Methods for tensor networks and applications in statistical physics

Pieter Bogaert

Supervisors: Prof. Frank Verstraete, Dr. Jutho Haegeman
Counsellor: Dr. Jutho Haegeman

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Engineering Physics

WE05
Chairman: Prof. dr. Dirk Ryckbosch
Faculty of Engineering and Architecture
Academic year 2013-2014

UNIVERSITEIT
GENT

Acknowledgements

I wish to thank prof. dr. Frank Verstraete for allowing me as an engineering physics student to conduct research in theoretical physics; Michaël Mariën for describing the idea of the corner method; dr. Matthias Degroote for introducing me to the linux shell; and the whole quantum group for their interesting talks delivered at the weekly quantum lunch seminar.
I especially want to express my gratitude to dr. Jutho Haegeman for his continuous, enthusiastic support throughout my thesis year.

Permission for usage

The author gives permission to make this master dissertation available for consultation and to copy parts of this master dissertation for personal use. In the case of any other use, the limitations of the copyright have to be respected, in particular with regard to the obligation to state expressly the source when quoting results from this master dissertation.

De auteur geeft de toelating deze masterproef voor consultatie beschikbaar te stellen en delen van de masterproef te kopiëren voor persoonlijk gebruik. Elk ander gebruik valt onder de beperkingen van het auteursrecht, in het bijzonder met betrekking tot de verplichting de bron uitdrukkelijk te vermelden bij het aanhalen van resultaten uit deze masterproef.

Pieter Bogaert, juni 2014

Methods for Tensor Networks and Applications in Statistical Physics

Pieter A.W. Bogaert

Master's dissertation submitted in order to obtain the academic degree of

Master of Science in Engineering Physics

Academic Year 2013-2014

Supervisors: Prof. dr. F. Verstraete & Dr. J. Haegeman

Department of Physics and Astronomy

Department Chair: Prof. dr. D. Ryckbosch

Faculty of Engineering and Architecture
Ghent University

Keywords

Tensor Networks, Matrix Product States, Projected Entangled Pair States, Ising Model, Iterative Methods

METHODS FOR TENSOR NETWORKS AND APPLICATIONS IN STATISTICAL PHYSICS

Pieter A.W. Bogaert

Supervisors: prof. dr. Frank Verstraete & dr. Jutho Haegeman

Abstract — **In this article, a summary is given of the results of the author's master's dissertation. In that dissertation, tensor networks were used to study the classical Ising model in two and three dimensions.**

Keywords — **Tensor Networks, Matrix Product States, Projected Entangled Pair States, Ising Model, Iterative Methods**

I. INTRODUCTION

The tensor network formalism [1] is successfully used to describe quantum many-body systems with entanglement. Through quantum-classical mapping, methods for a quantum system in n dimensions can also be applied to classical systems in $n+1$ dimensions. In this article, Matrix Product States and Projected Entangled Pair States are applied to the classical Ising model in respectively two and three dimensions.

II. COMPUTATIONAL METHODS

Given a classical Hamiltonian defined on a lattice, one can compute the partition function by using transfer matrices. Suppose that the Hamiltonian only contains nearest-neighbour terms and one-site terms, then such a transfer matrix should be positioned between every two neighbouring sites. This results in a tensor network, which then needs to be contracted in order to calculate the partition function. Once this has been found, other physical quantities, such as the magnetisation or the energy, can be calculated.

Thus, the crucial step is the contraction of a given tensor network (which is always translationally invariant for our purposes). Therefore, algorithms were constructed that accomplish this in two and three dimensions. In order to circumvent an exponential scaling of the bond dimensions, truncation is an important step in these algorithms and it is the aspiration of every algorithm to find a physically justifiable way of truncating the bond dimension.

A. 2D Network Contraction

In 2D, we propose a corner method where the truncation comes down to calculating the singular value decomposition (SVD) of a certain matrix and discarding the lowest singular values. A scheme of the method is drawn in figure (1), where T is the tensor of the network we wish to contract.

Starting from two initial guesses for the horizontal fixpoint (L) and the vertical fixpoint (R) and an initial guess for the matrix S, in each step L, S, R and T are contracted to form a tensor with four open indices, as shown in the right upper corner of figure (1). In order to maximise (up to normalisation) the overlap with this tensor, M is chosen to be equal to it. An SVD of M is then calculated and the inner indices are truncated. The fixpoints L and R are then updated as shown on the left, and S is replaced by S'. This procedure is repeated until convergence.

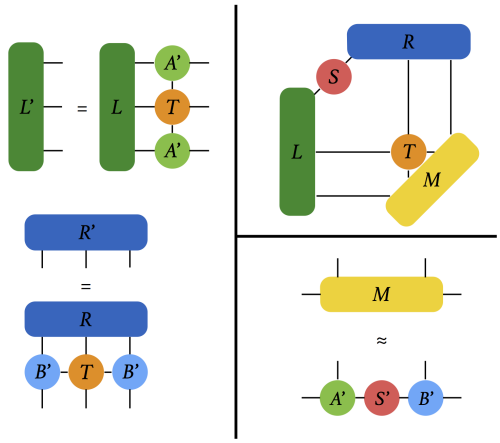


Figure 1: Some important steps in the corner method.

B. 3D Network Contraction

In 3D, there is no suitable generalisation of SVD for the truncation step. Instead, the procedure sketched in figure (2) is followed. In each iteration step, the 'vertical' fixpoint E is updated with a tensor T to form E'. The double-layer version of E' is then constructed, and the 'planar' fixpoints for the resulting 2D network are calculated using the 2D corner method.

In order to truncate the bond dimension, a projector ($P = WW^\dagger$, with W an isometry) is placed on the planar indices of E', projecting onto a subspace of a certain given dimension. The isometries W, which may differ in the two planar directions, are then optimised iteratively. A truncated E is finally obtained as shown by the dotted line in figure (2).

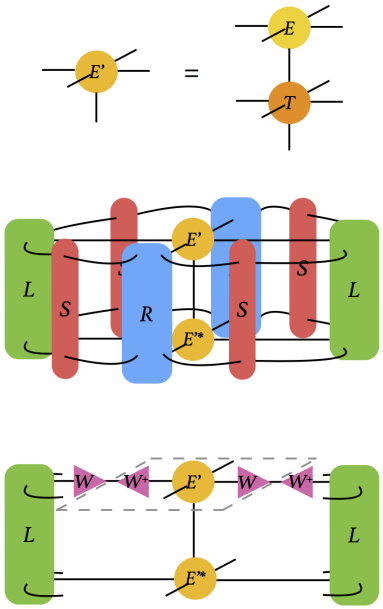


Figure 2: Schematic overview of the contraction algorithm in 3D. (In the bottom network, not all the tensors are drawn so as not to overload the picture.)

III. Classical Ising Model

The Ising model [2] studied in this article is given by the Hamiltonian

$$H = -J \sum_{<i,j>} S_i S_j - h \sum_i S_i \qquad (1)$$

defined on a lattice of spins $S_i = \pm 1$. J is the interaction strength, which is chosen $+1$ (the plus sign ensures ferromagnetic coupling), and h is an external field. The first sum goes over all lattice sites i and its nearest neighbours j. The lattice on which the Ising model is defined, is a cartesian lattice in two and three dimensions. In addition, we always work in the thermodynamical limit.

IV. The Ising Model in 2D

Figure (3) shows the average magnetisation per site of the 2D Ising model, together with the analytical result. The numerical results show an excellent agreement with the theoretical curve [3]. The estimate for the critical temperature from these numerical data yields $T_c = 2.7$ for a temperature step of 0.001, which agrees with the theoretical value $T_c \approx 2.269$. Analogous results can be obtained for the energy and the free energy, and for the $h \neq 0$ case.

V. The Ising Model in 3D

An analytical solution for the Ising model is lacking in 3D. However, it is easy to show that the free energy should obey the following limits: (1) for $T \to 0$, $f \to -3 - h$, and (2) for $T \to \infty$, $f \to -T \ln(2)$. It can be seen in figure (4) that the numerical results indeed conform to these limits.

Because the 3D algorithms become computationally very

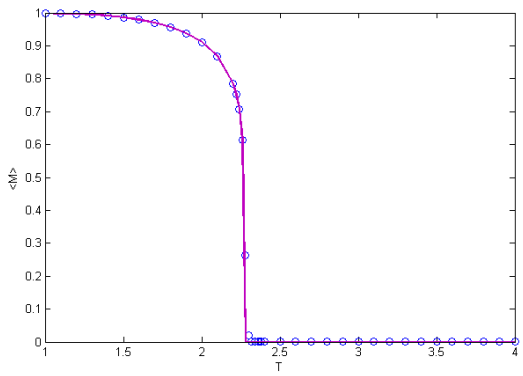


Figure 3: Magnetisation of the 2D Ising model. The solid curve is the theoretical result.

demanding, the bond dimension cannot be increased very much. For the magnetisation, this is mostly an issue around the phase transition.

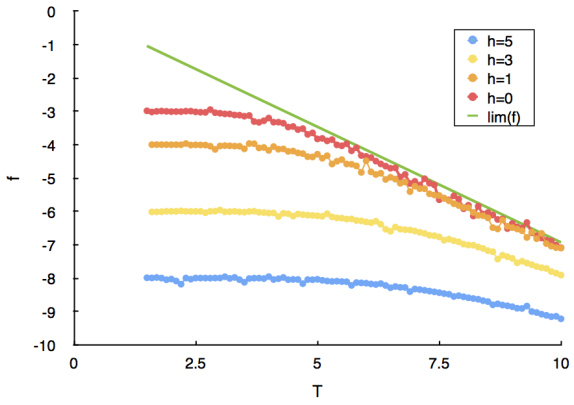


Figure 4: Free energy of the 3D Ising model. The straight line is the high-temperature limit.

VI. Conclusions

In this article, tensor networks were successfully applied to statistical physics. The computational methods generally give good results. In three dimensions, further research should be able to improve them within the constraints of the available computational power.

Acknowledgments

The author wishes to thank prof. dr. Verstraete and dr. Haegeman for promoting this work and for the support during its realisation.

References

[1] F. Verstraete, V. Murg, and J. I. Cirac. Matrix Product States, Projected Entangled Pair States, and variational Renormalization Group methods for quantum spin systems. *Advances in Physics*, 57(2):143–224, 2008.

[2] E. Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik*, 1925.

[3] R. J. Baxter. *Exactly Solved Models in Statistical Mechanics*. Academic Press, 1982.

Dutch Summary

1 Inleiding

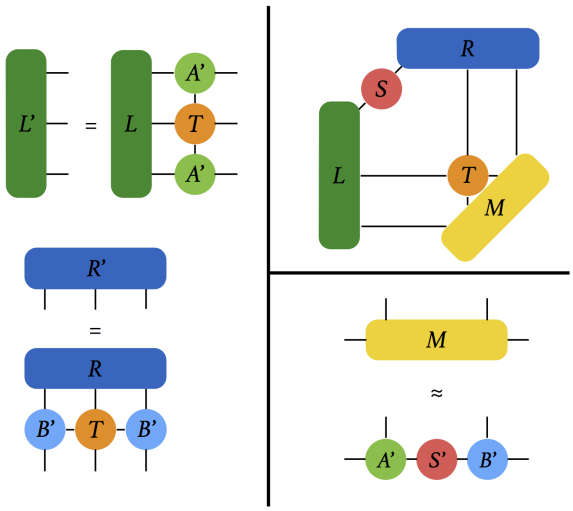
Het tensornetwerkformalisme [1] wordt met succes gebruikt om kwantumveeldeeltjessystemen met verstrengeling te beschrijven. Door *quantum-classical mapping* kunnen methodes voor kwantumsystemen in n dimensies overgezet worden naar klassieke systemen in $n + 1$ dimensies. Deze noodzakelijke achtergrondkennis wordt beschreven in Hoofdstuk 2.
In deze thesis werden *Matrix Product States* en *Projected Entangled Pair States* toegepast op het klassieke Isingmodel in respectievelijk twee en drie dimensies.

2 Computationele Methodes

Vertrekkend van een klassieke Hamiltoniaan gedefinieerd op een rooster, kan men de partitiefunctie berekenen door gebruik te maken van transfermatrices. Laat ons aannemen dat de Hamiltoniaan enkel dichtste-nabuur termen bevat, alsook termen werkend op één roosterpositie, dan kan zo een transfermatrix gepositioneerd worden tussen elke twee aangrenzende roosterposities. Het resultaat is een tensornetwerk, wat vervolgens gecontraheerd dient te worden om de partitiefunctie te bekomen. Eens die berekend is, kan men ook andere fysische grootheden, zoals de magnetisatie of de energie, berekenen.
De cruciale stap is bijgevolg het contraheren van een gegeven tensornetwerk, dat in onze gevallen steeds translatie-invariant is. Hiervoor werden algoritmes opgesteld, die dit bewerkstelligen in twee en drie dimensies (zie Hoofdstuk 3). Om exponentiële schaling van de *bond*dimensie met het aantal roosterposities te vermijden, is het trunceren van de bonddimensie een belangrijke stap in deze algoritmes en het is hun doel om het trunceren uit te voeren op fysisch verantwoorde wijze.

2.1 Contractie van een 2D Netwerk

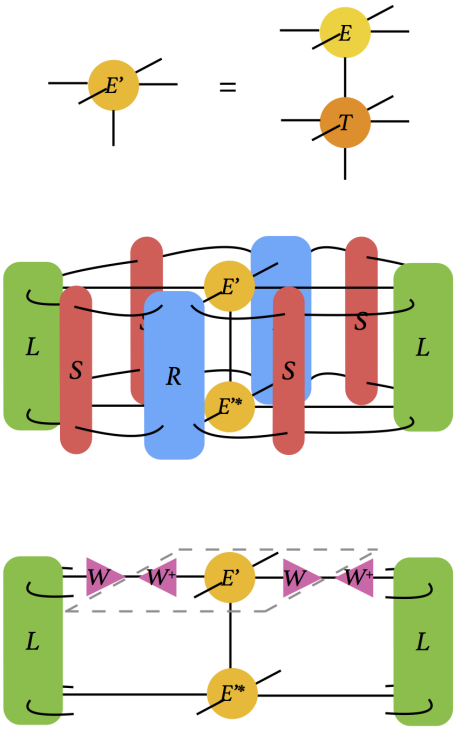
In 2D stellen we een hoekmethode voor waarin het trunceren neerkomt op het berekenen van een SVD van een zeker matrix en vervolgens de laagste singuliere waarden te elimineren. Een schema van de methode is gegeven in figuur (1), waar T de tensor is waaruit het netwerk dat we willen contraheren, is opgebouwd. Beginnend met een initiële waarde voor de horizontale en verticale *fixpoints* L en R, en een initiële matrix S, worden in elke iteratiestep L, S, R en T gecontraheerd tot een tensor met vier open indices, zoals in de rechterbovenhoek van figuur (1). Om de overlap met deze tensor te maximaliseren, op normalisatie na, wordt M eraan gelijkgesteld. Na het berekenen van de SVD van M worden de binnenindices getrunceerd. The fixpoints L en R worden dan geüpdatet zoals links getoond, en S wordt vervangen door S'. Dit wordt herhaald tot convergentie.



Figuur 1: Enkele belangrijke stappen in de hoekmethode.

2.2 Contractie van een 3D Netwerk

In 3D is er geen geschikte veralgemening van SVD voor het trunceren. In de plaats daarvan wordt de procedure gevolgd die geschetst wordt in figuur (2). In elke iteratiestap, wordt de 'verticale' fixpoint E geüpdatet met een tensor T tot E'. Van E' wordt de *double layer* geconstrueerd, en de 'planaire' fixpoints voor het resulterende 2D netwerk worden berekend met de hoekmethode. Om de bonddimensie te trunceren, worden projectoren ($P = WW^\dagger$, met W een isometrie) geplaatst op de bondindices van E', die projecteren op een deelruimte van een zekere dimensie. De isometrieën W, die kunnen verschillen in de twee planaire richtingen, worden vervolgens iteratief geoptimaliseerd. De getrunceerde E wordt dan bekomen zoals aangeduid door de stippellijn in figuur (2).



Figuur 2: Schematisch overzicht van het contractie-algoritme in3D. (In het onderste netwerk worden niet alle tensoren weergegeven om de figuur niet te overladen.)

3 Klassiek Isingmodel

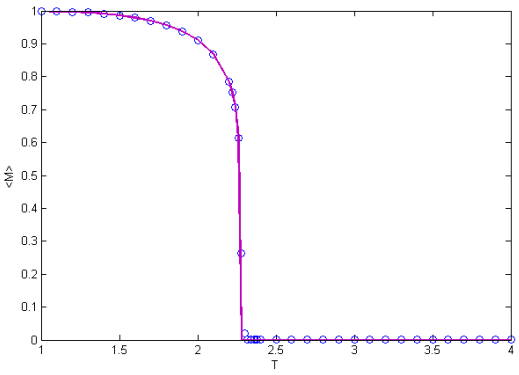
Het Isingmodel [2] wordt in deze thesis bestudeerd onder de vorm van volgende Hamiltoniaan, zie Hoofdstuk 4:

$$H = -J \sum_{<i,j>} S_i S_j - h \sum_i S_i \tag{1}$$

gedefinieerd op een rooster van spins $S_i = \pm 1$. J is de interactiesterkte, die $+1$ gekozen wordt (het plusteken zorgt voor ferromagnetische koppeling), en h is een extern veld. De eerste sommatie gaat over alle roosterposities i en hun dichtste naburen j. Het anisotrope Isingmodel, waar J varieert met de richting, werd ook in beperkte mate bestudeerd.
Het rooster waarop het Isingmodel was gedefinieerd, was een cartesiaans rooster in één, twee en drie dimensies. De oplossing van het 1D geval is eenvoudig en wordt niet besproken in deze samenvatting. In 2D is een analytische oplossing enkel gekend voor $h = 0$ [3], en in 3D ontbreekt elke analytische oplossing.

4 Het Isingmodel in 2D

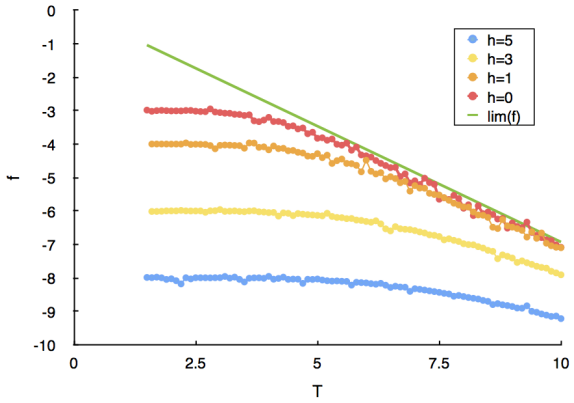
In Hoofdstuk 5 worden numerieke resultaten voor het 2D Ising model besproken. Figuur (3) toont de gemiddelde magnetisatie per roosterpositie voor het 2D Isingmodel, samen met de analytische oplossing. De numerieke resultaten vertonen een goede overeenkomst met de theoretische curve. Uit deze numerieke simulatie werd een waarde $T_c = 2.7$ bekomen voor de kritische temperatuur, met een temperatuurdiscretisatie van 0.001. Dit komt goed overeen met de theoretische waarde $T_c \approx 2.269$. Analoge resultaten worden bekomen voor de energie en de vrije energie, en voor het geval waar $h \neq 0$.



Figuur 3: Magnetisatie van het 2D Isingmodel. De doorlopende curve is het theoretische resultaat.

5 Het Isingmodel in 3D

In analogie met Hoofdstuk 5, wordt in Hoofdstuk 6 het 3D Ising model behandeld. Zoals eerder al gezegd is er geen analytische oplossing gekend in 3D. Men kan echter gemakkelijk aantonen dat de vrije energie moet voldoen aan de volgende limieten: (1) voor $T \to 0$, $f \to -3 - h$, en (2) voor $T \to \infty$, $f \to -T \ln(2)$. Het volgt duidelijk uit figuur (4) dat de numerieke resultaten inderdaad aan deze limieten gehoorzamen.
Omdat de 3D algoritmes computationeel veeleisend zijn, kan de bonddimensie maar beperkt verhoogd worden. Voor de magnetisatie heeft dit vooral gevolgen rond de fasetransitie.



Figuur 4: Vrije energie van het 3D Isingmodel. De rechte lijn is de hogetemperatuurslimiet.

6 Conclusies

In deze thesis werden tensornetwerken met succes toegepast op statistische fysica. De computationele methodes leveren over het algemeen goede resultaten op. In drie dimensies moet het mogelijk zijn ze na verder onderzoek te verbeteren, binnen de grenzen van de beschikbare rekencapaciteit.

Dankwoord

De auteur dankt prof. dr. Verstraete en dr. Haegeman voor het promoten van deze thesis en voor de begeleiding tijdens het werk eraan.

Referenties

[1] F. Verstraete, V. Murg, and J. I. Cirac. Matrix Product States, Projected Entangled Pair States, and variational Renormalization Group methods for quantum spin systems. *Advances in Physics*, 57(2):143–224, 2008.

[2] E. Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik*, 1925.

[3] R. J. Baxter. *Exactly Solved Models in Statistical Mechanics*. Academic Press, 1982.

Contents

List of Figures

List of Abbreviations

iMPS Infinite Matrix Product State

iPEPS Infinite Projected Entangled Pair States

MPS Matrix Product State

OBC Open Boundary Conditions

PBC Periodic Boundary Conditions

PEPS Projected Entangled Pair States

SVD Singular Value Decomposition

TI Translational Invariance

TN Tensor Network

Chapter 1

Introduction

1.1 Situation of the Topic

Quello che a noi è difficilissimo a intendersi, alla Natura è agevolissimo a farsi.[1] [14] Galileo Galilei (1564-1642) writes this on the fourth day of his Dialogue, when he discusses the tides. However, this quote is also well-suited to describe many-body physics. Even when the underlying Hamiltonian only contains 'simple' terms, the large number of constituents in a many-body system can cause it to become exceedingly intricate and hence computationally difficult to solve. Therefore, approximations need to be made, while still retaining the relevant physical behaviour.
It is the goal of this dissertation to analyse how tensor networks, a formalism which originated in quantum many-body physics, can be applied to study classical models in statistical physics. In particular, the two- and three-dimensional Ising model will be studied.
This will provide us with insight on two levels. First, by comparing the numerical results to the analytical solution where it exists, i.e. for the most simple case of the 2D Ising model, information is acquired on the applicability and accuracy of tensor network simulations. In the second place, where analytical solutions are unknown, the numerical results are able to provide us with a more detailed insight into the physics of the Ising model.

1.2 Origin and Use of Tensor Networks

The next chapter deals with the background on many-body physics and tensor networks that is needed for this dissertation. In this section, some non-technical remarks are made on the physical origin of tensor networks and why they are useful.
The force of tensor networks lies in the description of systems with entanglement. If there is not much entanglement in a system, it can be described using a mean field ansatz [42]. Because physically relevant Hamiltonians are composed of local terms, the ground states of those Hamiltonians occupy only a small part of the Hilbert space on which they are defined. In relation to this, bounds on the entanglement can be derived, such as in [20] for gapped Hamiltonians[2] in

[1] What is very difficult to understand for us, is very easy for Nature to make.

[2] 'Gapped' means that there is a finite separation between the lowest eigenvalue of the Hamiltonian, which may be degenerate, and its second lowest eigenvalue.

one dimension. In many cases, these bounds are so-called area laws: the entanglement entropy between two regions scales like the size of the boundary between them [9].

When constructing an ansatz, it therefore makes sense to concentrate all the entanglement at the boundary. For example, for a one-dimensional system, a linear chain of N sites, this can be achieved by maximally entangling the two sites on either side of the boundary. Since, of course, the boundary could lie between any two sites, every two neighbouring sites need to be maximally entangled. As such, each site has been split into two virtual subsystems, one of which is maximally entangled with a virtual subsystem in the site on its left and one with a virtual subsystem in the site on its right. These maximally entangled pairs are called *bonds* and their dimension is quite naturally called the bond dimension.
If the original sites are d-level spins, the last step is to map (project) the virtual subsystems from $\mathbb{C}^D \times \mathbb{C}^D$ back to $\mathbb{C}^d$. This mapping can be written using a $D \times D \times d$-tensor on every site. Hence, when writing a state as a linear combination of basis states, the coefficients are given by a product of $D \times D$ matrices. Therefore, these states are called *Matrix Product States* [41] [6]. It is easy to see that, indeed, this ansatz automatically complies with the area law: when cutting a Matrix Product State in two, the only entanglement between the two parts is due to the maximally entangled pair at the boundary. Hence, the entanglement is bounded by the bond dimension, but is independent of the number of sites in either region.

This procedure can be repeated in two dimensions [45]. In the case of a square lattice, every lattice site needs to be replaced by four virtual subsystems, each of them maximally entangled to a virtual subsystem in one of the four neighbouring sites. Nomenclature-wise, in two dimensions emphasis is laid on the projection aspect and the entangled pairs, as the states are there called *Projected Entangled Pair States*[3].
A priori, one could try to describe a two-dimensional system as a Matrix Product State by enumerating the lattice sites[4] [45]. However, as nearest-neighbour interactions become long-range then, and because the entropy does not present the right scaling for the area law, this description is inherently less suitable. Indeed, it seems more logical and natural to describe a two-dimensional state with Projected Entangled Pair States rather than forcing a one-dimensional structure upon it. That is an important point concerning the tensor network formalism: given a certain behaviour of the entanglement, they offer a very *natural* way to describe natural states that often emerge in many-body systems.

