1999.

[108] J. Bekenstein. Black holes and the second law. *Lettere Al Nuovo Cimento Series 2*, 4(15):737–740, August 1972.

[109] J. D. Bekenstein. Black holes and entropy. *Phys. Rev. D*, 7:2333–2346, April 1973.

[110] S. W. Hawking. Black hole explosions? *Nature*, 248:30–31, March 1974.

[111] S. Ryu and T. Takayanagi. Holographic derivation of entanglement entropy from the anti-de Sitter space/conformal field theory correspondence. *Phys. Rev. Lett.*, 96:181602, May 2006.

[112] B. Swingle. Entanglement renormalization and holography. *Phys. Rev. D*, 86: 065007, September 2012.

[113] G. Vidal. Class of quantum many-body states that can be efficiently simulated. *Phys. Rev. Lett.*, 101:110501, September 2008.

[114] M. Nozaki, S. Ryu, and T. Takayanagi. Holographic geometry of entanglement renormalization in quantum field theories. *Journal of High Energy Physics*, 2012(10):193, October 2012.

[115] A. Mollabashi, M. Nozaki, S. Ryu, and T. Takayanagi. Holographic geometry of cMERA for quantum quenches and finite temperature. *arXiv:1311.6095 [hep-th]*, December 2013.

[116] R. Orús. Advances on tensor network theory: symmetries, fermions, entanglement, and holography. *Eur. Phys. J. B*, 87(11):280, November 2014.

[117] F. Iglói and C. Monthus. Strong disorder RG approach of random systems. *Phys. Rep.*, 412(5-6):277–431, February 2005.

[118] S.-k. Ma, C. Dasgupta, and C.-k. Hu. Random antiferromagnetic chain. *Phys. Rev. Lett.*, 43(19):1434–1427, November 1979.

[119] C. Dasgupta and S.-k. Ma. Low-temperature properties of the random Heisenberg antiferromagnetic chain. *Phys. Rev. B*, 22:1305–1319, August 1980.

[120] D. S. Fisher. Random antiferromagnetic quantum spin chains. *Phys. Rev. B*, 50:3799–3821, August 1994.

[121] D. S. Fisher. Critical behavior of random transverse-field Ising spin chains. *Phys. Rev. B*, 51:6411–6461, March 1995.

[122] G. Refael and J. E. Moore. Entanglement entropy of random quantum critical points in one dimension. *Phys. Rev. Lett.*, 93:260602, December 2004.

[123] E. Westerberg, A. Furusaki, M. Sigrist, and P. A. Lee. Random quantum spin chains: A real-space renormalization group study. *Phys. Rev. Lett.*, 75(23): 4302–4305, December 1995.

[124] T. Hikihara, A. Furusaki, and M. Sigrist. Numerical renormalization-group study of spin correlations in one-dimensional random spin chains. *Phys. Rev. B*, 60:12116–12124, November 1999.

[125] E. Westerberg, A. Furusaki, M. Sigrist, and P. A. Lee. Low-energy fixed points of random quantum spin chains. *Phys. Rev. B*, 55(18):12578–12593, May 1997.

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

[127] G. Vidal. Algorithms for entanglement renormalization. *arXiv:0707.1454v2*, April 2008.

[128] I. Affleck. Exact correlation amplitude for the Heisenberg antiferromagnetic chain. *J. Phys. A: Math. Gen.*, 31(20):4573, May 1998.

[129] J. Moré and D. Sorensen. Computing a trust region step. *SIAM J. Sci. Stat. Comp.*, 4(3):553–572, September 1983.

[130] J. A. Hoyos, A. P. Vieira, N. Laflorencie, and E. Miranda. Correlation amplitude and entanglement entropy in random spin chains. *Phys. Rev. B*, 76: 174425, November 2007.

[131] N. Laflorencie. Scaling of entanglement entropy in the random singlet phase. *Phys. Rev. B*, 72:140408, October 2005.

[132] G. Refael and J. E. Moore. Entanglement entropy of the random S=1 Heisenberg chain. *Phys. Rev. B*, 76:024419, July 2007.