1.3 Overview of the Next Chapters

In Chapter 2, tensor networks are introduced in a more formal way. As they originally arose in the study of quantum many-body physics, the chapter includes the elements from quantum many-body physics needed to understand how this works. Two basic classes of tensor networks

[3] Technically, one could call Matrix Product States a particular instance of Projected Entangled Pair States, e.g. [6]. Yet in the literature, the term Projected Entangled Pair States is usually reserved for the two-dimensional case.

[4] E.g., row per row, alternatingly from left to right and from right to left (not unlike the ancient Greek bi-directional writing style $\beta ου \varsigma τ ροφηδόν$, meaning like an ox turning when ploughing).

are studied in some more detail, namely Matrix Product States and Projected Entangled Pair States, as those two kinds are needed in the subsequent chapters. The chapter concludes with an explanation of how these results from quantum physics can be applied to classical statistical physics.
Chapter 3 is computationally oriented and includes various algorithms to contract tensor networks in 1D, 2D and 3D. The algorithms are described, but apart from some convergence properties, the results of their application to physical systems (the classical Ising model in various dimensions) are postponed until later chapters.
The classical Ising model is introduced in Chapter 4. As an analytical solution is only known in a limited number of cases, particular attention is given to theoretical constraints, e.g. low- and high-temperature limits, which the results from the simulations should obey. As a first numerical example, results for the 1D Ising model are given at the end of the chapter.
In Chapter 5, the numerical results for the 2D Ising model are discussed. The first few sections discuss how the Ising model fits into a tensor network description. Afterwards, the numerical results are presented and analysed, both without an external field - which allows for a comparison with the analytical solution - and with an external field - for which no analytical solution is known. At the end of the chapter, a short comparison of the tensor network method to Monte Carlo methods is given.
Monte Carlo methods are further used in Chapter 6, as analytical solutions are totally lacking in 3D. Similarly to Chapter 5, the numerical results for the 3D Ising model are presented in that chapter.
Finally, an overview of this dissertation and some concluding remarks are given in the conclusion, Chapter 7.

Chapter 2

Tensor Networks

This chapter discusses the relevant theoretical background to the methods used afterwards. In the first section, tensor networks are introduced and some remarks are made on index contraction. The justification for the use of tensor networks in many-body physics is given in the next section, followed by the introduction of two important classes of tensor networks, namely Matrix Product States and Projected Entangled States. Afterwards, some mathematical results on power methods are reviewed. In the last section, the link between quantum many-body physics and classical statistical physics is elucidated. Reference is made to [9], [18], [33], [34], [35], [39], [41], [47] throughout this chapter.

2.1 Introducing Tensor Networks

2.1.1 Diagrammatic Notation for Tensors

At the risk of irritating many a mathematician, a tensor can for the purposes of this work be considered as a multidimensional array of complex or real numbers. The number of indices is called the rank of the tensor, such that a rank-0 tensor is a scalar, a rank-1 tensor a vector, and a rank-2 tensor a matrix. Since writing out tensor equations can become quite lengthy, we shall use a diagrammatic notation reminiscent of Penrose's more elaborate diagrams applicable to general relativity. Because we don't need a metric tensor et cetera, a very limited drawing scheme suffices.
Figure (2.1) shows tensors of rank zero to three. Dependent on whether the vertical index points 'up' or 'down', we can make a distinction between a bra and a ket vector (when using such a drawing convention, it should not be forgot to take the complex conjugate of the bra side). The same goes for higher-rank tensors, e.g. the rank-3 tensors drawn in figure (2.1).
In general, the dimensions of the different indices are not equal. There is, however, not always a need to make this explicit in the diagram. Nevertheless, figure (2.2) shows different options to do that, should the need arise.

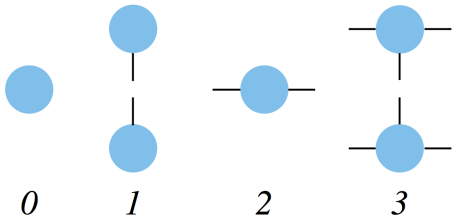


Figure 2.1: Tensors of rank 0 (scalar), 1 (vector), 2 (matrix) and 3. For rank 1 and 3, a possible convention for the distinction of kets and bras is shown.

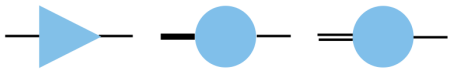


Figure 2.2: Possible ways to depict differences in the bond dimension. The bond dimension is higher on the left side than on the right side.

2.1.2 Index Contraction

The most important operation on tensors needed here, is index contraction. This is a straightforward generalisation of the matrix product

$$(AB)_{ij} = \sum_k A_{ik} B_{kj} \tag{2.1}$$

which is depicted in diagrammatic form in figure (2.3). By assembling various tensors through

Figure 2.3: Diagrammatic notation for the matrix product.

index contraction, a *tensor network* - from now on called TN - can be built up. An example is shown in figure (2.4). It should be clear that such a representation often provides more insight than an expansive non-graphical notation.

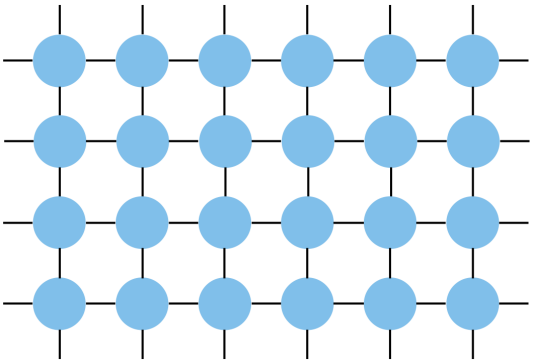


Figure 2.4: A simple two-dimensional TN.

2.1.3 Computational Cost of Index Contraction

It is important to note that even though the order of the internal contractions is not fixed theoretically, it does have an influence on the computational cost of the contraction. An example of a simple contraction is shown in figure (2.5). Let us suppose that all the index dimensions d are equal.

The first way to contract the network is to contract all vertical indices first, followed by a contraction over the horizontal indices from left to right, hence incurring a computational cost of $\mathrm{O}(d^5)$ for the vertical contractions and $\mathrm{O}(d^4)$ for the horizontal contractions.
A more efficient method, on the other hand, is to start from the left and to add one tensor at a time, alternating between the top and the bottom layer, which scales only as $\mathrm{O}(d^4)$. Since many contractions need to be calculated, it is important to contract in the right order[1].

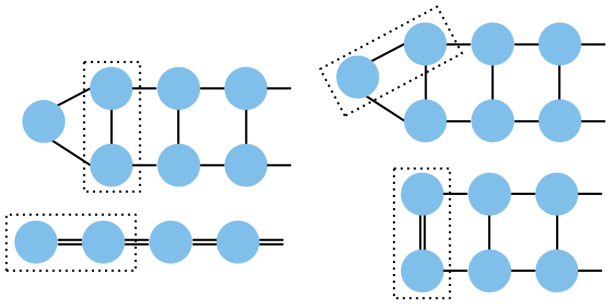


Figure 2.5: The contraction order has an influence on the number of operations required for the contraction.

2.2 Elements from Many-Body Physics

2.2.1 Hilbert Space of a Many-Body System

Let us look at a quantum many-body system consisting of N subsystems or sites. The thermodynamical limit, i.e. $N \to \infty$, will also be discussed as the methods and applications treated in the following chapters are precisely concerned with the thermodynamical limit. The term 'site' is especially natural for many-body systems defined on a lattice $\mathscr{L}$, i.e. a discrete subset of $\mathbb{R}^n$ with n the dimension of $\mathscr{L}$. In this work, we shall look only at integer lattices $\mathbb{Z}^n$ for $n = \{1, 2, 3\}$. In these dimensions, we call those lattices a linear chain, a square lattice and a cubic lattice respectively.
Each lattice site k, where in lattices of more than one dimension k is a multi-index, is occupied by a subsystem, namely a d_k-dimensional 'spin' variable, the state of which belongs to a Hilbert space $\mathcal{H}_k$. The Hilbert space $\mathcal{H}$ of the whole system can then be written as $\mathcal{H} = \bigotimes_{i=1}^{N} \mathcal{H}_k$. In practice, often the same d-dimensional subsystem occupies every site. In the classical Ising model, for example, each site is occupied by a scalar two-valued spin.
If $\{|i_k\rangle, i_k = 1, \ldots, d\}$ is a basis for $\mathcal{H}_k$, then the product basis for $\mathcal{H}$ is

$$\{|i_1 i_2 \ldots i_N\rangle\} = \{|i_1\rangle \otimes |i_2\rangle \otimes \cdots \otimes |i_N\rangle \, | i_k = 1, \ldots, d, \forall k = 1, \ldots, N\} \tag{2.2}$$

and consequently a general *pure* state $|\psi\rangle \in \mathcal{H}$ can be written as

$$|\psi\rangle = \sum_{i_1, i_2, \ldots, i_N = 1}^{d} c_{i_1, i_2, \ldots, i_N} \, |i_1 i_2 \ldots i_N\rangle \tag{2.3}$$

with the coefficients $c_{i_1, i_2, \ldots, i_N} \in \mathbb{C}$. From this formula, it is immediately clear what causes the difficulty for numerical simulations of (quantum) many-body systems. In order to fully

[1] I did not write my own algorithm for tensor contraction; this was kindly provided by my supervisor.

characterise a general state $|\psi\rangle \in \mathcal{H}$, d^N numbers $c_{i_1,i_2,\ldots,i_N}$ are needed, i.e. scaling exponentially with the lattice size. A straightforward numerical method would have an exponentially increasing memory and computational cost. Therefore, methods to circumvent this exponential scaling are a key component of simulations in many-body physics. The methods used in this work are discussed in Chapter 3.
Approximative methods (can) work because not the whole Hilbert space $\mathcal{H}$ is relevant, as already discussed in Chapter 1. The essential reason for this is that states that actually occur in Nature are not random. Because physically 'relevant' Hamiltonians consist of local few-body terms, relevant states occupy only a 'corner' of the whole Hilbert space $\mathcal{H}$.

2.2.2 Schmidt Decomposition

The general expansion of a pure state $|\psi\rangle$ was given in equation (2.3). A pure state is called a *product state* if it can be expanded as

$$|\psi\rangle = |\phi_1\rangle \otimes |\phi_2\rangle \otimes \cdots \otimes |\phi_N\rangle \tag{2.4}$$

with $|\phi_k\rangle$ states $\in \mathcal{H}_k$. Otherwise, it is called *entangled*.
Suppose that $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$, then a pure state $|\psi\rangle \in \mathcal{H}$ can be written using only one summation instead of two, by performing a *Schmidt decomposition*:

$$|\psi\rangle = \sum_{i=1}^{l} \lambda_i |\phi_A^i\rangle |\phi_B^i\rangle \tag{2.5}$$

In this decomposition, $|\phi_A^i\rangle$ and $|\phi_B^i\rangle$ are orthonormal states for subsystems A and B respectively. The coefficients $\lambda_i \in \mathbb{R}^+ \cup \{0\}$ are called the *Schmidt coefficients*, for which $\sum_i^l \lambda_i^2 = 1$.

Proof. As $|\psi\rangle$ is by definition a pure state, it can be expanded as

$$|\psi\rangle = \sum_{i_A,i_B} c_{i_A,i_B} |i_A\rangle |i_B\rangle \tag{2.6}$$

If $d_A = \dim(\mathcal{H}_A)$ and $d_B = \dim(\mathcal{H}_B)$ then c is a $d_A \times d_B$-matrix. Performing a Singular Value Decomposition (henceforth SVD), $c = u\sigma v^\dagger$, the expansion becomes

$$|\psi\rangle = \sum_{i_A,i,i_B} u_{i_A,i}\sigma_{i,i}v^*_{i_B,i} |i_A\rangle |i_B\rangle = \sum_{i=1}^{l} \sigma_{i,i} |\phi_A^i\rangle |\phi_B^i\rangle \tag{2.7}$$

with l=min(d_A, d_B) and

$$|\phi_A^i\rangle = \sum_{i_A}^{d_A} u_{i_A,i} |i_A\rangle \tag{2.8}$$

$$|\phi_B^i\rangle = \sum_{i_B}^{d_B} v^*_{i_B,i} |i_B\rangle \tag{2.9}$$

By setting $\lambda_i = \sigma_{i,i}$, we retrieve the form of equation (2.5).
Since u (respectively v) us unitary and the $|i_A\rangle$ ($|i_B\rangle$) form an orthonormal basis for the sub-

system A (B), the states $|\phi_A^i\rangle$ ($|\phi_B^i\rangle$) are still orthonormal. Lastly, because of normalisation, $\sum_i^l \lambda_i^2 = 1$. □

By successive application of this Schmidt decomposition, this can be extended to larger systems consisting of more subsystems. This is explained in more detail in section 2.3.5 when discussing the canonical form of an MPS.

2.2.3 Tensor Networks in Many-Body Physics

The coefficients $c_{i_1,i_2,\ldots,i_N}$ in equation (2.3) form a tensor of rank N. Let us call its N indices the *physical indices*. In order to be able to compute the coefficients more efficiently, the tensor c can be replaced by a tensor network with internal *bond* (or *ancillary*) *indices*[2], as shown in figure (2.6). The dimensions of the bond indices are called their *bond dimensions*, and the maximal bond dimension occurring in a TN is called *the* bond dimension of the TN. A handy convention is to write physical dimensions in small letters (usually d) and bond dimensions in capital letters (usually D).
For systems on a lattice, it is natural to assign one building block-tensor to each lattice site,

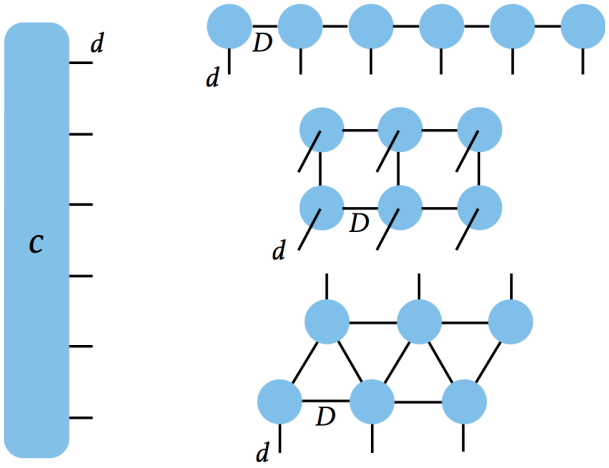


Figure 2.6: Possible TNs that can replace the original coefficient tensor c. The bond dimension is D, the physical index dimension is d.

thus scaling as $\mathrm{O}(N)$. Each tensor then has one physical index and k bond indices (k is the *coordination number*), so that each tensor has at most $\mathrm{O}(dD^k)$ coefficients, if D is the bond dimension of the TN. The overall scaling in system size N has now been reduced from the exponential scaling $\mathrm{O}(d^N)$ to a polynomial scaling $\mathrm{O}(NdD^k)$.
After contraction of the TN, the same number of coefficients is retrieved. However, they are not independent anymore as there has been added some structure through the TN. Even though they have no physical meaning, the bond indices are important in the way that through them, many-body entanglement is expressed (this was explained a little more in the introductory Chapter 1). There are two types of freedom in this description. Firstly, the topology of the TN influences the entanglement properties. As mentioned before, it is often logical for lattice systems to take the same topology for the TN. Secondly, the bond dimension has an impact on the entanglement properties as well. In the trivial case of a TN with bond dimension $D = 1$, no entanglement can be described. Therefore, the states are product states and we retrieve

[2]See also section 1.2 for an explanation of the bond indices in terms of maximally entangled pairs.

mean field theory. On the other hand, one could also increase the bond dimension so much (exponentially) that $\mathrm{O}(NdD^k) \geq \mathrm{O}(d^N)$. In this way, the TN representation is exact, but the advance in computational efficiency is then lost, of course. Understandably, the goal is to find, within the desired accuracy constraints, the description with the lowest possible bond dimension.

2.3 Matrix Product States

2.3.1 Definition

Given a one-dimensional system of N equivalent sites, a *Matrix Product State* or MPS $|\psi\rangle \in \mathcal{H} = \bigotimes_{i=1}^{N} \mathcal{H}_k$ is defined as

$$|\psi\rangle \triangleq \sum_{i_1,i_2,\ldots,i_N=1}^{d} \mathrm{tr}\left(A_{i_1}^{(1)} A_{i_2}^{(2)} \ldots A_{i_N}^{(N)}\right) |i_1 i_2 \ldots i_N\rangle \tag{2.10}$$

so that the coefficients $c_{i_1,i_2,\ldots,i_N}$ are given by

$$c_{i_1,i_2,\ldots,i_N} = \mathrm{tr}\left(A_{i_1}^{(1)} A_{i_2}^{(2)} \ldots A_{i_N}^{(N)}\right) = \sum_{\alpha,\beta,\ldots,\omega} A_{i_1;\alpha\beta}^{(1)} A_{i_2;\beta\gamma}^{(2)} \ldots A_{i_N;\omega\alpha}^{(N)} \tag{2.11}$$

where the summation goes over the appropriate bond dimensions to be discussed shortly. The physical indices are denoted by a subscript i_k, while the tensors are labelled according to their position via the superscript. The tensors A can then be interpreted as an array of d matrices with bond indices as their rows and columns; it is in this respect that the matrix product and trace are used in equations (2.10) and (2.11).
When dealing with MPS, sometimes the terminology *single layer* and *double layer* is used. By single layer, the MPS as defined in equation (2.10) is meant. The double layer of an MPS designates the bra-ket structure $\langle\psi|\psi\rangle$, used for example in the calculation of expectation values $\langle\psi|\hat{O}|\psi\rangle$, where the operator is inserted between the bra and the ket.

2.3.2 Boundary Conditions

In the general case, the bond dimensions of the different A's are not equal. $A_{i_k}^{(k)}$ is then a $D_{k-1} \times D_k$-matrix. For the first and the last site, two types of boundary conditions are possible, as drawn in figure (2.7).

- *Periodic Boundary Conditions (PBC)*: The first tensor/matrix $A_{i_1}^{(1)}$ is connected to the last one $A_{i_N}^{(N)}$ by a summation over α. Therefore $D_0 = D_N$. The linear chain is in this case actually a ring.
- *Open Boundary Conditions (OBC)*: If the index α is trivial, i.e. $D_0 = D_N = 1$, then the end matrices are vectors and therefore not connected to each other[3]. For finite systems, the question arises what determines the tensors at the boundary. In the thermodynamical limit, by contrast, we shall see in section 2.5 that (almost) all the effects of the boundary tensors are lost.

[3] Although the index α is superfluous for OBC, equations (2.10) and (2.11) are still correct.

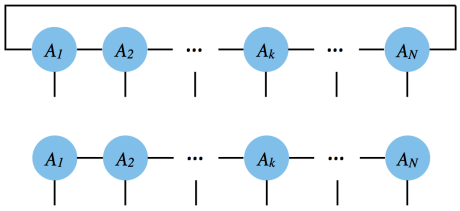


Figure 2.7: *Above:* MPS with periodic boundary conditions. *Below:* MPS with open boundary conditions.

2.3.3 Translational Invariance and Symmetry Breaking

Many interesting hamiltonians are *translationally invariant* (TI). We shall now analyse what the consequences are for MPS. A TI Hamiltonian with only nearest-neighbour interactions on a linear chain of N sites can be written as

$$\hat{H} = \sum_{i=1}^{N} \hat{h}_i \tag{2.12}$$

where $\text{supp}(\hat{h}_i)=\{i, i+1\}$ ($N+1 \equiv 1$ in the finite case with PBC[4]):

$$\hat{h}_i = \mathbb{1}_1 \otimes \mathbb{1}_2 \otimes \cdots \otimes \mathbb{1}_{i-1} \otimes \hat{h} \otimes \mathbb{1}_{i+2} \otimes \cdots \otimes \mathbb{1}_N \tag{2.13}$$

Defining the translation operator $\hat{\mathcal{T}}$ such that it acts on a basis state as

$$\hat{\mathcal{T}} \left| i_1 i_2 \cdots_N \right\rangle = \left| i_N i_1 \cdots_{N-1} \right\rangle \tag{2.14}$$

then

$$\hat{h}_i = \hat{\mathcal{T}} \hat{h}_{i-1} \hat{\mathcal{T}}^\dagger \tag{2.15}$$

such that indeed the Hamiltonian is TI:

$$\hat{H} = \hat{\mathcal{T}} \hat{H} \hat{\mathcal{T}}^\dagger \tag{2.16}$$

$$[\hat{H}, \hat{\mathcal{T}}] = 0 \tag{2.17}$$

The normalised ground state $|\psi_0\rangle$ defined as

$$|\psi_0\rangle \in \mathcal{H}, \| |\psi_0\rangle \| = 1 : \langle\psi_0| \hat{H} |\psi_0\rangle = \min_{|\psi\rangle \in \mathcal{H}} \frac{\langle\psi| \hat{H} |\psi\rangle}{\langle\psi|\psi\rangle} \tag{2.18}$$

is TI as well, i.e. $\hat{\mathcal{T}} |\psi_0\rangle = |\psi_0\rangle$. However, it may be a superposition of terms that lack translational invariance, or are only translationally invariant over multiple-site translations. This is just one example of the more general concept of symmetry breaking. Another instance of this universal concept is the ± 1 magnetisation in the ferromagnetic phase of the Ising model, discussed in section 4.2.6.

Without going into the group theoretical aspects of symmetry breaking, figure (2.8) shows a very simple case of symmetry breaking. Suppose that we have a TI Hamiltonian defined on a system of two-valued 'spins' (red/green), and that the first two states in the figure correspond to the lowest energy. Both states are only translationally invariant over even-site translations, but

[4] Note that in the finite case PBC are needed for TI.

by taking the symmetrical combination, a fully translationally invariant state is easily obtained. Returning to MPS, it is clear that an MPS constructed of site-independent tensors A is TI. The

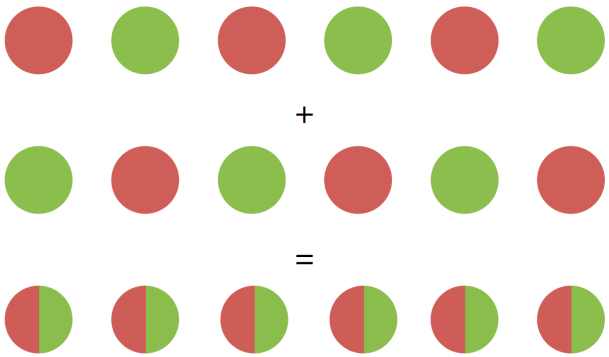


Figure 2.8: A simple instance of symmetry breaking states and the formation of a symmetric state as a combination of them.

question then remains whether a TI state can always be written with site-independent tensors. A proof that this is indeed possible, is given in [39].

2.3.4 Gauge Freedom

Finding a site-independent description is very much related to the concept of *gauge freedom*: an MPS is fixed when its constituent tensors are known, but the converse is not true. Because the bond dimensions are not physically relevant, gauge transformations that leave the physical indices undisturbed, don't alter the state. In practical calculations, the translation symmetry may be thus be impaired by site-dependent gauge transformations $X_{(k)}$:

$$\forall i_k : A^{(1)}_{i_1} A^{(2)}_{i_2} \dots A^{(N)}_{i_N} = \tilde{A}^{(1)}_{i_1} X_{(1)} X^{-1}_{(1)} \tilde{A}^{(2)}_{i_2} X_{(2)} X^{-1}_{(2)} \dots X_{(N-1)} X^{-1}_{(N-1)} \tilde{A}^{(N)}_{i_N} \tag{2.19}$$

In this equation, there is no gauge transformation between the first and the last sites because (i) in PBC there is still the freedom of a global gauge transformation, (ii) in OBC these are vectors.

2.3.5 Canonical Form

The gauge freedom can be employed to bring an MPS into a *canonical form*. The canonical form discussed in this section is applicable to infinite TI MPS. In section 2.2.2, the Schmidt decomposition was defined for $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$. The canonical form MPS corresponds to labelling each bond index according to the Schmidt basis. On each bond index, a matrix containing the Schmidt coefficients λ is placed, and as such the TN of tensors A is replaced by the canonical form $\{\Gamma, \lambda\}$ as shown in figure (2.9). Defining the right and left transfer matrices

$$(T_R)_{(\alpha\alpha')(\beta\beta')} \triangleq \sum_{i=1}^{d} (\Gamma_{i;\alpha\beta}\lambda_\beta)(\Gamma_{i;\alpha'\beta'}\lambda_{\beta'})^* \tag{2.20}$$

$$(T_L)_{(\alpha\alpha')(\beta\beta')} \triangleq \sum_{i=1}^{d} (\lambda_\alpha\Gamma_{i;\alpha\beta})(\lambda_{\alpha'}\Gamma_{i;\alpha'\beta'})^* \tag{2.21}$$

then the canonical form has to obey the following conditions

$$\sum_{\beta,\beta'} (T_R)_{(\alpha\alpha')(\beta\beta')}\delta_{\beta\beta'} = c\delta_{\alpha\alpha'} \tag{2.22}$$

$$\sum_{\alpha,\alpha'} \delta_{\alpha\alpha'}(T_L)_{(\alpha\alpha')(\beta\beta')} = c\delta_{\beta\beta'} \tag{2.23}$$

This means that the identity operator $\delta_{\alpha\alpha'}$ is a right eigenvector of T_R and a left eigenvector of T_L, with c the dominant eigenvalue, equal to unity to normalise the MPS. An algorithm that determines the canonical form starting from a given TI MPS, is given in section 3.1.3.
There are two advantages for using the canonical form. First, it makes calculating expectation

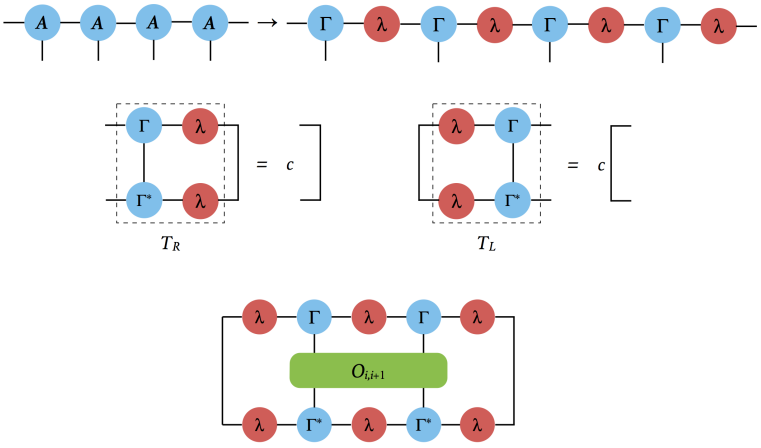


Figure 2.9: Top to bottom: (i) form of the canonical form, (ii) conditions on the canonical form, (iii) expectation value of a two-site operator using the canonical form.

values for local operators computationally easier. As an example, on the bottom of figure (2.9) the expectation value of a two-site operator is shown. Secondly, it offers a natural and physically justified way to truncate the bond dimension (see section 3.1.4).
Although in the applications of Chapters 4 through 6 the canonical form is not explicitly used, these two elements are incorporated in the algorithms. The former advantage is used in the one-dimensional single layer contraction algorithm of section 3.1.2, while the latter is used extensively in the two-dimensional corner method of section 3.2.

2.3.6 Matrix Product Operators

One can circumvent the restriction of dealing only with pure states by generalising the concept of MPS to an *Matrix Product Operator* or MPO $\hat{O}$:

$$\hat{O} \triangleq \sum_{i_1,i_2,\ldots,i_N=1}^{\tilde{d}} \mathrm{tr}\left(A^{(1)}_{i_1} A^{(2)}_{i_2} \ldots A^{(N)}_{i_N}\right) \hat{o}_{i_1} \otimes \hat{o}_{i_2} \otimes \cdots \otimes \hat{o}_{i_N} \tag{2.24}$$

where the operators $\{\hat{o}_i, i = 1, \ldots, \tilde{d}\}$ form a single particle basis (such as the Pauli matrices for spin-1/2 particles[5]). Graphically, an MPO is depicted in the same way as an MPS, but with an extra set of physical indices. Rather than using MPOs to describe mixed states, we shall need

[5]Note that for a given MPS with physical dimension d, the MPO typically has $\tilde{d} = d^2$. Indeed, for a spin-1/2 particle, there are three Pauli matrices and the identity matrix.

them as transfer matrices (of infinite dimensions) working on an MPS as displayed in figure (2.10)[6]. This is instrumental in the quantum-classical mapping described in section 2.6.

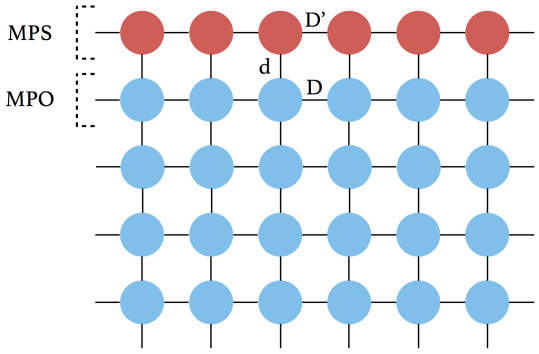


Figure 2.10: MPOs working on an MPS.

2.4 Projected Entangled Pair States

The natural generalisation of MPS to two dimensions, are *Projected Entangled Pair States* or PEPS. One can write down a definition analogous to the definition of MPS in equation (2.10), but it should be clear that certainly in the case of TN, a picture is often worth a thousand words.
Therefore, a PEPS is shown on the left side of figure (2.11). In general, the tensors at different lattices may be different, but as in the one-dimensional case, we will only work with TI descriptions. On the right side of figure (2.11), the generalisation of MPOs is shown.
As for MPS, OBC and PBC can be defined for PEPS (the boundary conditions are left undetermined in the figure). Just as a linear MPS in OBC becomes a ring in PBC, a PEPS in PBC has the shape of a torus.

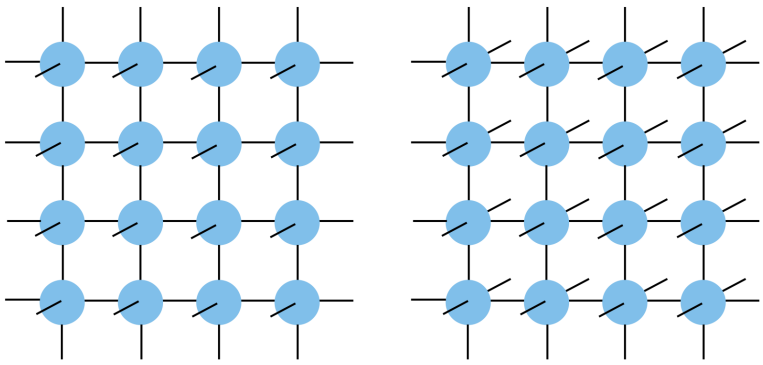


Figure 2.11: *Left:* PEPS. *Right:* PEPS operator

[6]So that the 'row' index $\tilde{d}$ is now equal to d, like in the figure.

2.5 Transfer Matrix and Power Methods

2.5.1 Thermodynamical Limit and Transfer Matrix

In the applications taking up Chapters 4 through 6, we shall always work in the thermodynamical limit, namely a number of lattice sites $N \to \infty$. Given a TI TN, be it in one, two or three dimensions, the first goal will be to contract it. Let us take the example of a two-dimensional TN without open indices, i.e. a PEPS without physical indices. Let us say that we want to contract in the vertical dimension first. The TN can then be thought of as consisting of an infinite number of copies of the same MPO and this contraction then amounts to finding the fixpoint of that MPO. This fixpoint is the *dominant* eigenvector of the MPO. The dominant eigenvector denotes the eigenvector corresponding to the dominant eigenvalue (supposing it is non-degenerate), which is the eigenvalue largest in absolute value. In the language of TN, this fixpoint takes the form of an MPS.
The same basic concept of fixpoint applies to the one- and three-dimensional case as well. Also, in section 3.2, a corner method is described that contracts a two-dimensional TN in a diagonal way. Even though the TN can then not be thought of as the repeated application of an MPO, the same basic concept of finding fixpoints still applies.
In order to determine the relevant fixpoints, we turn to power methods. The basic form of a power method, power iteration, allows to find the dominant eigenvector of a matrix A. To accomplish this, one should start with an initial guess x_0. Through repeated application of A, $x_i = A^i x_0$ will converge to the dominant eigenvector, as long as it was not perpendicular to the dominant eigenvector[7]. The mathematical theorem that proves the efficacy of the power method is given in the next section. In higher dimensions, additional steps such as bond truncation need to be added, but are in essence very similar to power iteration. The precise details are explained in Chapter 3.

2.5.2 Subspace Iteration

The mathematical result supporting power iteration is commonly referred to as *subspace iteration*. The theorem and proof [7] are given here in full, for two reasons. Firstly because of the importance of this theorem, underlying most of the methods of Chapter 3, and secondly because the proof gives information on the convergence error.

Theorem. *Given $A \in \mathbb{K}^{n\times n}$ with spectrum $\{\lambda_i, i = 1, \dots, n\}$ such that*

$$|\lambda_1| \geq \cdots \geq |\lambda_m| > |\lambda_{m+1}| \geq \cdots \geq |\lambda_n| \tag{2.25}$$

and $\{x_i \in \mathbb{K}^{n\times 1}, i = 1, \dots, m\}$ a set of linearly independent vectors (with an extra condition arising in the course of the proof). If $X_0 = [x_1, \dots, x_m]$, then under the iteration $X_k = AX_{k-1}$, $\mathbb{K}X_k$ will converge to sum of the generalised eigenspaces $\bigoplus_{|\lambda_j|\geq|\lambda_m|} M^{\lambda_j}$.

Proof. Defining the invariant subspaces $M_1 = \bigoplus_{|\lambda_j|\geq|\lambda_m|} M^{\lambda_j}$ and $M_2 = \bigoplus_{|\lambda_j|<|\lambda_m|} M^{\lambda_j}$, we choose a Jordan basis in each of them. With these vectors, we construct the matrices $V_1 \in \mathbb{K}^{n\times m}$

[7] As a result of numerical inaccuracies, this condition is relaxed in numerical simulations.

and $V_2 \in \mathbb{K}^{n\times(n-m)}$. A can then be brought into the Jordan canonical form by $[V_1, V_2]^{-1}A[V_1, V_2] = J$. As $J = \begin{bmatrix} J_1 & 0 \\ 0 & J_2 \end{bmatrix}$, $AV_i = V_iJ_i$.
The vectors x_i can be uniquely decomposed as a linear combination of the Jordan basis vectors of V_1 and V_2: $X_0 = V_1C_1 + V_2C_2$, with $C_1 \in \mathbb{K}^{m\times m}$ and $C_2 \in \mathbb{K}^{(n-m)\times m}$. As the rank of X_0 is m, C_1 is generally non-singular, but in the case it is singular, we can change X_0 (this is the condition on X_0 hinted at in the theorem). It follows that

$$X_k = A^kX_0 = A^kV_1C_1 + A^kV_2C_2 = V_1J_1^kC_1 + V_2J_2^kC_2 = (V_1 + E_k)J_1^kC_1$$

with $E_k = V_2J_2^kC_2C_1^{-1}J_1^{-k}$. Given that the spectral radii are $\rho(J_1^{-1}) = |\lambda_m|^{-1}$ and $\rho(J_2) = |\lambda_{m+1}|$, we get that $\|J_1^{-k}\| \leq \varepsilon_1|\lambda_m|^{-k}$ and $\|J_2^k\| \leq \varepsilon_2|\lambda_{m+1}|^k$ for sufficient k.
Hence,

$$\|E_k\| \leq \varepsilon \left|\frac{\lambda_{m+1}}{\lambda_m}\right|^k$$

such that $\lim_{k\to\infty}\|E_k\| = 0$.
As Im(X_k)=Im($V_1 + E_k$), we get that for sufficient k Im(X_k)≈Im(V_1)=M_1. □

For the purposes of this dissertation, $m = 1$. This means that the convergence rate depends on the ratio $|\frac{\lambda_2}{\lambda_1}|$. The matrix C_1 is then just a scalar and the condition that it be non-singular means that C_1 must not be zero. Hence, the condition on X_0 is that it have a component along V_1. In practical circumstances however, this condition can be relaxed due to numerical inaccuracies which supply X_k with a component along V_1.

2.6 From Quantum to Classical Problems

So far, TN and related concepts have been introduced, but it has remained rather vague what we want to do with them. It has been mentioned that the first step is to contract the network. Indeed, in this section, it is shown how the contraction of a TN leads to the partition function of a canonical[8] ensemble.
We shall first have a look at general quantum-classical mapping. Let us consider a quantummechanical system (quantum field theory in 0 dimensions), with Hamiltonian $\hat{H}_q$. The transition amplitude between an initial state $|i\rangle$ and a final state $|f\rangle$ is given by $\langle f|\, e^{-it\hat{H}_q}\, |i\rangle$. By performing a Wick rotation $\tau = it$ in the complex plane, we arrive at imaginary time evolution. The transition amplitude can now be expanded as an imaginary-time path integral[9]

$$\langle f|\, e^{-\tau\hat{H}_q}\, |i\rangle = \sum_{j_1,\ldots,j_N} \langle f|\, e^{-\delta\tau\hat{H}_q}\, |j_1\rangle \langle j_1| \ldots |j_N\rangle \langle j_N|\, e^{-\delta\tau\hat{H}_q}\, |i\rangle \tag{2.26}$$

where $N\delta\tau = \tau$ and $\delta\tau$ should be small compared to τ. On the other hand, for a classical system at inverse temperature β and with Hamiltonian H_c, the partition function $\mathcal{Z}$ is defined as

$$\mathcal{Z} = \mathrm{tr}(e^{-\beta H_c}) \tag{2.27}$$

[8] This use of the word 'canonical' bears no relation to the canonical form, of course. It means that the physical system is studied at fixed temperature, volume and number of particles (the latter two being infinite in our case).
[9] We write discrete equations as the classical systems of interest in this dissertation are defined on a lattice.

For a one-dimensional system of spins on a linear chain with nearest-neighbour interaction, one can define a transfer matrix T between two adjacent sites (this is explained in greater detail in Chapter 4). With this transfer matrix, the partition function can be rewritten as

$$\mathcal{Z} = \sum_{s_2,\ldots,s_{N-1}} \langle s_1| T |s_2\rangle \langle s_2| \ldots |s_{N-1}\rangle \langle s_{N-1}| T |s_N\rangle \tag{2.28}$$

where $\{s_i\}$ denote the spins and s_1 and s_N are fixed (OBC). In PBC, an extra T should be inserted between the first and the last site, and the sum over their spins should also be taken.
It is clear that with the proper renaming between equations (2.26) and (2.28), the 0D quantum case corresponds to the 1D classical case with transfer matrix $T = e^{-\delta\tau \hat{H}_q}$.
The same procedure can be followed in higher dimension, such that in general a d-dimensional quantum system corresponds to a $(d+1)$-dimensional classical system.

Returning to TN, let us take the example of a 1D quantum state, i.e. an MPS. The imaginary-time evolution similar to equation (2.26) corresponds to the repeated application of an MPO. In the manner we just explained, this corresponds to calculating the partition function of a classical system. So indeed, the partition function is obtained when contracting the TN constructed by joining the nearest neighbours in a lattice by the appropriate transfer matrix. More explicit details on how this is achieved, can be found in section 5.1, where we shall also discuss how one can calculate the expectation value of operators.

Chapter 3

Computational Methods

In this chapter, several algorithms are discussed, mainly for contraction of TI networks in 1D, 2D and 3D. These algorithms are always for the thermodynamical limit. Therefore, without exception their core is some kind of a power method. The focus of this chapter lies on the concepts and convergence properties of the algorithms, while in chapters 5 and 6, the classical Ising model is discussed as an application of these algorithms.
Most generally, these methods work for complex tensors. However, for applications in statistical physics, real numbers are sometimes sufficient[1]. For that reason, the complex conjugate in the bra side is not always explicitly written. However, even in the real case complex terminology (unitarity, Hermitian conjugate) is used instead of real terminology (orthogonality, transpose). In addition, the field over which the tensors are defined is, where necessary, denoted by $\mathbb{K}$ which can stand for either $\mathbb{C}$ or $\mathbb{R}$.

3.1 One-Dimensional Methods

In this section, a few one-dimensional algorithms are discussed. They will not be used for the applications of the next chapters, but are nonetheless given here for two reasons. Firstly, the more involved algorithms of the next sections draw on many of the concepts which appear clearly in these simpler algorithms. Secondly, due to their relative simplicity, it was with these algorithms that I started my explorations into the world of TN.

3.1.1 Determining the Fixpoint of an MPS - Double Layer

We first discuss two algorithms to calculate the left or right fixpoint of a double-layer MPS with tensors A. These fixpoints are needed, for instance, in the calculation of expectation values of local operators. The simplest method to achieve this is a power iteration in the double layer, a scheme of which is shown in figure (3.1). Defining the *transfer operator* E as

$$E_{\alpha\alpha',\beta\beta'} = \sum_{i=1}^{d} A_{i;\alpha,\beta} A^*_{i;\alpha',\beta'} \tag{3.1}$$

[1] Though not always, see section 3.3.1.

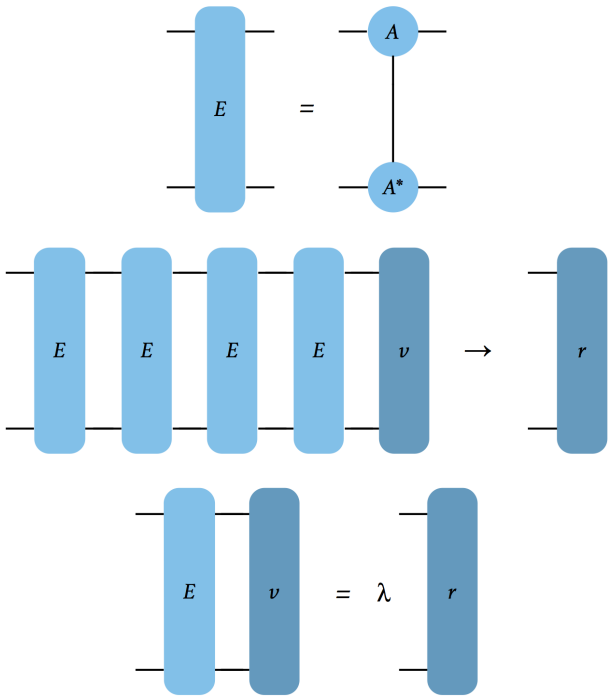


Figure 3.1: Double-layer algorithm for the contraction of a double-layer MPS.

determining the left and right fixpoints equates to finding the left and right dominant eigenvectors l and r of E. Starting with a vector v as an initial guess, the right eigenvector can be found by repeated multiplication of E on the left with v until convergence, whereby the new v is normalised in each iteration step to avoid a build-up of a factor λ^N with N the iteration step and λ a short-hand notation for the subsequent approximations of the dominant eigenvalue. It was mentioned in section 2.5.2 that a necessary condition for this method to work, is that $v \notin \mathbb{K}\{u_2, \ldots, u_n\}$, or again $v^\dagger u_1 \neq 0$, where $\{u_1, u_2, \ldots, u_n\}$ are all the eigenvectors, with u_1 the dominant one. Therefore, a good ansatz is to take a random vector as the initial guess. It should be noted in this respect that, due to numerical inaccuracies, even when the projection of the initial guess on the dominant eigenvector is zero, the right convergence can be obtained.
Let us now have a look at the scaling of this algorithm. The dimensions of the transfer operator are $D^2 \times D^2$ so that each multiplication with E requires $O(D^4)$ operations. However, it is computationally advantageous to use the bra-ket structure of E and add one A at a time. This corresponds to the way of contraction shown in the right half of figure (2.5). The number of operations needed in that case scales as $O(dD^3)$.

3.1.2 Determining the Fixpoint of an MPS - Single Layer

When working in the single layer picture on the other hand, another approach is possible. The proposed method is still essentially a power method, but is not anymore the straightforward construction of the dominant eigenvector of a matrix. This is because in the single layer, all the physical indices are left dangling. Yet through a proper choice of the gauge, one can avoid having to store all the results of the previous iterations steps.
To show this, let us look at an iteration from the left again. If the tensors of the MPS ket were all left-unitary ($Q^\dagger Q = \mathbb{1}$, but not necessarily $QQ^\dagger = \mathbb{1}$), contraction from the left with their bra counterpart would result in unity, as shown in figure (3.2).
A way to mould the given TI MPS into that form is by repeated QR-decomposition as drawn

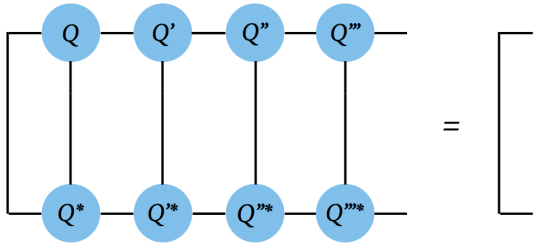


Figure 3.2: Contraction property of left-unitary tensors.

in figure (3.3). For the QR-decomposition $A = QR$, the physical index is considered part of the row index of A. For the QR-decomposition of square matrices, R is an upper-triangular matrix. In this case where the number of rows exceeds the number of columns, we get that

$$A = QR = \begin{bmatrix} Q1 & Q2 \end{bmatrix} \begin{bmatrix} R1 \\ 0 \end{bmatrix} = Q_1 R_1 \tag{3.2}$$

where $R1$ is an upper-triangular matrix. Mathematically, this economical form, sometimes called the *thin* QR-decomposition is not mandatory, but in view of the structure of the MPS, it is. Keeping the full indices would result in a meaningless increase in the bond dimension. The consequence is that Q_1 is not unitary anymore: its columns are still orthogonal, but its rows not necessarily so. Therefore, Q_1 is left-unitary, which satisfies our demands.
After its QR-decomposition, A is replaced by Q, and R is added to the A of the next site. Again,

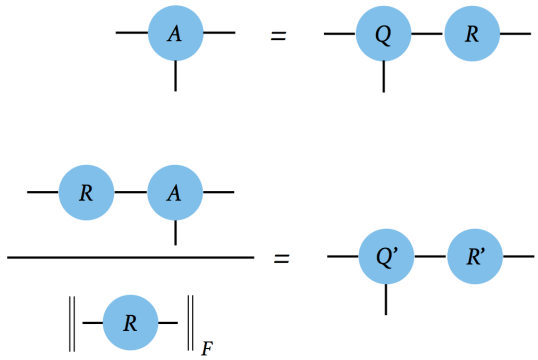


Figure 3.3: *Above:* Thin QR-decomposition of A. *Below:* Iteration step in the single-layer algorithm to find the fixpoints of an MPS using QR-decomposition.

to avoid a divergence of numerical factors, R is normalised in every iteration step. Yet, there is still some freedom in this gauge: if A is full rank in its columns, the thin QR-decomposition is unique up to the phase of the diagonal elements of R_1. Therefore, this phase is fixed in the algorithm to zero such that R_1 has real diagonal elements.
In Appendix A, an estimate for the number of operations required for thin QR-decomposition is given. It turns out that this requires $O(dD^3)$, and the multiplication RA also scales as $O(dD^3)$ (the multiplication of R with a diagonal phase matrix only scales as $O(D^2)$; the once-only contraction of the final R with R^* scales as $O(D^3)$). Therefore, the overall complexity is $O(dD^3)$. Consequently, this algorithm has the same scaling as the double-layer algorithm. Yet there is one big advantage to work in the single layer: it preserves the bra-ket structure of the fixpoint, i.e. it ensures that the fixpoints are Hermitian and non-negative when considered as a matrix. This is not a priori so in the double-layer algorithm.

3.1.3 Bringing an MPS into Canonical Form

The algorithm discussed in the previous section incorporates some elements of the canonical form as presented in section 2.3.5. In this section, we shall examine an algorithm [35] that really constructs the canonical form of an MPS:

i. Construct the dominant left and right eigenvectors v_L and v_R of the transfer operator E and let us call the dominant eigenvalue μ.

ii. Due to the special bra-ket structure of the transfer operator E, v_L and v_R are hermitian and non-negative. Hence, they can be decomposed as $v_R = XX^\dagger$ and $v_L = Y^\dagger Y$. A possible way to accomplish this, is by calculating the eigenvalue decomposition $v_R = U\Lambda U^\dagger$ and taking $X = U\sqrt{\Lambda}$.

iii. Place the identities XX^{-1} and $(Y^T)^{-1}Y^T$ to the left and right of the tensor A. Calculate the SVD of $Y^T X = U\lambda V^\dagger$.

iv. The canonical form $\{\Gamma, \lambda\}$ is given by (1) $\Gamma_i = V^\dagger X^{-1} A_i (Y^T)^{-1} U$ (the subscript i stands for the physical index) and (2) λ the Schmidt coefficients as found in the previous step.

This procedure is outlined in figure (3.4). It can easily be checked by direct substitution that this $\{\Gamma, \lambda\}$ fulfils the conditions of the canonical form of section 2.3.5.