[133] F. Iglói and Y.-C. Lin. Finite-size scaling of the entanglement entropy of the quantum Ising chain with homogeneous, periodically modulated and random couplings. *J. Stat. Mech. Theor. Exp.*, 2008(06):P06004, June 2008.

[134] G. Ramírez, J. Rodríguez-Laguna, and G. Sierra. Entanglement in low-energy states of the random-hopping model. *J. Stat. Mech. Theor. Exp.*, 2014(7): P07003, July 2014.

[135] R. Juhász and Z. Zimborás. Entanglement entropy in aperiodic singlet phases. *J. Stat. Mech. Theor. Exp.*, 2007(04):P04004, April 2007.

[136] M. den Nijs and K. Rommelse. Preroughening transitions in crystal surfaces and valence-bond phases in quantum spin chains. *Phys. Rev. B*, 40:4709–4734, September 1989.

[137] P. Lajkó, E. Carlon, H. Rieger, and F. Iglói. Disorder-induced phases in the S=1 antiferromagnetic Heisenberg chain. *Phys. Rev. B*, 72:094205, September 2005.

[138] N. Regnault, B. A. Bernevig, and F. D. M. Haldane. Topological entanglement and clustering of Jain hierarchy states. *Phys. Rev. Lett.*, 103:016801, June 2009.

[139] A. M. Läuchli, E. J. Bergholtz, J. Suorsa, and M. Haque. Disentangling entanglement spectra of fractional quantum Hall states on torus geometries. *Phys. Rev. Lett.*, 104:156404, April 2010.

[140] V. Alba, M. Haque, and A. M. Läuchli. Entanglement spectrum of the Heisenberg XXZ chain near the ferromagnetic point. *J. Stat. Mech. Theor. Exp.*, 2012 (08):P08011, August 2012.

[141] L. Lepori, G. De Chiara, and A. Sanpera. Scaling of the entanglement spectrum near quantum phase transitions. *Phys. Rev. B*, 87:235107, June 2013.

[142] A. Chandran, V. Khemani, and S. L. Sondhi. How universal is the entanglement spectrum? *Phys. Rev. Lett.*, 113:060501, August 2014.

[143] R. Sedgewick and P. Flajolet. *An Introduction to the Analysis of Algorithms.* Addison-Wesley, Westford, MA, second edition, 2013.

[144] H. Wiener. Structural determination of paraffin boiling points. *J. Am. Chem. Soc.*, 69(1):17–20, January 1947.

[145] M. E. Peskin and D. V. Schroeder. *An Introduction to Quantum Field Theory.* Westview Press, Boulder, CO, 1995.

[146] K. H. Rosen. *Discrete Mathematics and its Applications.* McGraw Hill, New York, NY, seventh edition, 2012.

[147] K. Leckey, R. Neininger, and W. Szpankowski. Towards more realistic probabilistic models for data structures: The external path length in tries under the Markov model. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA).* SIAM, 2013.

[148] L. A. Szekely, H. Wang, and T. Wu. The sum of the distances between the leaves of a tree and then 'semi-regular' property. *Discrete Math.*, 311:1197–1203, June 2011.

[149] H. Wang. Sums of distances between vertices/leaves in k-ary trees. *ICA bulletin*, 60:62–68, September 2010.

[150] P. Jacquet and M. Regnier. Normal limiting distribution for the size and the external path length of tries. Technical Report RR-0827, INRIA, April 1988.

[151] P. Kirschenhofer, H. Prodinger, and W. Szpankowski. On the balance property of Patricia tries: External path length viewpoint. *Theor. Comput. Sci.*, 68(1): 1 – 17, October 1989.

[152] P. Kirschenhofer, H. Prodinger, and W. Szpankowski. Digital search trees again revisited: The internal path length perspective. *SIAM J. Comput.*, 23 (3):598–616, June 1994.

[153] 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*, 81:062335, June 2010.

[154] M. Gerster, P. Silvi, M. Rizzi, R. Fazio, T. Calarco, and S. Montangero. Unconstrained tree tensor network: An adaptive gauge picture for enhanced performance. *Phys. Rev. B*, 90:125154, September 2014.

[155] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms.* MIT Press, Cambridge, Massachusetts, second edition, 1990.