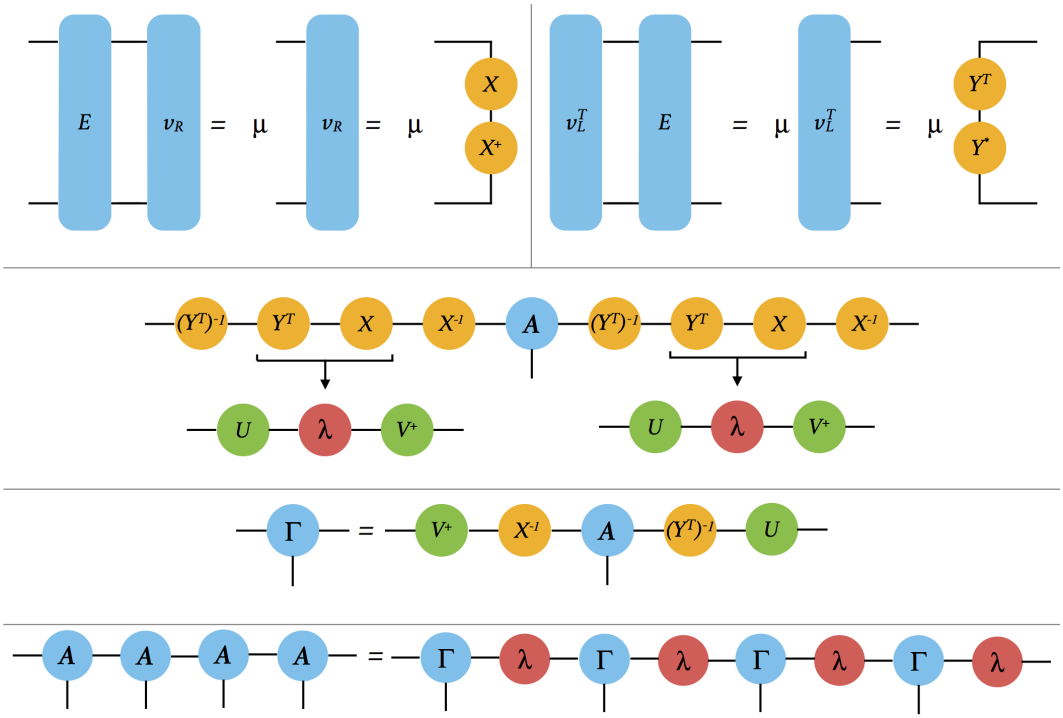


Figure 3.4: Algorithm to bring a TI MPS into canonical form $\{\Gamma, \lambda\}$.

3.1.4 Working with an MPO on an MPS

The canonical form offers a first possibility to contract a two-dimensional double-layer TN consisting of tensors T of rank four. To see how this comes about, the contraction should be split up into a horizontal and a vertical one. Let us assume that we start with the vertical

contraction[2] from top to bottom. This boils down to finding the fixpoint of a horizontal layer of the TN. Since such a layer is nothing else but an MPO, and the environment is an MPS, the crucial part of the contraction is the update of an MPS after an MPO has worked on it. When the MPS is in canonical form, this can be done in an efficient way.
Since the bond dimensions of the MPS increase after an MPO is applied to it, truncation will be necessary. Otherwise, it would be computationally not feasible to apply a whole series of MPO. The explicit presence of the Schmidt coefficients in the canonical form, is what makes it suitable for this purpose.
There are three steps to the algorithm [35], as summarised in figure (3.5):

i. *Update:* After contraction with a horizontal layer of tensors T, $\{\Gamma, \lambda\}$ is replaced by $\{\dot{\Gamma}, \dot{\lambda}\}$. $\dot{\Gamma}$ is given by

$$\dot{\Gamma}_{j;\dot{\alpha}\dot{\beta}} = \sum_{i=1}^{d} \Gamma_{i;\alpha\beta} T_{ij;\mu\nu} \tag{3.3}$$

with $\dot{\alpha} = (\alpha, \mu)$ and $\dot{\beta} = (\beta, \nu)$. λ on the other hand is not influenced by the MPO such that

$$\dot{\lambda}_{\dot{\alpha}} = \lambda_{\alpha}, \forall \mu \tag{3.4}$$

where $\dot{\lambda}$ and λ only have one index because they are diagonal.
If the bond dimensions of $\{\Gamma, \lambda\}$ is D, and the index dimensions of T are d[3], then the new bond dimension is dD.

ii. *Writing in the canonical form:* The updated MPS $\{\dot{\Gamma}, \dot{\lambda}\}$ is brought into canonical form $\{\ddot{\Gamma}, \ddot{\lambda}\}$. In the algorithm for constructing the canonical form, the old MPS now contains a λ-matrix as well. To incorporate this, in step (*iii*) the SVD should be calculated of $Y^T \lambda X$ rather than of $Y^T X$. Additionally, the transfer operator E should be adapted such that (1) for the right eigenvector it is composed of $A = \Gamma\lambda$ (and its bra version), and (2) for the left eigenvector it is composed of $A = \lambda\Gamma$ (and its bra version).

iii. *Truncation:* By keeping only the $D' < dD$ largest Schmidt coefficients, all bond indices are truncated and the end result is an approximate MPS $\{\Gamma', \lambda'\}$[4]. The characterisation of D' can vary. A first choice is to keep the bond dimension equal in every iteration step, i.e. $D' = D$. Secondly, D' could follow a fixed prescription, such as a linear increase with the iteration step. Lastly, D' could be determined dynamically such that in every step all the Schmidt coefficients smaller than a certain cut-off value are discarded.

The same procedure is repeated for the upwards contraction. The result of the vertical contractions is therefore a ket-MPS and a bra-MPS. A priori, these may be different, so that the single-layer algorithm for one-dimensional contraction does not apply. For that reason, in the general case a double-layer contraction algorithm needs to be used. In the presence of symmetry, the two fixpoints can be equal, in which case a single-layer contraction is possible.

[2] To avoid all ambiguity, vertical contraction is understood such that the result is a horizontal chain.
[3] The vertical and horizontal index dimension may vary, but are always equal in the applications discussed in this work.
[4] Again, to avoid ambiguity, 'approximate' does not mean that $\{\Gamma', \lambda'\}$ is not a real MPS. If the MPS originally describes $|\psi\rangle$, the new MPS should describe a new state $|\psi'\rangle$. $\{\Gamma', \lambda'\}$ is approximate in the sense that it only approximately corresponds to $|\psi'\rangle$ due to the truncation step.

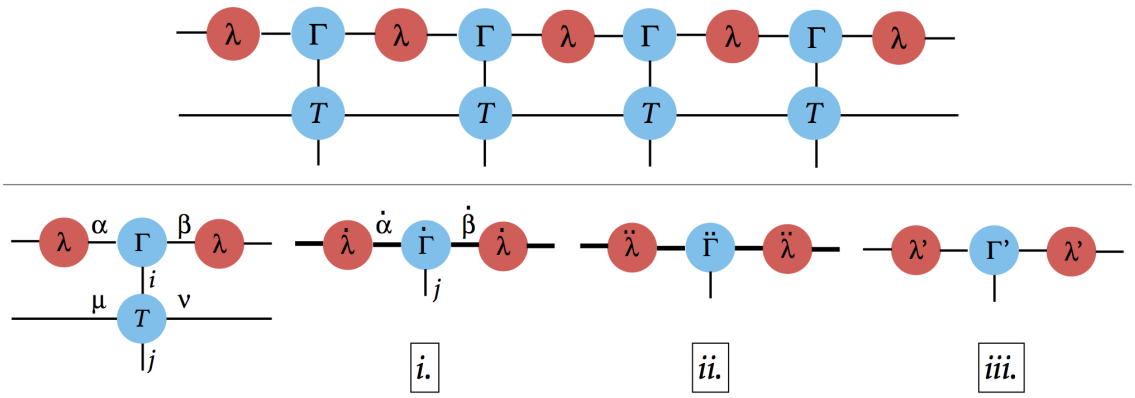


Figure 3.5: Working with an MPO on an MPS.

3.2 Corner Method for Network Contraction in 2D

In addition to a straight-forward contraction of a two-dimensional tensor network as a repeated application of the same MPO, other methods can be constructed that explicitly take the two dimensions together. Such methods can be found in the literature [33] [36], and a variant is presented here. Again, we work with a 2D TN without physical indices, which means it could either be the double layer version of a PEPS, or a classical partition function (an application of which will take up chapter 5).

3.2.1 Basic Idea

We start with an infinite two-dimensional tensor network of tensors T. The basic idea is to add, in every iteration step, not a flat layer, but a layer with a right angle in it (⅃-shaped). As shown in figure (3.6), the environment is described by horizontal tensors A and vertical tensors B, which may of course be equal in the presence of symmetries. The same environment is then placed on the other side of the added layer. Since the lattice size is increased by adding a layer of Ts, an A and B are missing at the angle, depicted by M in the figure. A new A, B and S are then constructed, and this procedure is repeated until convergence. Of course, if the solution has not converged yet, the different As and Bs will not be the same. After convergence, it is still not trivial that the As and Bs should have converged themselves, but that is in fact so, can be seen in section 3.2.3 (hence the fact that they are drawn equally in the figure). In the next section, a more detailed account of the method is given.

3.2.2 Details of the Method

In the following description of the algorithm, reference is made to figure (3.7).

Initialisation

Suppose the dimension of T is d^4. A random d^3-tensor is chosen as initial environments L and R. Since the proposed method is a power method, we expect that the precise choice of the environment does not matter as long as its projection on the dominant eigenvector with largest eigenvalue is non-zero (which is, as mentioned before in section 2.5.2, not strictly required due

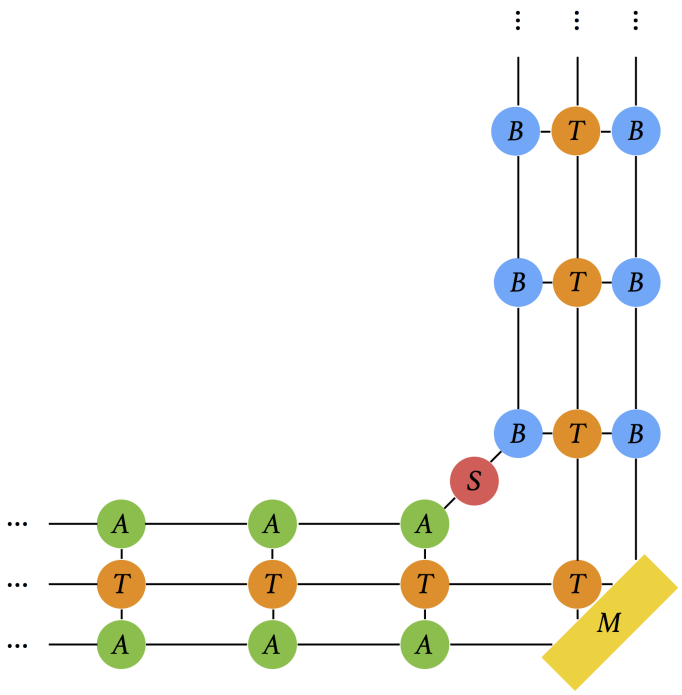


Figure 3.6: One update step in the corner method

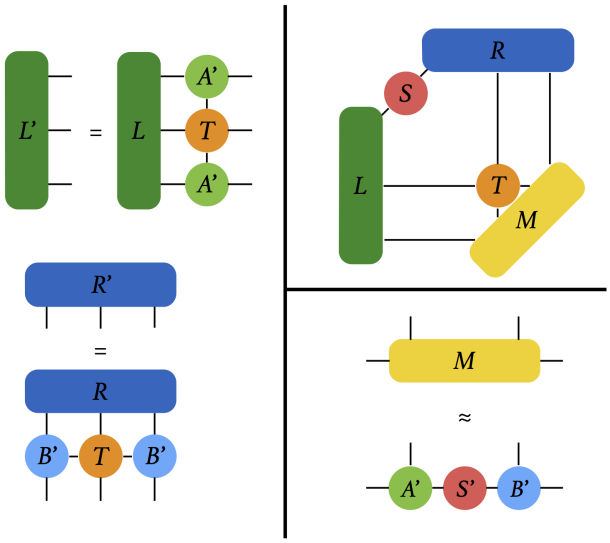


Figure 3.7: Some important steps in the corner method.

to numerical inaccuracies).
And yet, the initial choice of the environment can still have an influence on the outcome. An example of this is the Ising model, where the choice of the environment determines the symmetry breaking in the ferromagnetic phase (see section 4.2.6).
For S, the initial guess is just the identity matrix.

Construction of M

Let us first look at the meaning of placing the environment at the other side of the T-layer in figure (3.6). This can be interpreted as an overlap between (1) the old environment updated with a layer of Ts and (2) the new environment, where convergence is anticipated by using the same tensors for the new environment as for the old environment, except at the angle. The

required value for the overlap is one, of course, when all the tensors are properly normalised. Therefore, tensors L, S, R and T are contracted and interpreted as a $d^2 \times d^2$-matrix. The overlap with M is maximised by taking it equal to this matrix. M is then normalised to avoid divergence.

Construction of a new A, B and S by truncation

An SVD of M gives $M = U\Sigma V^\dagger = \tilde{A}'\tilde{S}'\tilde{B}'$, where all matrices are $d^2 \times d^2$. Truncation is achieved by selecting only the biggest singular values and the corresponding columns of $\tilde{A}'$ and $\tilde{B}'$, which gives us S', A' and B'. Let us call the new bond dimension D.
The new environments are computed by adding an $A' - T - A'$ and $B' - T - B'$ layer to L and R, respectively, making the new dimensions d^2D.

Convergence criterion

Since the property that the old environment contracted with a T-layer should equal the new environment, is used in the algorithm, a new convergence criterion is needed. To that end, the overlap of the old and the new environment are calculated without the intermediary T-layer, as shown in figure (3.8). Even though the old environment is one site shorter than the new environment, this does not matter, since, again, the overlap is closed on the left by a random environment.

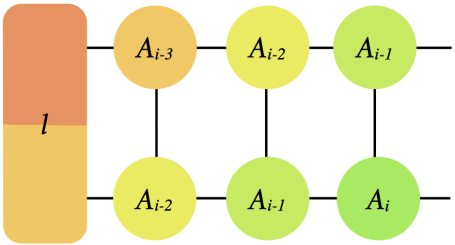


Figure 3.8: Overlap matrix for the As.

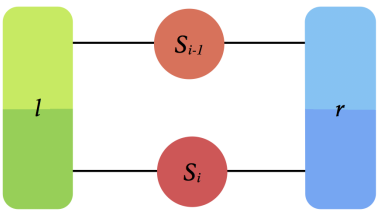


Figure 3.9: Overlap between the old and new environment used for convergence.

The same is done for the Bs, and the final overlap between the old and new environment is shown in figure (3.9). Denoting the overlap by x, the error is defined as $(x-1)^2$. A typical example of the convergence is shown in figure (3.10), where x is shown as a function of the iteration step.

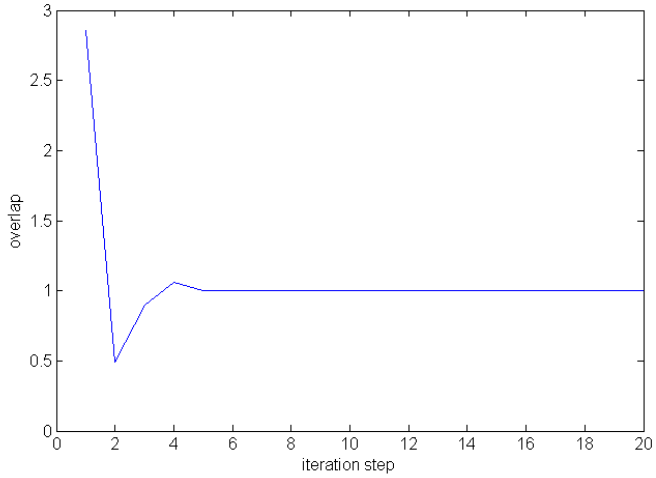


Figure 3.10: Typical convergence behaviour of the corner method.

3.2.3 Convergence of A and B

After convergence of this method, it is desirable that the tensors A and B converge as well. A priori, it is not clear, however, that this should be the case, since the repeated SVD may induce site-independent gauge transformations.
To test whether this is the case, one can calculate the overlap of the old A (or B) with the new A (or B). Since the bond dimensions are a priori not equal in different iteration steps[5], we hereby truncate the biggest bond dimension to match the smaller one. Because the As and Bs originate from an SVD, this truncation corresponds to throwing away extra Schmidt vectors corresponding to the smallest Schmidt coefficients. As such, this truncation is relatively safe when checking the convergence of A and B. Figure (3.11) shows an example of the convergence of A, for the two-dimensional Ising model around the critical temperature (see Chapter 5).

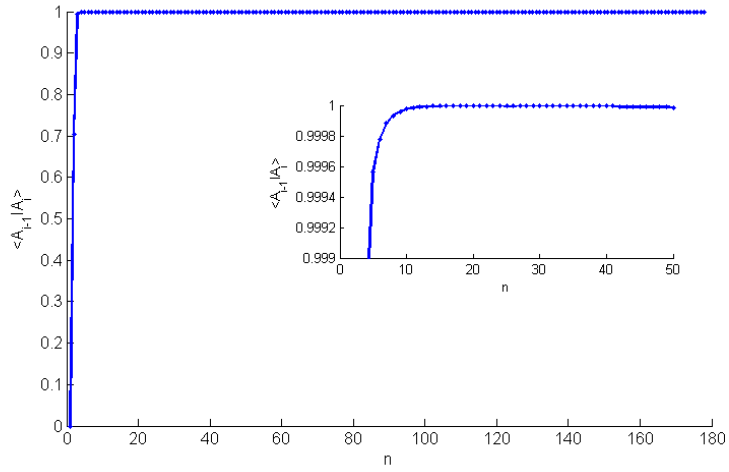


Figure 3.11: Convergence of the tensor A in the corner method. The inset shows a magnified version of the bigger curve.

[5] Although, at least away from critical points, the bond dimension was found to converge.

3.2.4 Expectation Values

The next step is to calculate expectation values, for which this method is used to calculate the proper environments. In the most general case, the corner method needs to be applied four times, as shown in figure (3.12) for a one-site operator O. Afterwards, a one-dimensional contraction method, such as the ones proposed in section 3.1, is needed to find an environment for the four arms of the cross.
However, if the tensors T are left-right and (or) up-down symmetric, only one (two) environ-

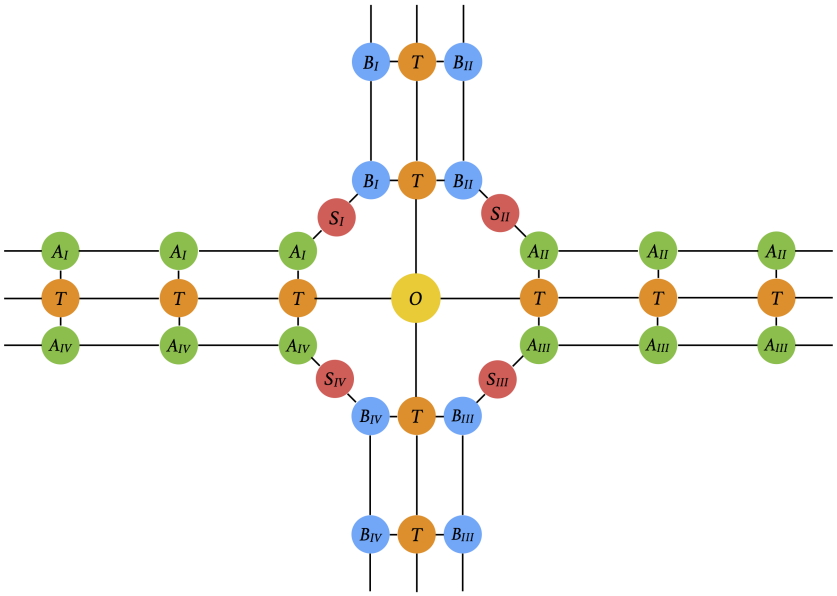


Figure 3.12: Expectation value of a one-site operator using the corner method.

ment(s) need to be calculated. Suppose both symmetries are present, then it is clear that no extra one-dimensional method is needed, as the required environments are generated in the corner method.
The generalisation to multi-site operators is straightforward. For the two-site operators used in the Ising model, the modified environments were also calculated directly in the corner method instead of applying a one-dimensional method afterwards.

3.3 Network Contraction in 3D

Lastly, we shall have a look at the contraction of a three-dimensional TN of tensors T of rank six. Similarly to the two-dimensional MPS/MPO algorithm in section 3.1.4, we can look at the three-dimensional case as follows. The contraction over the first index is performed by finding the two-dimensional environment (a PEPS) which corresponds to the eigenvector with the largest eigenvalue of the transfer matrix consisting of a two-dimensional layer of tensors T. For the remaining contractions, the two-dimensional corner method mentioned above (or another method) can be used. The main conceptual difference to the two-dimensional case is that, due to the absence of a suitable generalisation of singular value decomposition in higher dimensions, a different approach to truncation is needed here.

3.3.1 Method

A schematic overview of some key steps is shown in figure (3.13).

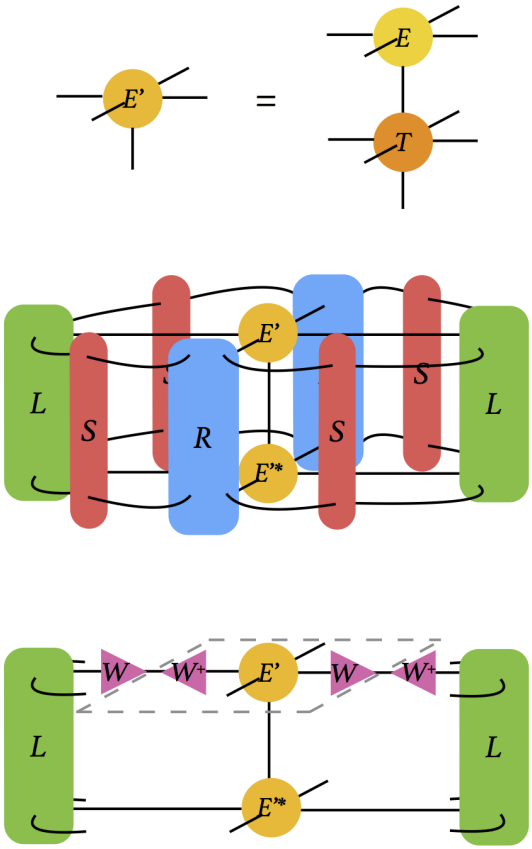


Figure 3.13: Schematic overview of the contraction algorithm in 3D. (In the bottom network, not all the tensors are drawn so as not to overload the picture.)

Update

We start with an initial guess for the environment, namely a TI PEPS of tensors E. By contracting E with T (above in figure (3.13)), we get the new environment tensor E'. A two-dimensional contraction is then performed on the double layer version of E'. The result is shown in the middle of figure (3.13).

Truncation (1): concept

Due to the absence of a suitable SVD in higher dimensions, an alternative method of truncation is needed. To that end, an orthogonal projection matrix $P = WW^\dagger$ is inserted at the four bond indices of the ket E, with a distinction between the horizontal W_h and the vertical W_v (i.e., horizontal and vertical in the plane). This is shown in the bottom part of figure (3.13), but only for the horizontal W_h to keep the figure simple.
Hence, the truncated environment tensor is E with a matrix W on its four bond indices. The second W matrix that is placed on every bond index belongs to the truncated E of the neighbouring sites. If the original overlap is normalised to unity, the goal is to find Ws such that the

new overlap approaches unity.

Since idempotence is a necessary and sufficient condition for a matrix (endomorphism) to be a projection, it is clear that P is indeed a projection if W is left-unitary, i.e. $W^\dagger W = \mathbb{1}_{D\times D}$ (we shall further discuss the matrices W in a minute). Let us denote the D-dimensional vector space on which P acts, by V. P then accomplishes a projection onto the $\tilde{D}$-dimensional subspace $\mathrm{Im}(P) \prec V$ along the subspace $\mathrm{N}(P) \prec V$, with $V = \mathrm{Im}(P) \oplus \mathrm{N}(P)$.
In this case, P is an orthogonal projection. A necessary and sufficient condition is that P be self-adjoint (as an endomorphism) or hermitian (as a matrix). Due to the special decomposition into W and $W^\dagger$, this is indeed the case. We can therefore write $U = \mathrm{Im}(P)$, $U^\perp = \mathrm{N}(P)$, and $V = U \oplus_\perp U^\perp$.

As we just mentioned, left-unitarity of W is a sufficient condition for P to be a projection. When thinking of W as a transformation $W : U \to V$, it is an isometry. This means that $\forall u_1, u_2 \in U$

$$\langle Wu_1, Wu_2\rangle_V = \langle u_1, u_2\rangle_U \tag{3.5}$$

As W is left-unitary, this holds indeed[6].

Truncation (2): iteration

The question remains how the matrices W should be constructed. Starting from an initial left-unitary guess (e.g. an identity matrix supplemented with rows of zeros), they are converged in an iteration loop. As all the horizontal isometries are equal, as are all the vertical ones[7], only two isometries need to be updated in every iteration step. In every step, first the horizontal isometry is updated while keeping all the others fixed and then the same is repeated for the vertical isometry (or vice versa, of course).

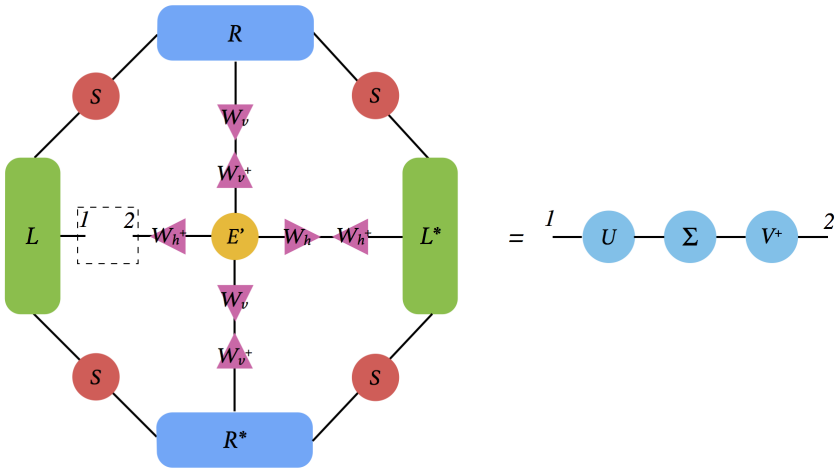


Figure 3.14: TN for the update of W_h.

In figure (3.14), the situation for an update of W_h is shown. On the left, the network without one W_h is drawn (the bra side is not explicitly drawn). As there are only two open indices, this

[6] Yet W is not right-unitary, however, so the transformation $W^\dagger$ is no isometry.
[7] In the presence of symmetry, the horizontal and vertical projectors may naturally be equal.

is basically a matrix M, of which an SVD $M = U\Sigma V^\dagger$ can be calculated as shown on the right. The two indices are numbered to show the relation of the indices between the two forms. In the SVD, U has dimensions $D \times \tilde{D}$, and Σ and V are $\tilde{D} \times \tilde{D}$. This means that U is left-unitary and V is unitary.
In order to optimise W_h, the overlap $\mathrm{tr}(W_h^\dagger M)$ with $M = U\Sigma V^\dagger$ should be maximised. In general, when W_h is complex as in the next paragraph, the overlap is complex as well. In that case, the real part of the overlap should be maximised, which then minimises the norm $\|W_h - M\|_F^2 = \|W_h\|_F^2 + \|M\|_F^2 - \mathrm{tr}(W_h^\dagger M + M^\dagger W_h)$. In the case of square matrices, this problem is known as finding the closest unitary matrix to a given matrix [16]. In our case, M is not square, and W_h is only left-unitary, but the same results hold.
Given that $M = U\Sigma V^\dagger$, we can write

$$\mathrm{tr}(W_h^\dagger M + h.c.) = \mathrm{tr}((V^\dagger W_h^\dagger U)\Sigma + h.c.) = \mathrm{tr}(X\Sigma + h.c.) = \sum_i (x_{ii} + x_{ii}^*)\sigma_i \tag{3.6}$$

where

$$x_{ii} = \sum_{j,k} V_{ji}^* (W_h^*)_{kj} U_{ki} \tag{3.7}$$

As the norms of the rows and columns of W_h, U and V are bounded by unity, $|x_{ii}| \leq 1$. Therefore, the expression in equation (3.6) is maximised when $x_{ii} = 1$. The left-unitary W_h that accomplishes this, is given by $W_h = UV^\dagger$. With this choice for W_h, the overlap becomes

$$\mathrm{tr}(W_h^\dagger U\Sigma V^\dagger) = \mathrm{tr}(VU^\dagger U\Sigma V^\dagger) = \mathrm{tr}(V^\dagger VU^\dagger U\Sigma) = \mathrm{tr}(\Sigma) \tag{3.8}$$

As a test for convergence, the overlap between the old and new E, as partly shown in the bottom of figure (3.13), is calculated. This overlap should converge to unity, with a certain precision dependent on the truncated bond dimension.

Truncation (3): stability

In order to increase the stability of the truncation algorithm, a relaxation step can be added to the update. Care should be taken that the relaxed W_{relax} matrices are still isometries. Geometrically, the update W_{update} can be thought of as a rotation of the old W. A relaxed W_{relax} corresponds to the old W rotated around the same axis, but over a smaller angle.
Since $P_{update} = W_{update} W_{update}^\dagger$ and $P = WW^\dagger$ are both orthogonal projections from a D-dimensional vector space to a $\tilde{D}$-dimensional subspace, they are unitarily equivalent. This is because they share the same spectrum (eigenvalue 1 with multiplicity $\tilde{D}$, and eigenvalue 0 with multiplicity $D - \tilde{D}$). Hence, we can write

$$W_{update} W_{update}^\dagger = UWW^\dagger U^\dagger \tag{3.9}$$

so that $W_{update} = UW$, with U a unitary matrix. Taking the eigenvalue decompositions $W_{update} W_{update}^\dagger = V_{update} \Lambda V_{update}^\dagger$ and $WW^\dagger = V\Lambda V^\dagger$, we can write

$$V_{update} = UV \tag{3.10}$$

$$U = V_{update} V^\dagger \tag{3.11}$$

Since U is unitary, it can be reformulated as $U = \exp(iK)$ with K a Hermitian matrix. This means that

$$iK = \log(V_{update}V^\dagger) \tag{3.12}$$

A relaxed rotation U_{relax} is constructed

$$U_{relax} = \exp(i\alpha K) \tag{3.13}$$

by taking $\alpha \in (0, 1)$. Finally, $W_{relax} = U_{relax}W$.

Convergence

Once a truncated E has been found, it is update with T again, et cetera, until convergence. The measure for convergence is x as defined in figure (3.15). Starting from the environment E, M and m are constructed as shown in the figure. On these two tensors, the corner method is applied, in which the environments L and l (which are normalised) are calculated. The ratio x may at first sight seem to appear out of thin air, but in Chapter 6, it is shown that x leads directly to the free energy and is as such actually a very physical measure for convergence.

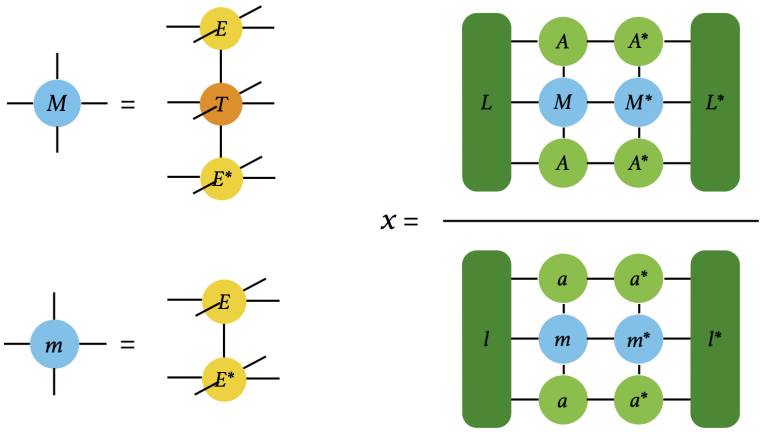


Figure 3.15: Convergence measure for the 3D contraction algorithm.

3.3.2 Relating SVD to Projectors

One can wonder how the method of keeping the largest Schmidt coefficients relates to finding the projection on an optimal subspace of a certain dimension. In the corner method, see figure (3.7), the new A' and B' are obtained by taking an SVD of M and discarding a certain amount of small Schmidt coefficients.
Figure (3.16) shows an alternative way of truncation using projectors: M is the original matrix and M' is the truncated approximation. As U and V are unitary, they drop out of the figure. In a certain basis, the projector $P = WW^\dagger$ is diagonal, consisting of ones and zeros. It is in that basis that the overlap is maximised, as Σ is diagonal as well. The overlap of figure (3.16) then reduces to

$$\mathrm{tr}(\Sigma P \Sigma P) = \mathrm{tr}(\Sigma^2 P^2) = \mathrm{tr}(\Sigma^2 P) \tag{3.14}$$

which is clearly maximised if the ones in P occupy the upper entries, and the zeros the lower entries, provided that Σ is ordered from the greatest to the smallest singular value. There is

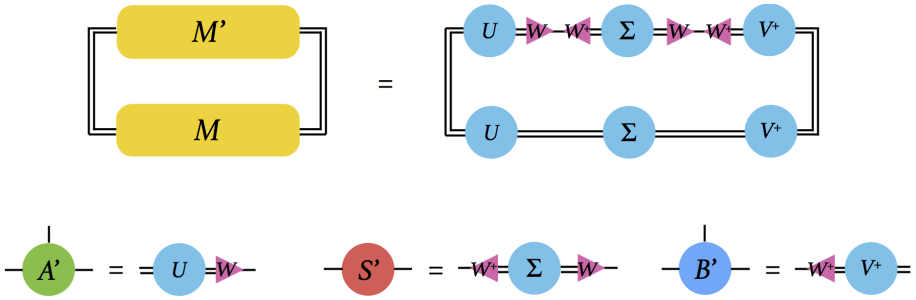


Figure 3.16: Alternative truncation step for the corner method.

gauge freedom in the choice of W, but the easiest choice is to choose W as an identity matrix with rows of zeroes added as necessary. In this manner, the original truncation is retrieved. This argument also validates the disposal of the smallest singular values. The elegant form of P is due to the fact that through the SVD, M is rewritten in the basis of the left- and right-unitary vectors. If one desires to retain the largest singular values in order to maximise equation (3.14), precisely the projection on those vectors is needed.

Chapter 4

Classical Ising Model

This chapter provides the necessary theoretical aspects of the classical Ising needed in the numerical simulations. This encompasses both analytical solutions - for 1D and the most simple 2D case - and other theoretical considerations such as Peierls bounds on the critical temperature. The simulations themselves are discussed in Chapter 5 for the 2D case, and Chapter 6 for the 3D case. The numerical results for 1D are included at the end of this chapter, since they are so simple and merely act as an illustration of straightforward power iteration.

4.1 History and Importance of the Ising Model

The Ising model is arguably the most famous model in statistical physics. Originally proposed by Wilhelm Lenz (1888-1957) and developed by his student Ernst Ising (1900-1998) in 1924, the initial goal of the Ising model was to describe ferromagnetism. Despite its failure to provide a satisfactory explanation for that phenomenon, the Ising model is still a fruitful research topic in a variety of scientific branches.

Through a nearest-neighbour term in the Hamiltonian, the Ising model is capable of describing interactions in a many-particle system. As such, it can be used as a simplification of complex interactions in many-particle systems. Many examples can be found in the literature, on a variety of subjects. I present a short list of examples to get a feeling of the possibilities of applying the Ising model in (1) biology: hydrophobicity structures in protein chains [22]; (2) genetics: analysis of genetic markers [28]; (3) economy: the value of money resulting from trading [4]; et cetera.

Probably the most attractive feature of the Ising model is its simplicity. It is one of the select group of models for which some analytical solutions do exist, while still showing a phase transition. This makes the Ising model useful for testing whether a given computational method works in the correct way (which is exactly what will be done in this chapter).

In his thesis of 1924 and the subsequent publication in 1925 [23], Ising solved 'his' model in one dimension. This does not give rise to a phase transition, from which Ising concluded, through a wrong generalisation, that no phase transitions would occur in any dimension. Sir Rudolf Peierls (1907-1995) discovered in 1936 that such a phase transition was possible after all in two dimensions, by what is now called a Peierls argument [37]. Finally, in 1944, Lars Onsager (1903-1976), solved the two-dimensional case in zero field analytically [32].

4.2 Theoretical Results

4.2.1 Ising Hamiltonian

The most general form of the Ising Hamiltonian on a general n-dimensional spin lattice is given by

$$H = -\sum_{<i,j>} J_{ij} S_i S_j - \mu \sum_i h_i S_i \tag{4.1}$$

where i runs over all the lattice sites. $< i, j >$ means that site j should be a nearest-neighbour of site i (counting each pair only once). As this is a classical model, spin S is just a scalar assigned to each lattice site, where $S_i = \pm 1$. J_{ij} is the interaction strength between sites i and j; h_i is the external magnetic field on site i and μ is the magnetic moment.
In the following, several simplifications will be made. Firstly, we will only consider cartesian lattices, giving two nearest neighbours in 1D, four in 2D, and six in 3D. Secondly, we will consider the case where the interaction strength and external field are constant, i.e. the Hamiltonian is translation invariant:

$$H = -J \sum_{<i,j>} S_i S_j - h \sum_i S_i \tag{4.2}$$

where h has been redefined to include μ. Furthermore, the energy unit is chosen such that $J = \pm 1$. Positive J means that parallel alignment of two neighbouring spins lowers their interaction energy. This means that taking $J = +1$ leads to a ferromagnetic model.

4.2.2 Useful Results from Statistical Physics and Thermodynamics

Of course, the goal of looking at the Ising model is to calculate some interesting physical quantities. Therefore, a brief summary of useful results from statistical physics and thermodynamics is given here [1]. We work at a fixed temperature, and with a fixed volume and number of particles (taking the thermodynamical limit afterwards), so in a canonical ensemble. The canonical partition function $\mathcal{Z}$ is then given by

$$\mathcal{Z} = \sum_s e^{-\beta H_s} \tag{4.3}$$

where the sum over s stands for a sum over all possible configurations (microstates) of the lattice. β is the reciprocal temperature; in all formulæ and calculations $k_B = 1$. The probability that the system is in a certain configuration is given by

$$P_s = \frac{e^{-\beta H_s}}{\mathcal{Z}} \tag{4.4}$$

The expectation value of an operator O, i.e. the energy or magnetisation, can then be calculated according to

$$\langle O \rangle = \sum_s P_s O_s \tag{4.5}$$

The Helmholtz free energy $F \equiv U - TS$ can also be rewritten in terms of the partition function:

$$F = -T \ln(\mathcal{Z}) \tag{4.6}$$

If the partition function is known (however, in the numerical solutions, the expectation values will rather be calculated directly), the expectation values for the energy U and magnetisation $M = \sum_i S_i$ can also be found by:

$$\langle U \rangle = -\frac{\partial \ln(\mathcal{Z})}{\partial \beta} \tag{4.7}$$

$$\langle M \rangle = T\frac{\partial \ln(\mathcal{Z})}{\partial h} \tag{4.8}$$

4.2.3 Solution in 1D

Before going to higher dimensions, it is useful to look at the analytical solution in 1D, i.e. for a chain of N sites. The solution given here uses the transfer matrix technique[1], which allows for quantum-classical mapping on tensor networks as discussed in section 2.6. We begin by rewriting the Hamiltonian in a more symmetrical way, at the same time imposing PBC $S_{N+1} = S_1$:

$$H_s = -J\sum_{i=1}^{N} S_i S_{i+1} - \frac{h}{2}\sum_{i=1}^{N}(S_i + S_{i+1}) \tag{4.9}$$

In order to calculate the partition function, we need to sum $e^{-\beta H_s}$ over the states. This sum is more explicitly given by

$$\sum_s = \sum_{S_1=\pm 1}\sum_{S_2=\pm 1}\cdots\sum_{S_N=\pm 1} \tag{4.10}$$

We can now write each sum as a matrix product. Let us consider the transfer matrix

$$T = \begin{bmatrix} e^{\beta J+\beta h} & e^{-\beta J} \\ e^{-\beta J} & e^{\beta J-\beta h} \end{bmatrix} \tag{4.11}$$

where the (1,1) element stands for (S_i=1, $S_{i+1} = 1$), et cetera. Starting with T, describing, let us say, the interaction between sites N and $N + 1 \equiv 1$, each sum can be replaced by a further multiplication with T. Eliminating all sums except for the first one, we get

$$\mathcal{Z}_N = \sum_{S_1=\pm 1} T^N = \mathrm{tr}(T^N) = \lambda_1^N + \lambda_2^N \tag{4.12}$$

with $\lambda_{1,2}$ the eigenvalues of T. This procedure is shown in figure (4.1).

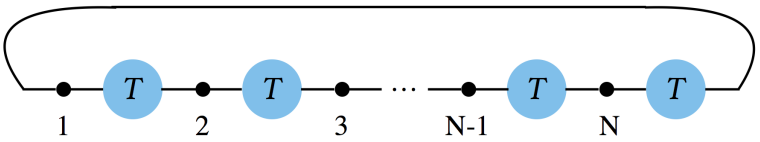


Figure 4.1: Transfer matrix formulation of the 1D Ising model.

After some elementary algebra, we arrive at the following expression for the eigenvalues of T:

$$\lambda_{1,2} = e^{\beta J}\left(\cosh(\beta h) \pm \sqrt{\sinh^2(\beta h) + e^{-4\beta J}}\right) \tag{4.13}$$

[1] To avoid confusion, it should be noted that this transfer matrix is not the same as the MPO transfer matrix of section 2.5.

We are interested in the thermodynamical limit, i.e. $N \to \infty$, so we only need to consider the plus sign. Both the partition function and the total free energy diverge in this limit. However, $\mathcal{Z}$ diverges exponentially, while F only increases linearly with N. Therefore it makes sense to look at the free energy per site, f:

$$f \equiv \frac{F}{N} = -T\left(\beta J + \ln\left(\cosh(\beta h) + \sqrt{\sinh^2(\beta h) + e^{-4\beta J}}\right)\right) \tag{4.14}$$

In absence of an external field, this reduces to

$$f_{h=0} = -T\ln(2\cosh(\beta J)) \tag{4.15}$$

From equations (4.7) and (4.8), we see that the expectation value of the energy and magnetisation also increase linearly with N. In the same way as for the free energy, we now define and calculate the average energy and magnetisation per site (in absence of an external field):

$$u_{h=0} \equiv \frac{\langle U\rangle_{h=0}}{N} = -\frac{J}{2}\tanh(\beta J) \tag{4.16}$$

$$m_{h=0} \equiv \frac{\langle M\rangle_{h=0}}{N} = \left.\frac{\sinh(\beta h)}{\sqrt{\sinh^2(\beta h) + e^{-4\beta J}}}\right|_{h=0} \tag{4.17}$$

where different terms in $m_{h=0}$ have already been replaced by their value for $h = 0$. For finite temperatures, $m_{h=0} = 0$. However, if $T \to 0$ or $\beta \to \infty$, then $m_{h=0} = \pm 1$ depending on the sign of h. At first sight, it is slightly odd that the sign of h comes into play even though $h = 0$. Mathematically, this is just a consequence of taking the limit either from the left or from the right. Physically, this can be interpreted as spontaneous symmetry breaking: a priori, neither value for m is preferred (note that the zero-field Ising Hamiltonian is invariant under global spin flips), but in the physical realisation one of both values is chosen.
Apart from this subtlety around $T = 0$, there is no phase transition in one dimension: there appears no ferromagnetic phase, only a paramagnetic phase. Or in other words, the phase transition occurs at zero temperature and the system is in the paramagnetic phase for all finite temperatures.

4.2.4 Presence of a Phase Transition in Higher Dimensions and Peierls Arguments

In higher dimensions, a phase transition does appear. As mentioned in the introduction to this chapter, this was shown by Peierls for the two-dimensional case before the analytical solution was known. The fact that an analytical solution has been found, does not make the Peierls arguments (the plural denoting similar arguments by other authors, which are all commonly referred to as Peierls arguments) lose all their interest. Since in higher dimensions, no analytical solution is known, Peierls arguments are useful in those cases. A rigorous approach would lead us too far off the main topic, but can be found in the literature, e.g. [8], [17], [3]. Instead, I give a more intuitive argument for the existence of a phase transition.

Let us start by returning to the zero-field one-dimensional case. Assume that all the spins are +1 ('up') and define a Peierls droplet as a region of L adjacent spins -1 ('down'). If such a droplet

has the tendency to decrease in size, the system is in a ferromagnetic phase; if the droplet size increases, the system is in a paramagnetic phase.
The energy cost of creating a droplet is $4J$, i.e. the cost of flipping two spins such that at the boundaries we go from parallel alignment to antiparallel alignment. To determine the entropy advantage, we have to count the number of microstates in which a given spin is part of a droplet and take the logarithm. For droplets of size L, a given spin can be part of L droplets, so $\delta S = \ln L$. The total free energy change when creating a droplet becomes

$$\delta F = 4J - T \ln L \tag{4.18}$$

which means that, unless $T = 0$, the droplet tends to increase in size. Therefore, there is no ferromagnetic phase at finite temperatures.

In two dimensions, we redefine L as the number of sites on the perimeter of a droplet. This gives $\delta E = 2JL$. In order to determine the number of droplets n_L of size L a site can belong to, we look at the perimeter as a random walk problem of length L on the graph induced by the lattice. At each site, there are four possibilities for the next move, hence the total number of random walks $4^L = e^{L \ln 4}$.
Since the perimeter of a droplet is closed, this estimate is an upper bound, albeit a rather crude one. It can be refined by banning back-tracking steps, i.e. allowing only three possibilities at every vertex. In addition, the perimeter must not intersect, so at some vertices only two directions are possible. Even with these two conditions, not all unwanted walks are eliminated, but the upper bound is now restricted enough to be of interest. Replacing 4 in the first estimate by $2 < C < 3$, we get $\delta S = L \ln C$ and finally

$$\delta F = L(2J - T \ln C) \tag{4.19}$$

(i) If $T < T_c^* \equiv \frac{2J}{\ln C}$, with T_c^* the (approximation for the) Curie - or critical - temperature, after Pierre Curie (1859-1906), $\delta F > 0$ which means that the droplet will decrease in size; the system is in a ferromagnetic phase.
(ii) If $T > T_c$, the system behaves paramagnetically.
(iii) At $T = T_c$, there is no advantage in either way. This deserves more explanation, which can be found in section 4.2.6.

In 1941, Kramers and Wannier [26] found an exact solution for T_c by what is now called Kramers-Wannier duality. Their basic idea was to look at the dual system. The dual lattice of a square lattice is square again, but by taking the dual one goes from high to low temperature (disorder to order). A hand-waiving argument is that, through the Heisenberg relations for Fourier transforms, a broad signal (corresponding to high temperature) is converted into a narrow signal (low temperature). If one then lowers the temperature in one system, it increases in the other one. When assuming only one phase transition, both systems meet in one point, namely T_c. The result is $T_c \approx 2.269J$ (cf. (4.29)).
Comparing this to $C = 2$ and $C = 3$ we get

$$\frac{2J}{\ln 3} \approx 1.820J < T_c \approx 2.269J < \frac{2J}{\ln 2} \approx 2.885J \tag{4.20}$$

while the exact $C \approx 2.414$, which shows the agreement of the estimate with the exact solution.

The intuitive Peierls argument looses its clarity in 3D and more involved calculations are necessary. As an upper bound T_c^+, we can use the following result from [11]:

$$T_c^+ = \frac{2J}{\ln\left(\frac{z}{z-2}\right)} \tag{4.21}$$

where z is the number of nearest neighbours. In two dimensions this gives precisely $T_c^+ = 2.885J$; in three dimensions $T_c^+ = 4.933J$. Even better bounds can be found, e.g. $T_c^+ = 4.865J$ in [12]. As a lower bound, we use the result of [13]. This bound has not been found by looking at random walk problems, but by finding a bound on the correlation $\langle S_i S_j \rangle$. The result temperature estimate is $T_c^- = 3.957J$.
These two bounds are in accordance with Monte Carlo simulations as in [2], [10], [40], where $T_c = 4.511J$.

4.2.5 Results in 2D

As mentioned in the introduction, the analytical zero-field solution for a two-dimensional square lattice was published by Onsager in [32]. To be more precise, the partition function and energy are solved there, with the possibility of a different interaction strength horizontally from vertically (thus including the linear chain). Onsager's solution is mathematically complex, involving quaternions and Lie algebra's in order to solve an eigenvalue problem as in the 1D case. Since his solution, others have found simpler approaches, such as Schultz, Mattis and Lieb in [43] via second quantisation for fermions. Obviously, we need not be concerned with the details in this work, so only the results are given.

The energy is given by

$$u = -J \coth 2J\beta \left(1 + (2\tanh^2 2J\beta - 1)\frac{2}{\pi}K(x)\right) \tag{4.22}$$

with $K(x)$ the complete elliptical integral of the first kind, defined as

$$K(x) = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1 - x^2 \sin^2\theta}} \tag{4.23}$$

and

$$x = \frac{2\sinh 2J\beta}{\cosh^2 2J\beta} \tag{4.24}$$

The partition function per site is constructed as follows:

$$\lambda = \ln(2\cosh 2J\beta) + \frac{1}{\pi}\int_0^{\pi/2} d\theta \ln\left(\frac{1}{2}\left(1 + \sqrt{1 - x^2\sin^2\theta}\right)\right) \tag{4.25}$$

$$z \equiv \lim_{N\to\infty} (\mathcal{Z})^{1/N} = e^{\lambda} \tag{4.26}$$

The free energy is of course

$$f = -T\ln z = -T\lambda \tag{4.27}$$

In 1952, Yang published the result for the magnetisation [53].

$$m = \pm \left(1 - (\sinh 2J\beta)^{-4}\right)^{\frac{1}{8}} \tag{4.28}$$

for $T \leq T_c$ and 0 for $T > T_c$.
The Curie temperature itself is given [26] by

$$\sinh 2J\beta_c = 1 \tag{4.29}$$

or

$$T_c = \frac{2J}{\ln\left(1+\sqrt{2}\right)} \approx 2.269J \tag{4.30}$$

4.2.6 Behaviour at the Phase Transition(s) and Critical Exponents

All these quantities are continuous. Note that the energy and magnetisation are a first order derivative of the logarithm of the partition function, i.e. of the free energy. The heat capacity $c = \frac{\partial e}{\partial T}$ and magnetic susceptibility $\chi = \frac{\partial m}{\partial h}\big|_{h=0}$ on the other hand are second order derivatives. These quantities are not continuous, hence the phase transition is second order.
The case $h \to 0$ is the critical point of a first-order transition: the magnetisation is not continuous at this point but changes sign (for $T < T_c$) as h changes sign. Another way of looking at the spontaneous symmetry breaking at $h = 0$, is to change the boundary conditions and place the system in a bigger spin ± 1 sea. One can then say that for $T < T_c$ the system is dependent on the boundary conditions, while for $T > T_c$ it is not. This way, the notion of an external field has been eliminated, while the influence of an environment will prove a useful concept when discussing the numerical results.