[156] B. R. Priess. *Data Structures and Algorithms with Object-Oriented Design Patterns in C++.* John Wiley & Sons, 1999.

[157] I. S. Gradshteyn and I. M. Ryzhik. *Table of Integrals, Series and Products.* Academic Press, 1980.

[158] S. Kanemitsu, M. Katsurada, and M. Yoshimoto. On the Hurwitz-Lerch zeta-function. *Aequationes Math.*, 59(1-2):1–19, February 2000.

[159] H. Srivastava. Riemann, Hurwitz and Hurwitz-Lerch Zeta functions and associated series and integrals. In P. M. Pardalos and T. M. Rassias, editors, *Essays in Mathematics and its Applications*, pages 431–461. Springer Berlin Heidelberg, 2012.

[160] J. Guillera and J. Sondow. Double integrals and infinite products for some classical constants via analytic continuations of Lerch's transcendent. *Ramanujan J.*, 16(3):247–270, July 2008.

[161] Wolfram Research Inc. *Mathematica Version 9.0.* Champaign, IL, 2012.

[162] M. M. Wolf, G. Ortiz, F. Verstraete, and J. I. Cirac. Quantum phase transitions in matrix product systems. *Phys. Rev. Lett.*, 97:110403, September 2006.

[163] T. Koshy. *Catalan Numbers with Applications.* Oxford University Press, Oxford, second edition, 2009.

[164] P. J. Larcombe. The 18th century Chinese discovery of the Catalan numbers. *Math. Specr.*, 32(1):5–7, September 1999.

[165] P. J. Larcombe and P. D. C. Wilson. On the trail of the Catalan sequence. *Mathematics Today*, 34:114–117, August 1998.

[166] L. Euler. Enumeratio modorum, quibus figurae planae rectilineae per diagonales diuiduntur in triangula. Auctore I. A. de Segner pag. 203. *Novi. Comm. Acad. Sci. Imp. Pet.*, 7:13–15, 1761.

[167] J. A. von Segner. Enumeratio modorum, quibus figurae planae rectilineae per diagonales diuiduntur in triangula. *Novi. Comm. Acad. Sci. Imp. Pet.*, 7: 203–209, 1761.

[168] E. Catalan. Note sur une équation aux différences finies. *J. Math. Pures Appl.*, 3:508–516, October 1838.

[169] H. S. Wilf. *generatingfunctionology.* Academic Press, London, second edition, 1994.

[170] L. Råde and B. Westergren. *Mathematics Handbook for Science and Engineering.* Springer-Verlag Berlin Heidelberg, fifth edition, 2004.

[171] A. D. Polyanin and A. V. Manzhirov. *Handbook of Mathematics for Engineers and Scientists.* Chapman & Hall / CRC, Boca Raton, Florida, 2007.

[172] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* Dover Publications Inc., New York, ninth Dover printing, tenth GPO printing edition, 1964.

[173] P. Kirschenhofer. On the height of leaves in binary trees. *J. Combin. Inform. System Sci.*, 8:44–60, 1983.

[174] P. Kirschenhofer. Some new results on the average height of binary trees. *Ars Combin.*, 16A:255–260, 1983.

[175] A. Panholzer and H. Prodinger. Moments of level numbers of leaves in binary trees. *J. Stat. Plan. Inference*, 101(12):267 – 279, February 2002.

[176] B. Swingle and M. V. Raamsdonk. Universality of gravity from entanglement. *arXiv:1405.2933 [hep-th]*, May 2014.

[177] D. Basko, I. Aleiner, and B. Altshuler. Metal-insulator transition in a weakly interacting many-electron system with localized single-particle states. *Ann. Phys.*, 321(5):1126 – 1205, May 2006.

[178] A. Chandran, J. Carrasquilla, I. H. Kim, D. A. Abanin, and G. Vidal. Spectral tensor networks for many-body localization. *arXiv:1410.0687 [cond-mat.dis-nn]*, October 2014.

[179] V. L. Quito, J. A. Hoyos, and E. Miranda. Emergent SU(3) symmetry in random spin-1 chains. *arXiv:1404.1924 [cond-mat.str-el]*, April 2014.

[180] Wordle. URL `http://www.wordle.net/`. Used on 21/03/2015.