Around the critical point, the different physical quantities may be described by critical exponents [1], defined as follows:

$$t = \frac{T - T_c}{T_c} \tag{4.31}$$

$$\begin{aligned} \chi &\sim |t|^{-\gamma} && (4.32)\\ c &\sim |t|^{-\alpha} && (4.33)\\ m &\sim (-t)^{\beta} && (4.34)\\ |m| &\sim |h|^{\frac{1}{\delta}} && (4.35) \end{aligned}$$

where the last one is defined at $T = T_c$ itself, and the last but one is defined for $T < T_c$.
Away from the critical point, the correlation length ξ is defined as

$$\langle S_i S_j \rangle - \langle S_i \rangle \langle S_j \rangle = e^{-\frac{|i-j|}{\xi}} \tag{4.36}$$

and its critical exponent as

$$\xi \sim |t|^{-\nu} \tag{4.37}$$

This means that the correlation around the critical point scales with the distance (in the temperature 'space') to the critical point as equation (4.37).

On the critical point, the correlation length diverges, but for the correlation itself, a new critical exponent η is introduced such that

$$\langle S_0 S_i \rangle \sim \frac{1}{|i|^{d-2+\eta}} \tag{4.38}$$

with d the dimension of the lattice and $|i|$ the distance of site 0 to site i.
These critical exponents obey some scaling laws

$$\alpha + 2\beta + \gamma = 2 \tag{4.39}$$
$$\beta\delta = \beta + \gamma \tag{4.40}$$

and hyperscaling laws ('hyper' meaning that they involve d).

$$d\nu = 2 - \alpha \tag{4.41}$$
$$2 - \eta = d\frac{\delta - 1}{\delta + 1} \tag{4.42}$$

In 2D, the values for these exponents are:

$$\alpha = \log \qquad \beta = \frac{1}{8} \qquad \gamma = \frac{7}{4} \qquad \delta = 15 \qquad \nu = 1 \qquad \eta = \frac{1}{4} \tag{4.43}$$

where "log" means that $c \sim -\log(|t|)$. In the scaling laws, however, α counts for 0.

4.2.7 Low- and High-Temperature Limits of the Free Energy

In addition to Peierls arguments about a phase transition, we shall now have a look at low- and high-temperature limits of the free energy. These two results combined enable us to make a first assessment of computational results.
As $f = u - Ts$, we have to look at both the energy and the entropy. For the low-temperature limit $T \to 0$, only u needs to be considered. Since we know that for $T \to 0$ the system is in its ferromagnetic phase[2] (for the case $J = +1$), we can take all spins $+1$. From equation (4.1), it follows that $U = -\frac{k}{2}N - hN$, where k is the number of nearest-neighbours or coordination number. Therefore, $u = -\frac{k}{2} - h$, and as $T \to 0$, $f \to -\frac{k}{2} - h$.
For $T \to \infty$, $U \to 0$ and only an entropic contribution $f = -Ts$ is present. For high temperatures, the fraction of spins $+1$ approaches $N/2$ (following a $\sqrt{N}$ rule, such that this is exact in the thermodynamical limit). As a consequence, the number of microstates corresponding to the macroscopic non-magnetised state is given by

$$\Omega = \frac{N!}{(\frac{N}{2})!\,(\frac{N}{2})!} \tag{4.44}$$

Using Stirling's approximation $\ln(x!) \approx x\ln(x) - x + O(\ln(x))$, we get for the (Boltzmann) entropy $S = \ln(\Omega)$

$$S \approx N\ln(2) \tag{4.45}$$

[2] For 1D, this only holds in the limit itself.

so that $f \to -T\ln(2)$ as $T \to \infty$.

4.3 Numerical Results in 1D

In order to give a brief example of how power iteration works in its most basic form, i.e. with matrices, let us start by looking at the numerical results for the one-dimensional Ising model. As can be seen in figure (4.2) for the free energy, the numerical results agree with the theoretical solution. (The tolerated error in these simulations was 1.10^{-8}). In figure (4.3) the effect of an external field on the magnetisation is depicted. As expected, for $h = 0$, no ferromagnetic phase is present. By increasing h, on the other hand, there is alignment in the low-temperature region. It should be noted that the real purpose of this comparison between theory and simulation does not lie in the agreement itself: we shall not forget that in the analytical solution too, a power iteration was assumed by discarding the smallest eigenvalue of the transfer matrix. Therefore, the real interest of figure (4.2) lies in the fact that even for a modest number of iterations n, nearly perfect agreement can be obtained. Figure (4.4) shows the number of iterations needed for a certain degree of accuracy. One can conclude that a high degree of accuracy is already obtained after a low number of iterations. This means that, indeed, power iteration has practical use.
The number of required iterations is not constant over the temperature domain: n has a maximum. This maximum is a marker for the temperature region where the thermal energy starts to outweigh the effect of the external field, resulting in random alignment at higher temperatures.

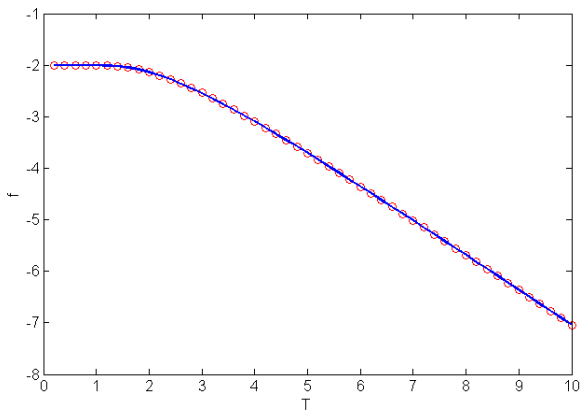


Figure 4.2: Free energy of the 1D Ising model with $h = 1$. Solid line is the theoretical result.

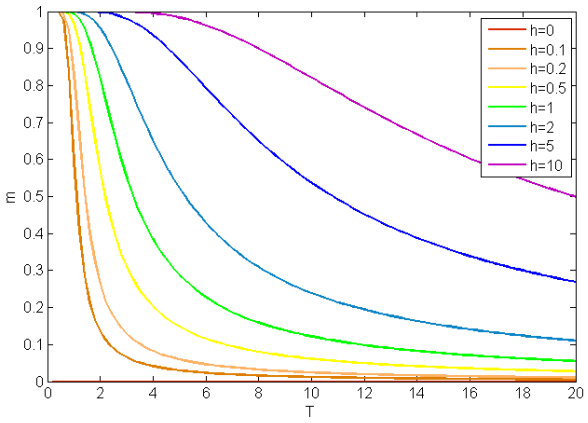


Figure 4.3: Magnetisation of the 1D Ising model for different external fields h.

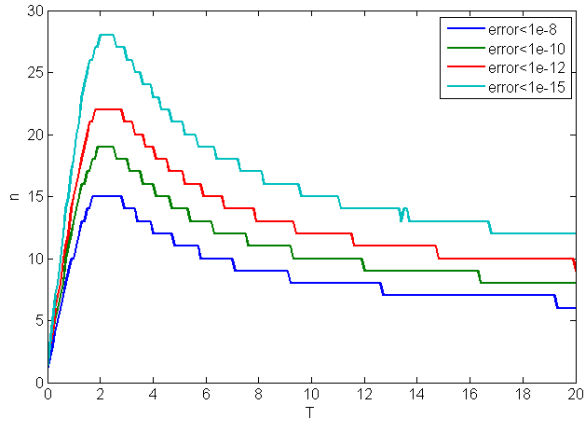


Figure 4.4: Number of iterations needed for a certain degree of accuracy in power iteration.

Chapter 5

Numerical Results in 2D

In this chapter, the numerical results for the 2D Ising model are discussed. The first section explains how the 2D Ising Model is translated into the TN formalism. In the next section, it is explained how physical quantities such as the magnetisation are calculated. Afterwards, we discuss the numerical results, starting with the convergence properties. The physical results for the Ising model follow, both without and with an external field. Lastly, the TN results are compared to Monte Carlo simulations.

5.1 Rewriting the Ising Model

In order to cast the Ising model in a form suitable for tensor network methods, we use the same transfer matrix (4.11) as in the one-dimensional case, with $h = 0$. This leads to the network shown in figure (5.1). That network is not suitable for calculations yet, for the transfer matrices

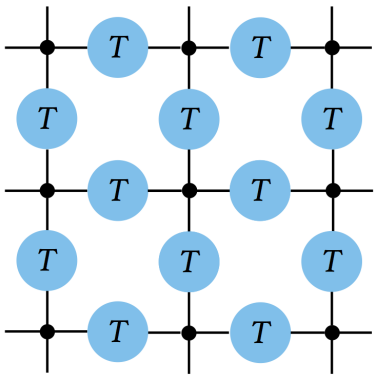


Figure 5.1: Tensor network describing the Ising model interactions.

appear between two vertices. The first question is what should happen at the vertices, where four edges (indices of the transfer matrix) join. The physical meaning of a vertex is a ± 1 spin. If the vertex is interacting with its neighbours, it should take on the same spin value for all of them, meaning it should interact through the same row column of the transfer matrix, depending on its spin value. Therefore, at all vertex points a sort of Kronecker delta tensor should be placed:

$$\delta(i,j,k,l) = \begin{cases} 1 & \text{if } i = j = k = l \\ 0 & \text{elsewhere} \end{cases} \tag{5.1}$$

Next, the transfer matrices need to be contracted with the vertices. A general method can be, for example, to contract every horizontal T with its left neighbour and every vertical T with its upper neighbour. This is certainly an easy method, however, in this way, left/right and up/down symmetry is lost. This means that in the numerical methods, more environments will need to be calculated.
In the ferromagnetic case, T is positive definite and can be rewritten[1] as $T = t^2$. When this square root has been found, a more symmetrical contraction of the transfer matrices is to put one square root per T in both vertices, as shown in figure (5.2). The square root t itself can

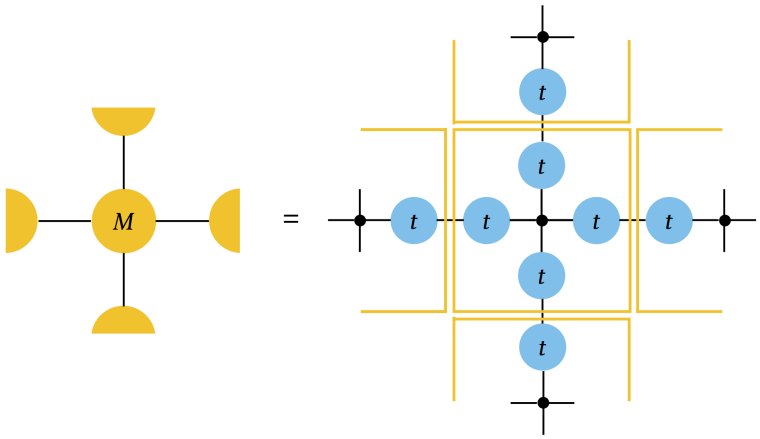


Figure 5.2: Symmetrical composition of the final tensor network out of the transfer matrices.

be found after some algebra:

$$t = \sqrt{\frac{1}{2}} \begin{bmatrix} \sqrt{\cosh\beta} + \sqrt{\sinh\beta} & \sqrt{\cosh\beta} - \sqrt{\sinh\beta} \\ \sqrt{\cosh\beta} - \sqrt{\sinh\beta} & \sqrt{\cosh\beta} + \sqrt{\sinh\beta} \end{bmatrix} \tag{5.2}$$

On this tensor network, the methods of Chapter 3 can be applied.

5.2 Calculating Physical Properties

5.2.1 Magnetisation and Energy

Since the partition function diverges for increasing lattice size, a normalisation is required in each iteration step. Therefore, we need to change the meaning of the normalisation in (4.4). Since the environments are normalised, the new normalisation factor is shown in figure (5.3).

The easiest physical quantity to calculate is the average magnetisation m. Let us recall that the partition function is given by the sum of the Boltzmann factors over all the states, cf. (4.3); for the expectation value the spin operator is added to the summand, cf. (4.5). Therefore, the delta tensor needs to be replaced at one vertex by a tensor describing the spin operator. This is easily achieved by

$$\delta_S(i,j,k,l) = \begin{cases} 1 & \text{if } i=j=k=l=1 \\ -1 & \text{if } i=j=k=l=2 \\ 0 & \text{elsewhere} \end{cases} \tag{5.3}$$

[1] The square root is not unique, but there is only one positive definite square root of a positive definite matrix.

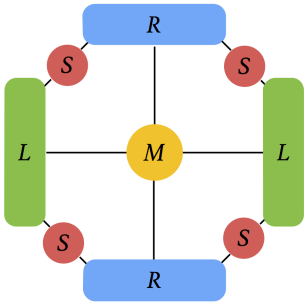


Figure 5.3: New normalisation factor replacing the partition function.

By replacing M in figure (5.3) by the new M_S which contains δ_S instead of δ, we obtain m after division by the original factor.
In order to calculate the average energy e, we need to evaluate products of spin operators. Therefore, a two-site environment is needed, as shown in figure (5.4). The same needs to be done in the vertical dimension and after adding both terms, normalisation is now performed by a two-site 'partition function'.

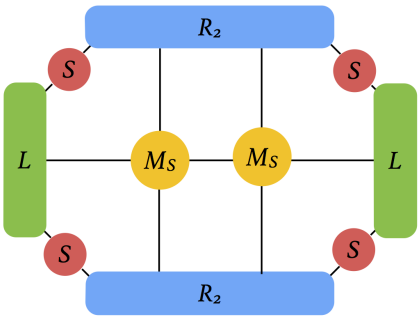


Figure 5.4: Network for calculating the energy.

5.2.2 Free Energy

To calculate the free energy in 1D, one can see from equation (4.12) that in the thermodynamical limit only the largest eigenvalue of the transfer matrix is needed. We shall now construct a similar power method argument in 2D.
Let us contract the tensor network first in the vertical direction (N_y sites) and then horizontally (N_x sites). As $N_y \to \infty$, the environment converges to the eigenvector with the largest eigenvalue of the MPO consisting of one layer of sites. Each time we add a layer after convergence, the eigenvector is multiplied by this eigenvalue, giving us $\mathcal{Z} = \lambda^{N_y}$. Doing this simultaneously from top to bottom and vice versa, we end up with a double layer linear chain. By performing a power method horizontally, through the same reasoning $\lambda = \mu^{N_x}$. In the end, $\mathcal{Z} = \mu^{N_x N_y}$. Since $z = \mathcal{Z}^{1/(N_x N_y)} = \mu$ and $f = -T \ln z$, we finally obtain $f = -T \ln \mu$.
It should first be noted that when all the environments are normalised (such that μ is rescaled to unity in the previous formulæ), the free energy can be obtained by adding one site to the lattice[2]. To find a way to insert this into the corner method for contraction, one can look at the update

[2]So, calculating the free energy in 2D amounts to a vertical contraction, with normalisation, followed by a calculation of the free energy of the resulting linear chain.

of L in figure (3.7). We should remind ourselves that A is left-orthogonal[3]. A corresponding right-unitary update does not emerge naturally from the corner method: this method inherently mixes the horizontal and vertical dimensions, which makes it difficult to implement the previous argument for the free energy literally.
However, it is sufficient to calculate the norm of L', as L' contains precisely one extra T compared to L. Although we can only access $\sqrt{\mu^2}$ in this way, we know that the free energy should be real, and hence μ should be positive. Lastly, it should be noted that it is generally not possible to take the overlap of L' with L, as the left and right bond dimensions of A need not be equal.

5.3 Convergence Properties

Before showing the physical results, it is useful to look at the convergence properties of the simulations. As a first test of the number of iterations necessary for convergence, the bond dimension was increased by one in each iteration step, bounded by a pre-determined maximum. Figure (5.5) shows temperature sweep for a variable maximum of iteration steps. There is of course a one-to-one correspondence between the number of iterations, n, and the bond dimension, $n+1$. Not surprisingly, it is clear from the figure that a higher bond dimension is needed around the

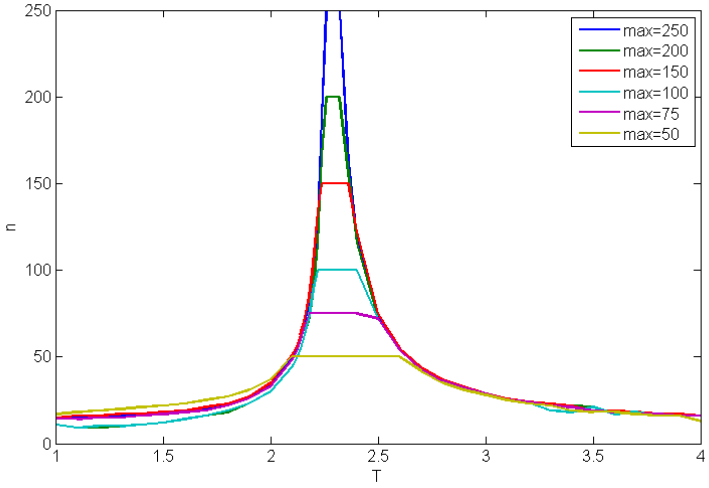


Figure 5.5: Number of iterations necessary for convergence.

critical point, whereas a bond dimension of around 20 suffices in the low and high temperature limits.

In addition to a linear increase, other methods for truncation are possible. For example, one could set a maximum bond dimension and postpone truncation until this maximum value has been achieved, after which the bond dimension is kept constant at the maximum value. This is computationally not very smart, as those high bond dimension are not always needed.
A more adaptive method is what we shall call singular value-induced truncation. In this method, a minimum value σ_{min} is set for the Schmidt coefficients. In each iteration step, the bond dimension is truncated such that only the Schmidt coefficients larger than this minimum are preserved.

[3]Or left-unitary when working over the complex field, which we will use for the calculation of the free energy in 3D.

Analogously to figure (5.5), a temperature sweep of the bond dimension needed for convergence when using singular value-induced truncation is shown in figure (5.6).
Even though a linear increase of the bond dimension and a singular-value induced truncation

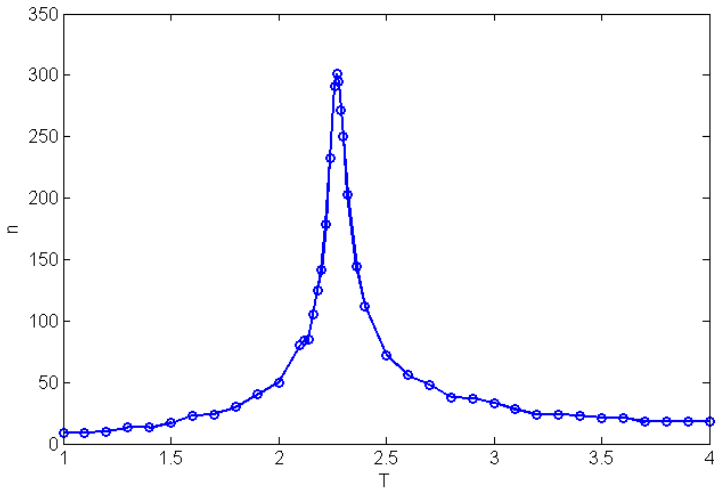


Figure 5.6: Singular value-induced bond dimension needed for convergence with $\sigma_{min} = 1.10^{-6}$ and maximally 150 iterations.

do not show notable differences except for the free energy (vide infra), the latter method is nevertheless more apt for conceptual reasons. There is no physical reason why the bond dimension should be increased linearly. On the other hand, the truncation step is centred around discarding the smallest Schmidt coefficients, such that it makes sense to control the truncation directly at that level.
On top of that, the peak in graphs is more defined for singular-value induced truncation. This means that from this graph a potentially better estimate for the critical temperature can be distilled. For example, when using the results from figure (5.6), this gives $T_c = 2.27 \pm 0.01$ which is in agreement with the theoretical results $T_c \approx 2.269$.

5.4 Physical Results Without an External Field

5.4.1 Thermodynamical Properties

Figure (5.7) shows the magnetisation for a maximal bond dimension of 200. It is clear that the numerical method reproduces the theoretical result.
The influence of the environment mentioned in section 4.2.6 is clear when considering the magnetisation. Depending on the initial value for the environments, the ferromagnetic phase acquires either +1 or -1 magnetisation. In the figures of this text, however, the absolute value is drawn for easier comparison.
The influence of the maximal bond dimension is shown in figure (5.8) where it is varied from 50 to 500. When increasing the bond dimension, the accuracy increases, but it is also noteworthy that the improvement slows down.

Plotting the difference between the numerical and the theoretical value for the magnetisation is a second way to quantify T_c as the maximum of the error is expected to occur at the phase

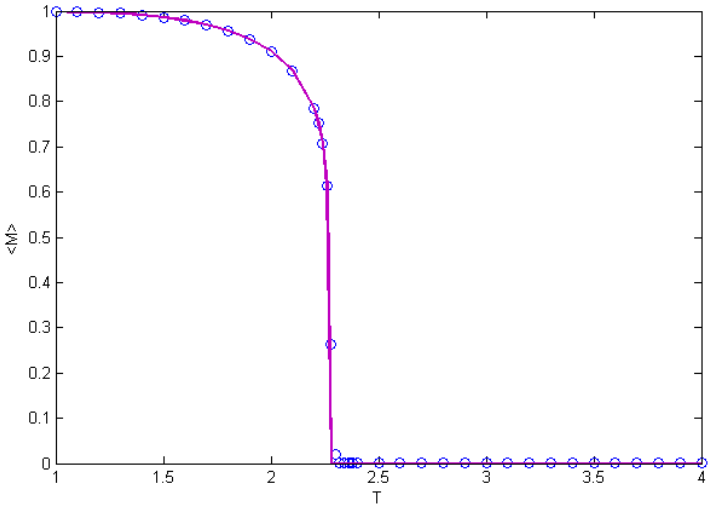


Figure 5.7: Magnetisation of the 2D Ising model for a maximal bond dimension of 200. The solid line is the theoretical result.

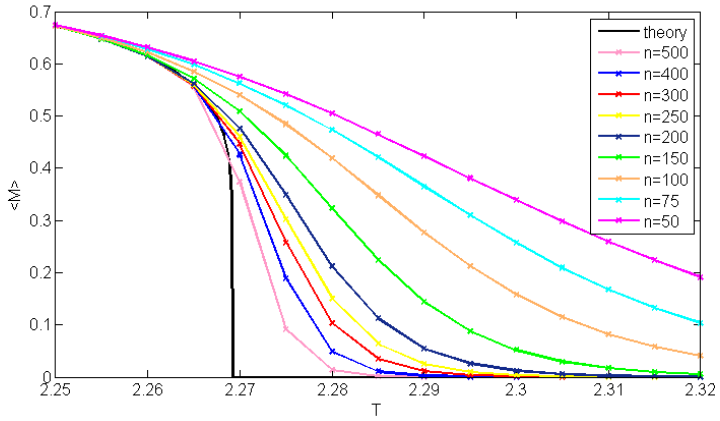


Figure 5.8: Influence of the bond dimension (n) on the magnetisation.

transition[4]. This is shown in figure (5.9) and yields T_c=2.27 for a temperature step of 0.001 around T_c.

For the energy, the results are shown in figure (5.10) for a maximal bond dimension of 150. It is clearly visible that the theoretical results are well reproduced, and with less inaccuracies around the critical point than for the magnetisation.

The results for the free energy are shown in figure (5.11). Two simulations are displayed there: the first one with a linearly increasing bond dimension, the second one with singular value-induced truncation. It is clear from the figure that the latter performs better. The free energy is the only physical quantity discussed here where such a notable difference was visible between the two truncation methods.

[4]So, basically, we are using inaccuracies to our advantage.

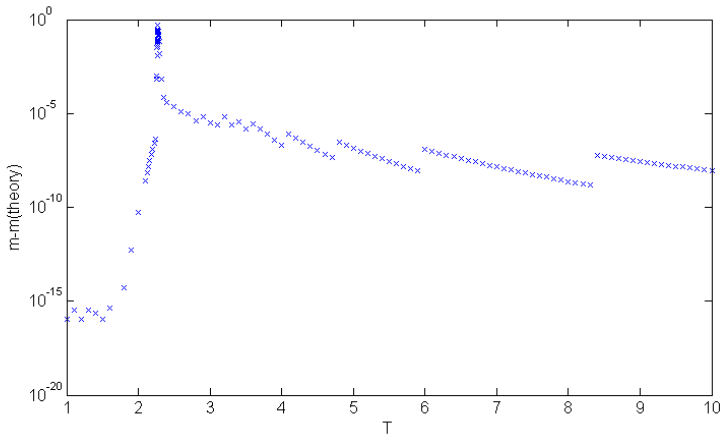


Figure 5.9: Absolute value of the difference between the numerical and theoretical magnetisation of the 2D Ising model. The peak corresponds to $T = 2.27$.

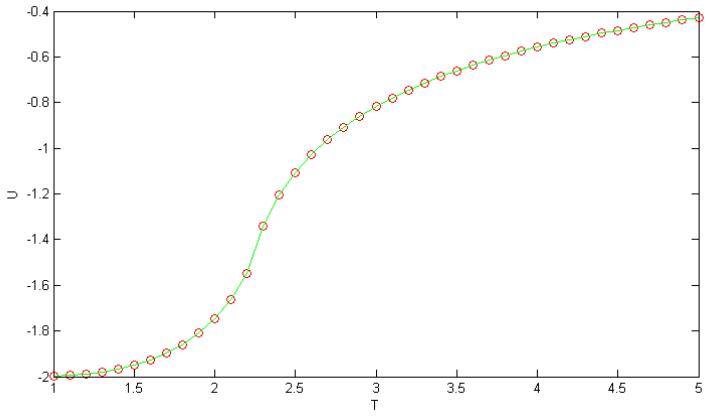


Figure 5.10: Energy of the 2D Ising model for a maximal bond dimension of 150. The solid line is the theoretical result.

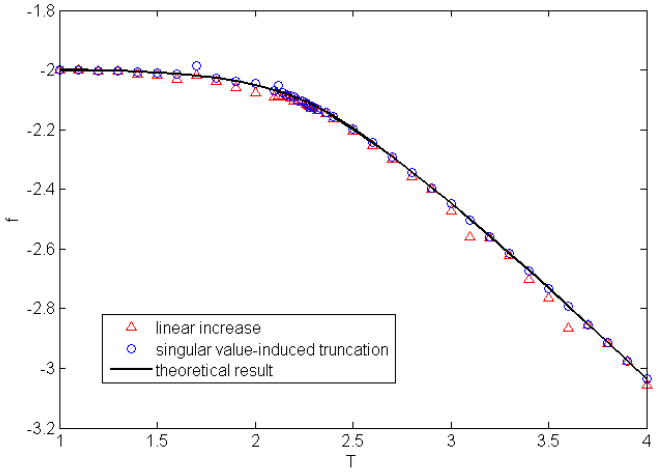


Figure 5.11: Free energy of the 2D Ising model.

5.4.2 Correlations and Magnetic Susceptibility

The magnetic susceptibility χ can be written as [1]

$$\chi = \frac{1}{NT} \langle (M - \langle M \rangle)^2 \rangle = \frac{1}{NT} (\langle M^2 \rangle - \langle M \rangle^2) \tag{5.4}$$

As $M = \sum_k S_k$, χ is a sum of correlations $\langle S_i S_j \rangle - \langle S_i \rangle \langle S_j \rangle$. Because of translation invariance, site i can be fixed at a central site, and the sum over i cancels with the factor $1/N$.
In order to obtain an estimate for the magnetic susceptibility, correlations up to three sites away were calculated as shown in figure (5.12). To this end, three-site environments were needed (due to the symmetry of the quadrants in figure (5.12)). The separate contributions to the susceptibility (i.e. correlations divided by the temperature) themselves are given in figure (5.13).
After taking into account the number of neighbours of a particular kind, the estimate for the magnetic susceptibility is drawn in figure (5.14). There, we have also divided by the number of sites to obtain the magnetic susceptibility per site.
The decay to zero in the paramagnetic phase is not fast enough, but as the correlations with more distant sites decay faster (cf. figure (5.13)), it is expected that this be remedied by taking into account more sites.
In comparison to the previous physical quantities, the results for the susceptibility appear to be much less accurate. As the only 'interesting' temperature region for the susceptibility lies around the critical temperature, this shows again how the behaviour at the phase transition is the most difficult part of the simulation.

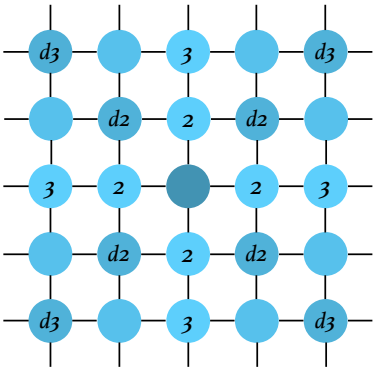


Figure 5.12: Identification of the tensors used in the approximation for the susceptibility.

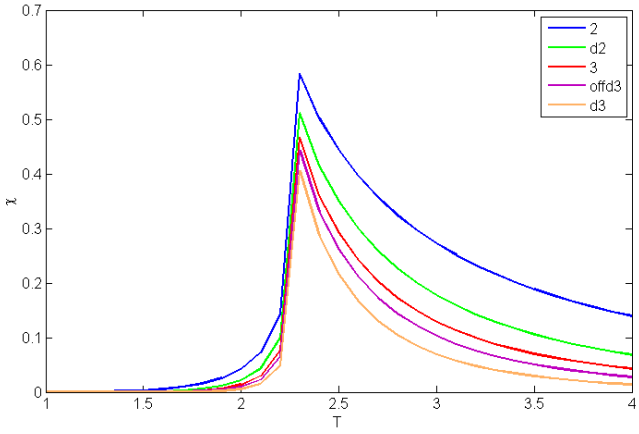


Figure 5.13: Correlations divided by the temperature. The legend matches the labelling of figure (5.12) where the sites 'offd3' are the non-marked, light-coloured ones.

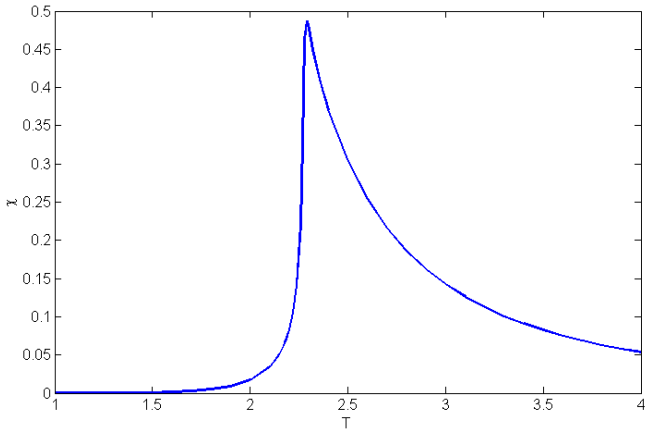


Figure 5.14: Estimate for the magnetic susceptibility

5.4.3 Critical Exponents

With the previous results, we shall now have a look at some of the critical exponents given in section 4.2.6, namely β for the magnetisation as a function of temperature, δ for the magnetisation as a function of the external field, and γ for the susceptibility. Through a fit of the data, a value for these critical exponents can be obtained. In addition, T_c was also treated as an unknown fitting parameter to estimate that as well.

First, the results for $m \sim (-t)^{\beta}$ are shown in figure (5.15). With a goodness of fit $R^2 = 1$, $T_c = 2.267$ with 95% confidence bounds (2.267; 2.267) which is not exactly right, but still in good agreement with the theoretical value. The fitting value of $\beta = 0.1132$ (0.1124; 0.114) is nearly 10% off the correct value $\beta = \frac{1}{8}$.

The fitting for $|m| \sim |h|^{\frac{1}{\delta}}$ yields $\delta = 14.78$ (14.55; 15.01) with $R^2 = 0.9996$ and thus shows a

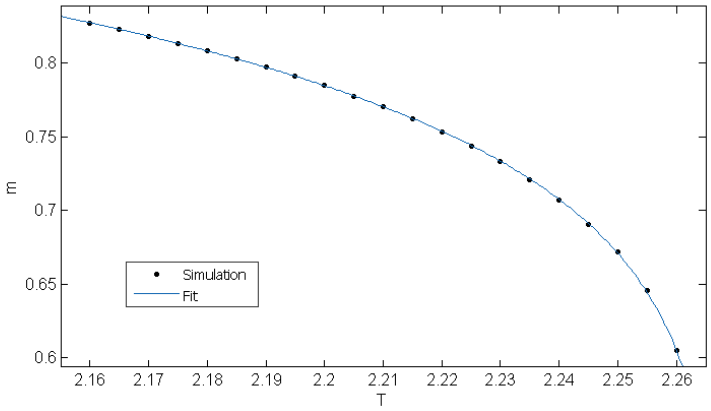


Figure 5.15: Fit for the critical exponent β and T_c.

much better agreement with the theoretical value $\delta = 15$.

As a last example, for $\chi \sim |t|^{-\gamma}$ we get a worse estimate for the critical temperature, $T_c = 2.343$ (2.32; 2.366). The result for γ=1.771 (1.4307; 2.105) lies very close to the theoretical value $\gamma = \frac{7}{4}$, but it needs to be noted that the confidence interval is wide (nearly 40% of γ itself).

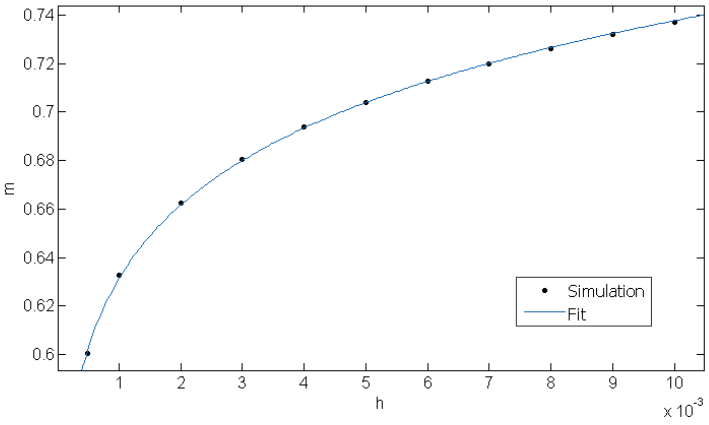


Figure 5.16: Fit for the critical exponent δ.

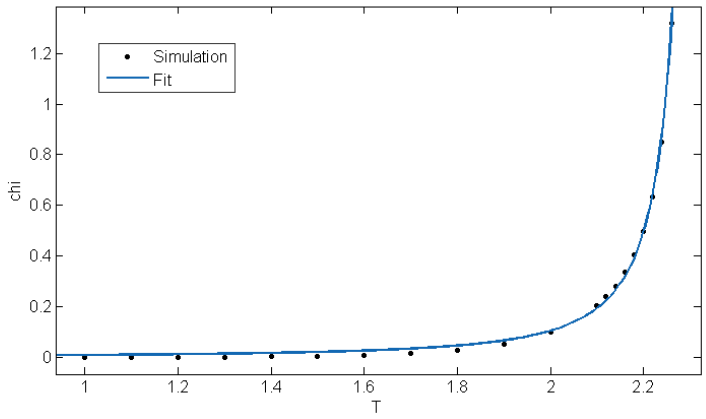


Figure 5.17: Fit for the critical exponent γ and T_c.

5.5 2D Ising Model in the Presence of an External Field

When $h \neq 0$, the complete transfer matrix from equation (4.11) needs to be used. Determining the square root is more complicated now, and the result is given in Appendix B. An important remark on the right way to include h in the transfer matrix is also made in that Appendix.
A first effect of the external field is the fixation of the symmetry breaking: the low-temperature magnetisation is in the same direction as the external field. Secondly, through the ordering effect of the external field, the critical temperature increases and the phase transition becomes smoother. The results for the magnetisation are shown in figure (5.18); in those simulations the minimum singular value in the truncation $\sigma_{min} = 1.10^{-6}$, the tolerance on the error (as defined in section 3.2.2) was 1.10^{-18} and the maximal number of iterations was 200.

An analogous plot for the free energy is given in figure (5.19). It can easily be seen that the correct limits, as derived in section 4.2.7 are retrieved: the low-temperature limit is $f = -2 - h$, while in the high-temperature limit f approaches $-T\ln(2)$. Another comparison of the limits is shown in figure (5.20), where the free energy for both the 1D and 2D Ising model are shown for a few values of h. In that figure, the curves are grouped in colour by their low-temperature limit: the free energy in 1D for an external field h resembles the free energy in 2D for $h - 1$. Of course, for the magnetisation one does not find the same degree of resemblance due to the absence of a phase transition in 1D. But nonetheless, and certainly as h increases, the same grouping as for the free energy is found.

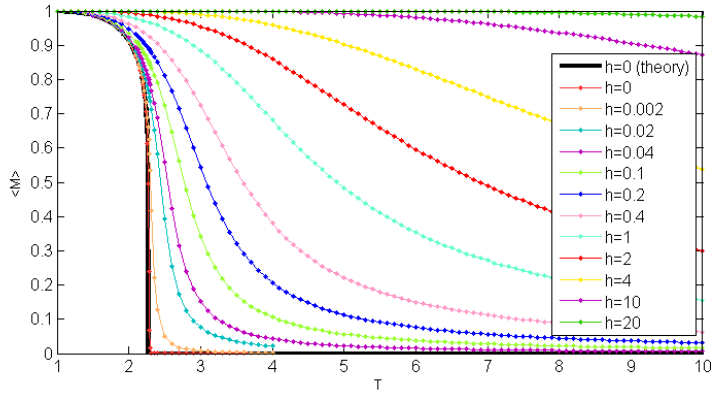


Figure 5.18: Magnetisation of the 2D Ising model with an external field h.

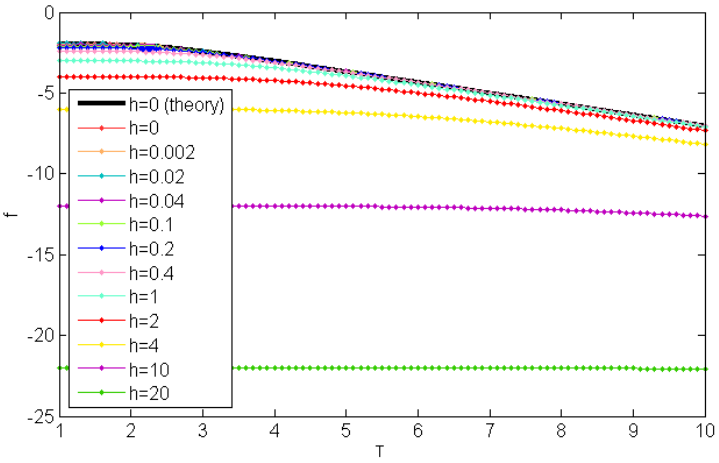


Figure 5.19: Free energy of the 2D Ising model with an external field h.

It is expected that in 3D, the case $h-2$ will correspond to $h-1$ in 2D and h in 1D. That this is indeed the case, is shown in figure (6.7).

Figure (5.22) shows how the number of iterations needed for convergence changes as h is

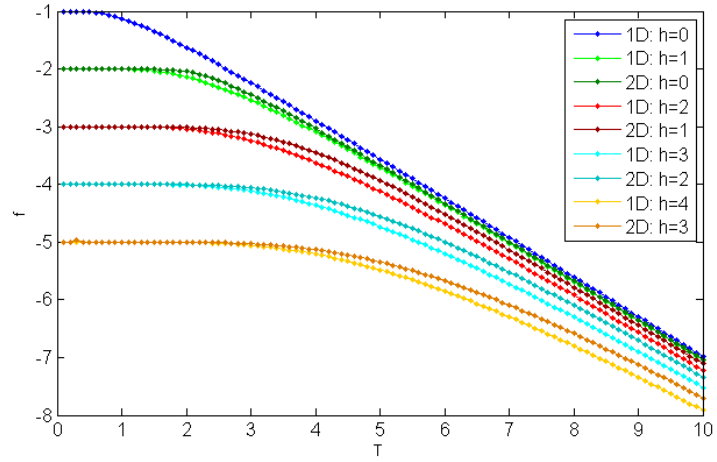


Figure 5.20: Free energy of the 1D and 2D Ising model for various external fields h.

increased. It is clear that this number decreases, and the maxima of the curves move to the right as h grows. This is consistent with the slower and smoother decrease of the magnetisation

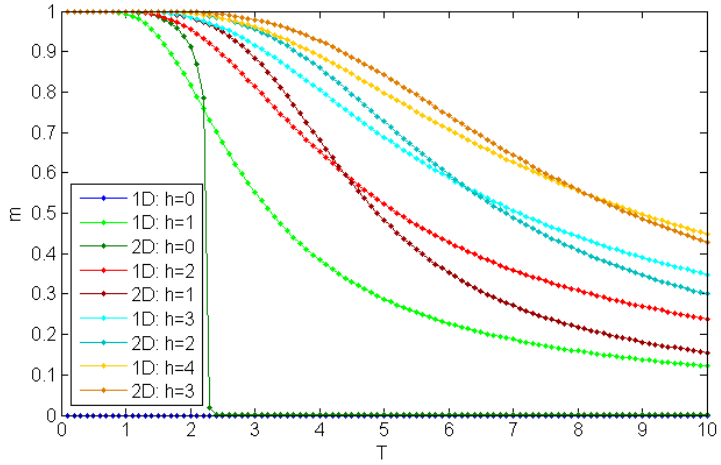


Figure 5.21: Magnetisation of the 1D and 2D Ising model for various external fields h.

shown in figure (5.18). It is noteworthy that the curves are adapted in such a way that they are all enveloped by the curve for $h = 0$, and that each curve bounds all the curves for higher h. Physically, this can easily be explained, e.g. for high temperatures all curves approach the $h = 0$ curve since at high temperatures the ordering effect of h is cancelled due to thermal fluctuations. This physical rationale is transferred to the 'computational level' of the simulation through the convergence rate of subspace iterations, see section 2.5.2.

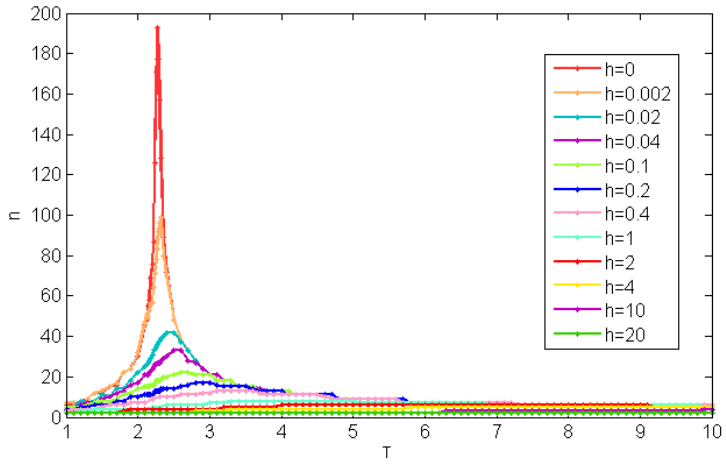


Figure 5.22: Number of iterations needed for convergence, for various external fields h.

5.6 Monte Carlo Simulations

Monte Carlo methods are a widespread approach to problems in statistical physics. As there exists a vast variety of algorithms, a rigorous comparison of TN methods to Monte Carlo methods is not straightforward. Therefore, only a short introduction to Monte Carlo methods is given in this section, followed by some numerical results. The main interest for using Monte Carlo methods within this dissertation is to provide us with some results in the three-dimensional case. In two dimensions, where an analytical solution is known, it was easy to check whether the numerical solution gives the proper results, at least when $h = 0$. In three dimensions, we

shall use a Monte Carlo solution instead.

5.6.1 Introduction to Monte Carlo Methods - Metropolis-Hastings Algorithm

Basic Idea

The exact calculation of the partition function in equation (4.3) would require a sum over all the possible configurations of the lattice, i.e. 2^N terms for an $N \times N$-lattice. As this is generally not feasible, in Monte Carlo simulations they are replaced by a limited number of 'representative' configurations. This requires sampling from the configurations that give the greatest contributions and is called *importance sampling*. Of course, various technical conditions need to be fulfilled, for which we refer to the literature [51].
There exist a variety of Monte Carlo algorithms that can be used in statistical physics. We have used a simple method, namely the Metropolis-Hastings algorithm [29] [21].

Metropolis-Hastings Algorithm

Initially, the lattice is configured in a random state with a certain probability for the spin to be $+1$. An order at which to look at the lattice sites is also determined at random.
When considering a site, the energy advantage of a spin flip is calculated. This depends on the neighbouring sites, so for each site the neighbours are stored in a $N^2 \times 4$-matrix. If it is energetically advantageous, the spin is flipped. However, if a spin flip raises the energy, a spin flip is not excluded. In that case, the spin flip is accepted with a Boltzmann probability $e^{-\beta \Delta E}$.
The magnetisation is obtained by summing all the spins and dividing that sum by the number of sites. For the energy, each spin is multiplied by the spins on the neighbouring sites and then, again, the normalised sum over the whole lattice is taken. This is then repeated for a number of iterations to average out fluctuations. Having said that, sampling is only started after the lattice has come into equilibrium. Indeed, when starting from a random initial configuration, it takes a number of iteration steps for the lattice to evolve towards an equilibrium state.

Discussion

It should be clear that the approach used in Monte Carlo methods is very different from the one used in TN algorithms. In addition to the freedom of choice in Monte Carlo algorithms, this further hinders a comparison between the two strategies. For example, the precision of TN methods is determined by the bond dimension and the number of iterations. In contrast, the accuracy of Monte Carlo methods depends on the lattice size and the number of iterations.
To avoid comparing cheese and chalk, one could therefore - in the case where an analytical solution is known - impose a certain accuracy and determine the corresponding bond dimension and lattice size respectively. The computational efficiency of the two approaches could then be compared by using these values for the bond dimension and lattice size[5].
It is also noteworthy that there are physical quantities that are more suited to one method than

[5] Of course, one should also be wary of the influence of the programming language.

to the other. For example, as the lattice configuration is explicitly constructed in the Metropolis-Hastings algorithm, correlations are easily calculated, and hence the magnetic susceptibility as well. The corresponding calculation in the TN viewpoint is more involved, as explained in section 5.4.2. On the other hand, with TNs it is easy and natural to calculate the free energy (see section 5.2), while this is not straightforward in Monte Carlo simulations as the free energy cannot be expressed as a canonical ensemble average.

5.6.2 Some Results and Their Comparison with TN Simulations

Figures (5.23) and (5.24) show some results of Monte Carlo methods compared to the results using the corner method. For the Monte Carlo simulations, 500 iteration steps were used to allow the lattice to equilibrate, followed by another 500 iteration steps for calculating the magnetisation and free energy. This was done for three lattice sizes, namely 10×10, 20×20 and 50×50. A good agreement between TN and Monte Carlo results can be seen, quantitatively for the free energy and the ferromagnetic magnetisation, and at least qualitatively for the magnetisation around the phase transition[6]. As expected, the accuracy of the Monte Carlo simulations increases with increasing lattice size.

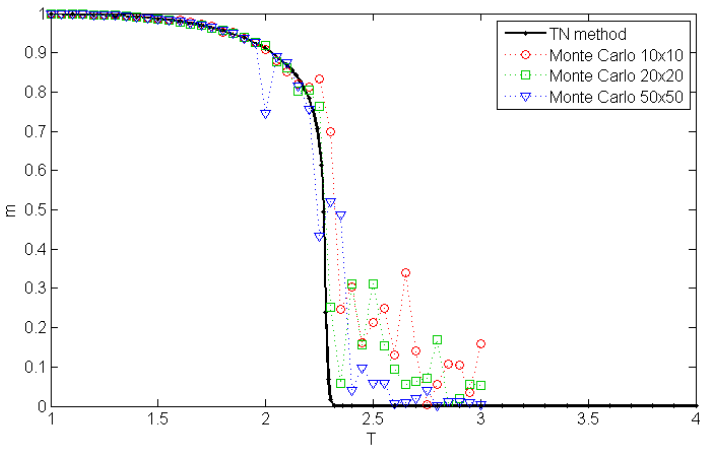


Figure 5.23: Magnetisation for the 2D Ising model resulting from TN and Monte Carlo methods.

[6]The fluctuations in the Monte Carlo simulations can be reduced by increasing the number of iterations or by increasing the lattice size. However, as the Metropolis-Hastings algorithm suffers from so-called critical slowing down around the phase transition, other approaches such as cluster algorithms can give better results there.

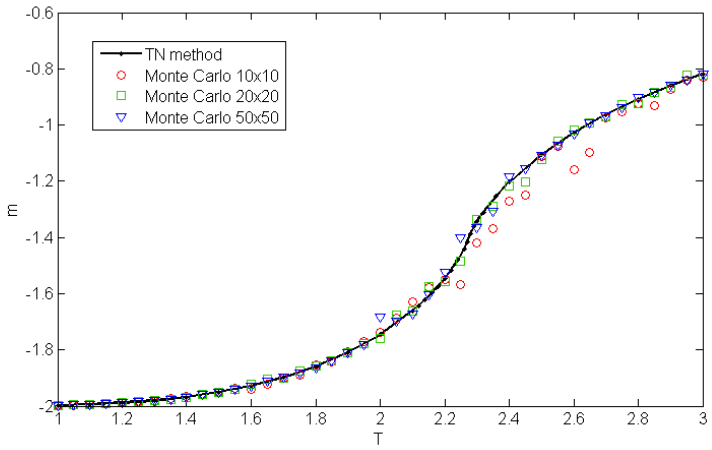


Figure 5.24: Free energy for the 2D Ising model resulting from TN and Monte Carlo methods.

Chapter 6

Numerical Results in 3D

After having discussed the 2D Ising model in the previous chapter, we now continue with the 3D case. In the first section, details on the calculations are given, similarly to the first two sections of Chapter 5. In the second section, the physical results themselves are given, and they are compared to a Monte Carlo simulation as well.

6.1 Calculations in 3D

When tackling the 3D Ising model, many of the concepts used in the 2D case can be transferred. In 2D, the tensors M of the TN were composed of four t matrices and one central δ-tensor. Now, in one extra dimension, M has six indices and is hence built up out of six t matrices and a δ. The calculation of the magnetisation is a straightforward generalisation of section 5.2.1 as well. The normalisation factor shown in figure (5.3) is replaced by the network drawn in figure (6.1). In that network, L, S and R are obtained by applying the corner method to the central tensor $E - M - E^*$. The magnetisation itself is given by the ratio of (1) the network of figure (6.1) with M_S (cf. equation (5.3)) instead of M, to (2) the original network of figure (6.1).
In order to access the free energy, we need the influence of adding one site again. The contraction

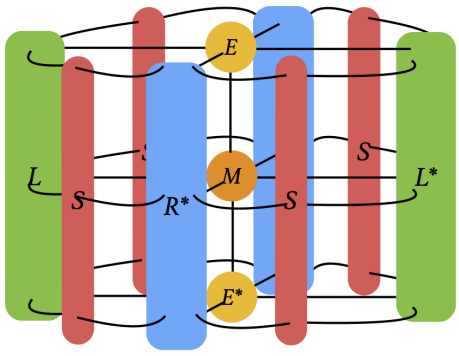


Figure 6.1: TN needed for the calculation of the magnetisation.

over the first, 'vertical', dimension is shown in figure (6.2). This has reduced the original three-dimensional network to a two-dimensional network with the appropriate normalisation factor. The next step is to reduce both two-dimensional networks so that, within them, only the influence of adding one site is retained. In similarity to the two-dimensional case as put forward in section 5.2.2, this requires the calculation of the 'free energy' of the resulting C- and c-networks. So

indeed, the free energy is finally obtained by $f = -T\ln(\sqrt{x})$ with x defined as in figure (3.15), where this quantity was used as the convergence measure in the three-dimensional contraction algorithm.

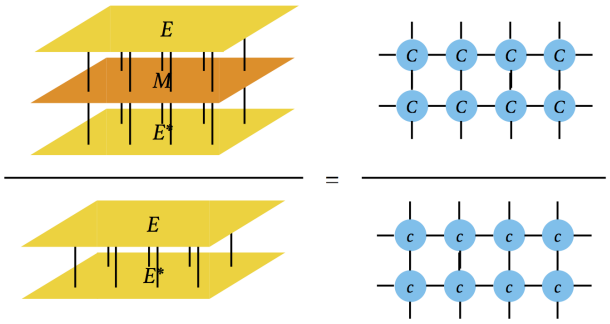


Figure 6.2: First contraction for the free energy in 3D.

6.2 Physical Results

6.2.1 2D Ising Model Using Projectors

As the algorithm for contraction in 3D involves applying the 2D corner method to the tensor $E - T - T^* - E^*$, the computational cost is high. Indeed, in 2D the corner method was used on a tensor of rank four with index dimension 2, whereas the index dimension of $E - T - T^* - E^*$ is $4D^2$ with D the bond dimension of the environment E. Increasing the bond dimension as much as in 2D will therefore be impossible. As this in itself sets a limit on the accuracy of the final result, we shall first apply the projector-based method to the 2D case[1]. This provides us with the assurance that the method works properly.
Before showing a result, a small remark on the truncation should be made. When optimising the projector, the dimension of the subspace on which it projects, is fixed. For 2D, this is a drawback compared to singular value induced truncation. In the 3D case, however, very limited increase in bond dimension is possible, so fixing it to a certain (low) value is a necessity anyway. Figure (6.3) shows, as an example, the magnetisation for the 2D Ising model. The graph contains three results: one obtained from the corner method with increasing the bond dimension, a second one from the corner method without an increase in the bond dimension. The third set of data are the results of the projector method, also without an increasing bond dimension. It is clearly visible that the projector method is less stable than the corner method, but nonetheless, there is an obvious agreement between the two methods.

6.2.2 The Anisotropic Ising Model

In the most general formulation of the Ising Hamiltonian, cf. equation (4.1), the interaction strength J_{ij} can depend on the lattice sites. Until now, we have alway considered the interaction to be constant, but it is interesting to add anisotropy to the interaction. Going back to the two-dimensional Ising model, we can take $J_x \neq J_y$ where x and y denote the two dimensions of the lattice. J_x is, as in the constant-J case, rescaled to unity, and J_y is allowed to vary in [0,1].

[1] The details of this algorithm were not given in Chapter 3, but it requires no great effort to derive them from the 3D algorithm.

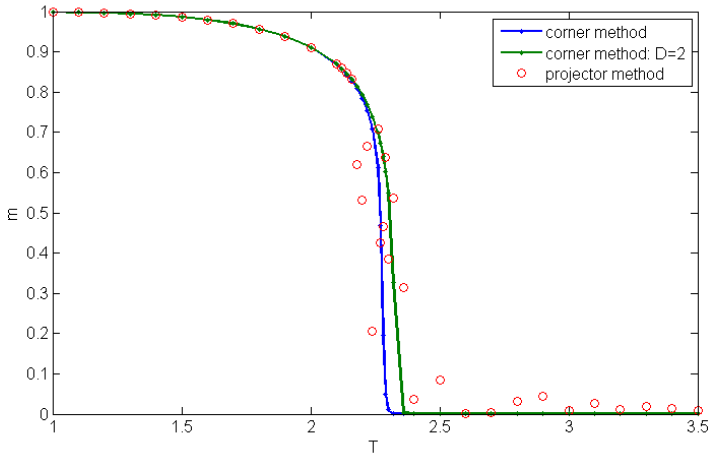


Figure 6.3: Comparison of the magnetisation as obtained (1) in the corner method, (2) in the corner method without increasing the bond dimension, and (3) in the projector method without an increase in the bond dimension.

Of course, it is very easy to implement this: it suffices to use t with a different J horizontally and vertically in the construction of the tensor M in figure (5.2). Because the corner method for network contraction works with As horizontally and Bs vertically, it can cope with anisotropy without further adaptations.

The Ising model can then be studied for varying J_y, just as several physical quantities were discussed for the isotropic case in section 5.2. However, that is not what interests us here. It is the special case $J_y = 0$ that is of interest here. Taking $J_y = 0$ means that instead of being properly two-dimensional, the lattice can better be thought of as an infinite collection of non-interacting linear chains. Hence, when performing the calculations, the results from the one-dimensional model should be obtained. An example for the free energy is shown in figure (6.4). It is straightforward to derive that the low-temperature limit for the free energy is now equal to $f = -J_x - J_y = -1 - J_y$, which agrees with the graph.

Studying the one-dimensional model using a method devised for use in two dimensions is of

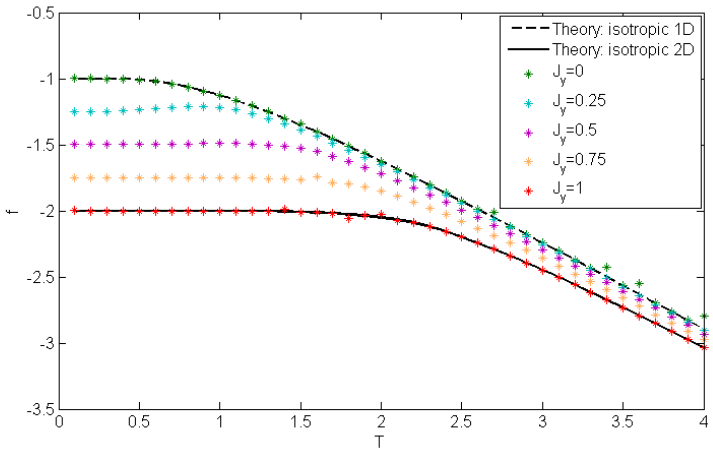


Figure 6.4: Free energy for the anisotropic Ising model in two dimensions.

course complete overkill. But if an analytical solution is not known in a certain dimension, this procedure allows us to verify the method in a lower dimension. This is precisely what we shall now do in three dimensions. By putting $J_x = J_y = 1$ and $J_z = 0$, we should get the results of

the two-dimensional case. As an example, this is shown for the magnetisation in figure (6.5).

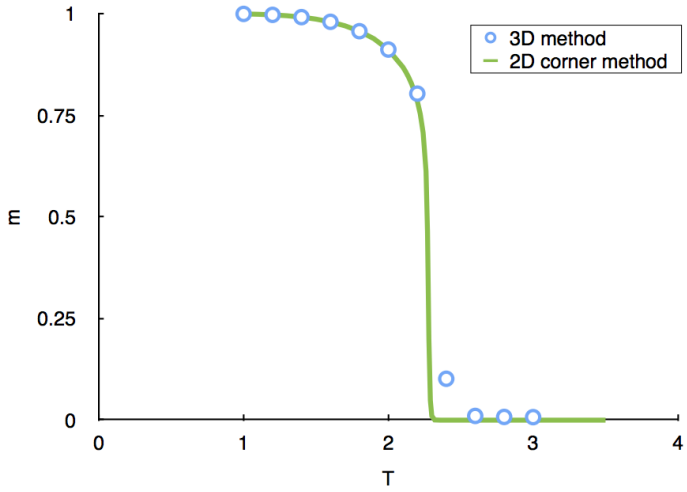


Figure 6.5: Magnetisation for the 2D model by studying the 3D model for $J_z = 0$.

6.2.3 Results for the 3D Ising Model Itself

Finally, we shall now discuss the results for the three-dimensional Ising model itself, both without an external field and with an external field. Just like in two dimensions, the presence of an external field makes the problem computationally easier. As described in section 6.1, the measure for convergence x leads to the free energy by $f = -T\ln(\sqrt{x})$. Inaccuracies can hereby be inflated and therefore $\sqrt{x}$ itself is also plotted here. In lower dimensions, this was not necessary because the simulations were more accurate and stable there.

Figures (6.6), (6.7) and (6.8) respectively show the magnetisation, free energy and $\sqrt{x}$ for a bond dimension equal to two. These results qualitatively fulfil the expectations and obey the right low- and high-temperature limits on the magnetisation and the free energy. For the free energy, results from 1D and 2D are added to the figure, as in figure (5.20). It was seen in that figure that the free energy of the 1D case with field h is slightly lower than the free energy for the 2D case with field $h-1$ (in the mid-temperature region). Now, we can also see how the free energy of the 3D case with field $h-2$ is still a little higher (by about the same amount as the difference between the 1D and 2D curves). It is by no means proven here that this should be the case, but it is an indication that good results are obtained for the free energy.

A shortcoming is the lack of a well-defined phase transition in the magnetisation when $h = 0$. In figure (6.6), a Monte Carlo simulation is included that does show a clear phase transition. The vertical line is positioned at the approximate critical temperature $T_c \approx 4.5$. The fact that the TN simulation does not show a clear phase transition is probably due to the low bond dimension, as a similar effect was observed in 2D (see figure (5.8) for the influence of the bond dimension on the phase transition).

When looking back at the contraction method as described in section 3.3.1, the isometries used in the truncation step are not necessarily equal in the two planar directions. However, for the Ising model, there is $\pi/2$ rotational symmetry and one could therefore force the horizontal and vertical isometries to be equal. When comparing this to the case where they are allowed to be different, it was found that more stable results were obtained in the latter case. Indeed, when relaxing the rotational symmetry, more degrees of freedom are available to improve the

truncation step.
It is clear from these results for the magnetisation that there is room for improvement, in spite of the qualitatively good results. Further research should shed light on how to increase the accuracy of the 3D method without increasing the computational cost to unattainable heights.

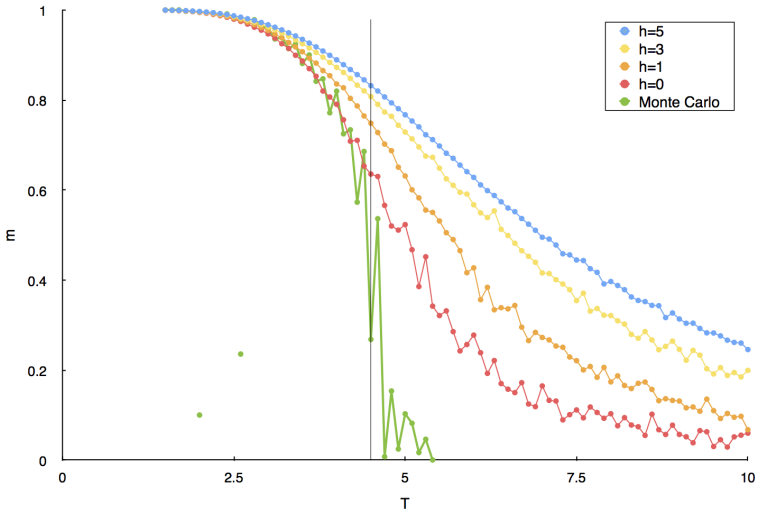


Figure 6.6: Magnetisation for the 3D Ising model for various external fields h. A Monte Carlo simulation ($h = 0$) is also included, as is a vertical line at the position $T_c \approx 4.5$.

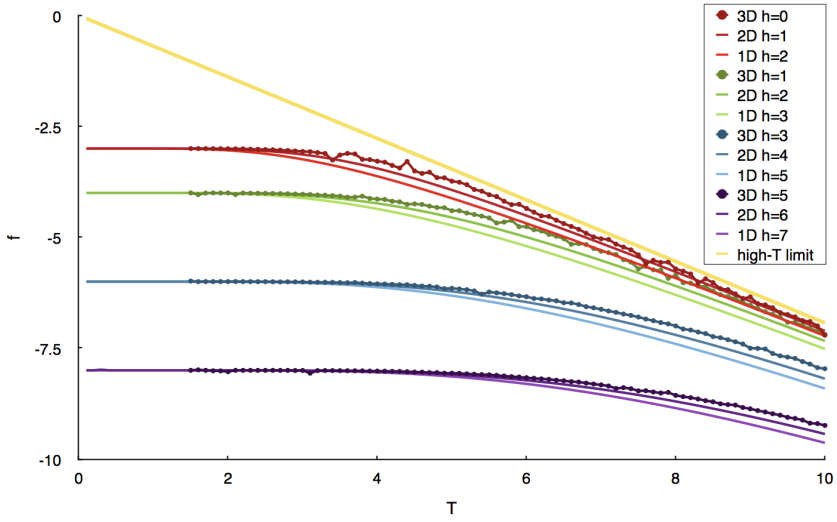


Figure 6.7: Free energy for the 1D, 2D and 3D Ising models for various external fields h.

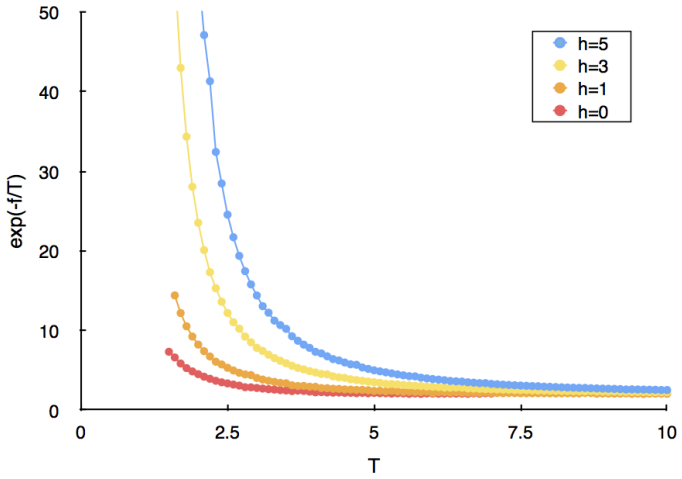


Figure 6.8: $\sqrt{x} = e^{-f/T}$ for the 3D Ising model for various external fields h.

Chapter 7

Conclusions

Now that we have come to the end of this dissertation, this chapter provides the reader with a concise non-technical overview of the past chapters and a few concluding remarks.

In this dissertation, a classical system, i.e. the classical Ising model, was examined. The tensor network formalism was used to address this model. This formalism springs from quantum many-body physics, where it is used in classical simulations of quantum systems. These simulation methods in quantum physics find their way into classical physics via quantum-classical mapping. The elements from quantum many-body physics and the tensor netwerk formalism that are needed for the understanding of this dissertation were given in Chapter 2.

The reason for choosing the Ising model is that it behaves in a physically interesting way despite its very simple Hamiltonian (e.g. there is a phase transition in two and higher dimensions). As such, it is ubiquitous in statistical physics and the numerical results obtained in this work can be compared to the expansive literature on the Ising model. Some theoretical elements were therefore presented in Chapter 4.

In Chapters 5 and 6 the results of the simulations for, respectively, the two-dimensional and three-dimensional Ising model were presented and discussed. It was found that these results concur with the theoretical results, certainly in two dimensions. In three dimensions, good results were obtained for the free energy. However, as an increase in the bond dimension is very restricted due to the high computational cost of the 3D algorithm, the magnetisation did not show a well-defined phase transition in the absence of an external field. Further improvements should be made in order to remedy this.

Of course, in the application of the methods described in Chapter 3, we need not limit ourselves to the Ising model. It should be interesting to explore the performance of the computational methods when applied to other physical systems, be they of classical or quantum-mechanical nature.

Appendix A

Scaling of QR-decomposition

In this appendix, we give an estimate for the number of operations needed to perform a thin QR-decomposition of an $m \times n$-matrix. As we do not need Q explicitly, we only consider at the steps needed to find R.
Here, we analyse a QR-algorithm based on Householder transformations. There are other possible approaches, but with this algorithm it is found that QR-decomposition is not computationally more demanding than the RA step in the single-layer contraction algorithm. Possible more efficient QR-algorithms would a fortiori comply with this observation, thus not changing the overall computational cost of the single-layer contraction algorithm.

Householder transformations are unitary linear transformations that perform reflections through hyperplanes through the origin. If n is a unit vector orthogonal to the hyperplane, the reflection of a vector x through the hyperplane is given by

$$x - 2\langle x, n\rangle n \tag{A.1}$$

In our case, we can look at the first column of $A \in \mathbb{K}^{m\times n}$ as a vector. By applying a suitable Householder transformation, this vector can be projected onto the basis vector $e_1 = (1, 0, \ldots, 0)^T$. The hyperplane corresponding to this projection is a bisection plane of e_1 and x, the non-normalised normal of which is given by $x - \|x\|e_1$. Next, the same procedure can be repeated on the submatrix of the new A where the first row and column are discarded, and so on until the new A is upper-triangular (this takes $\min(m-1, n)$ steps). This is R in $A = QR$, and all the unitary Householder matrices together form Q.

In order to determine the scaling of a QR-decomposition, we want to determine the number of operations needed in the following algorithm.

Algorithm 1 QR-decomposition using Householder transformations

```
[m,n]=size(A)
1: for i=1:min(m-1,n) do
2: x=A(i:m,i)                                   ▷ vector to be projected
3: N=x
4: N(1)=N(1)-norm(x)
5: N=N/norm(N)                                  ▷ normal to the hyperplane
6: A(i:m,i:n)=A(i:m,i:n)-2*N*N'*A(i:m,i:n)      ▷ actual Householder transformation
7: end for
```

Let us denote the size of the submatrix of A under consideration in the i^{th} step by $k \times l$ with $k = m - i + 1$ and $l = n - i + 1$.

1. The construction of N in line 4 requires the calculation of the norm of a k-dimensional vector and one subtraction. The norm consists of a square root, k multiplications and $k - 1$ additions. Let us count the square root as one operation, even though it is computationally more demanding than an addition or a multiplication (but even when counting for a square root a number of operations up to the order kl, our final estimate will not change). The norm requires $2k$ operations in total.

2. In line 5, another norm is calculated ($2k$ operations), and k divisions are needed.

3. l inner products of length-k vectors need to be calculated in line 6, requiring $l(2k - 1)$ operations. k more operations are needed for the scalar multiplication of 2 with N. Every inner product also needs to be multiplied with N (kl operations) and finally subtracted (again, kl operations).

The total number of operations per iteration step is $4kl + 6k - l + 1$, in which the term $4kl$ will dominate. The sum over i yields

$$\sum_{i=1}^{n} 4kl = \sum_{i=1}^{n} 4(m - i + 1)(n - i + 1) \approx 2mn^2 - \frac{2n^3}{3} \tag{A.2}$$

where it was assumed that $n \leq m - 1$. In this result, the two third-order terms are shown, but the dominant one will be the first one in our case.
In the single-layer contraction algorithm, A is a $dD \times D$-matrix, so the final scaling is $O(dD^3)$.

Appendix B

Transfer Matrix in the Presence of an External Field

B.1 Square Root

The transfer matrix T for the Ising hamiltonian in the presence of an external field is given by

$$T = \begin{bmatrix} \exp(\beta J + \beta h) & \exp(-\beta J) \\ \exp(-\beta J) & \exp(\beta J - \beta h) \end{bmatrix} \tag{B.1}$$

In order to have a symmetrical TN, we need a square root of T. In the first place, we check whether T is positive definite and whether consequently, there exists a unique positive-definite square root t. According to Sylvester's criterion, a necessary and sufficient condition for T to be positive definite is that all its leading principal minors are positive[1].
Therefore, $\exp(\beta J + \beta h)$ and $\exp(2\beta J) - \exp(-2\beta J)$ should be positive, which is indeed the case. As $\beta \to \infty$, the second term becomes zero. It is tempting to assert that in that case, T is positive-semidefinite. In this case, that is true, but in Sylvester's criterion, one needs to check if all principal minors are non-negative, i.e. also $\exp(\beta J - \beta h) \geq 0$.
Defining the square root t as

$$t = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \tag{B.2}$$

it is easily verifiable through direct calculation of t^2 that $t_{12} = t_{21}$. Opposed to the $h = 0$ case, however, $t_{11} \neq t_{22}$. The expressions for the components of t are lengthy and were calculated using Maple[2]. By putting

$$x = \exp(4\beta J + 4\beta h) - 2\exp(4\beta J + 2\beta h) + \exp(4\beta J) + 4\exp(2\beta h) \tag{B.3}$$

and

$$y = \exp(2\beta J) + \exp(2\beta J + 2\beta h) \tag{B.4}$$

[1] As a reminder: 'principal' means that the same rows as columns are deleted and 'leading' means that the last several rows and columns are deleted

[2] It is therefore very well possible that the expressions may be simplified.

we get

$$\begin{aligned} t_{11} &= \frac{\sqrt{2}}{4}\frac{1}{\sqrt{x}}\Bigg[\exp\left(-\frac{\beta J}{2}-\frac{\beta h}{2}\right)\left(\sqrt{y-\sqrt{x}}+\sqrt{y+\sqrt{x}}\right)\sqrt{x} \\ &\quad +\exp\left(\frac{3\beta J}{2}+\frac{3\beta h}{2}\right)\left(\sqrt{y+\sqrt{x}}-\sqrt{y-\sqrt{x}}\right) \\ &\quad +\exp\left(\frac{3\beta J}{2}-\frac{\beta h}{2}\right)\left(\sqrt{y-\sqrt{x}}-\sqrt{y+\sqrt{x}}\right)\Bigg] \end{aligned} \tag{B.5}$$

$$t_{12}=t_{21} = \frac{\sqrt{2}}{2}\exp\left(-\frac{\beta J}{2}+\frac{\beta h}{2}\right)\frac{\sqrt{y+\sqrt{x}}-\sqrt{y-\sqrt{x}}}{\sqrt{x}} \tag{B.6}$$

$$\begin{aligned} t_{22} &= \frac{\sqrt{2}}{4}\frac{1}{\sqrt{x}}\Bigg[\exp\left(-\frac{\beta J}{2}-\frac{\beta h}{2}\right)\left(\sqrt{y-\sqrt{x}}\sqrt{x}+\sqrt{y+\sqrt{x}}\sqrt{x}\right) \\ &\quad +\exp\left(\frac{3\beta J}{2}+\frac{3\beta h}{2}\right)\left(\sqrt{y-\sqrt{x}}-\sqrt{y+\sqrt{x}}\right) \\ &\quad +\exp\left(\frac{3\beta J}{2}-\frac{\beta h}{2}\right)\left(\sqrt{y+\sqrt{x}}-\sqrt{y-\sqrt{x}}\right)\Bigg] \end{aligned} \tag{B.7}$$

The zero-field expression of equation (5.2) can easily be retrieved. When $h = 0$, the last two terms of t_{11} and t_{22} cancel each other. In the remaining terms, $\sqrt{y \pm \sqrt{x}}$ can be rewritten as a term proportional to $\sqrt{\cosh(\beta J)}$ or $\sqrt{\sinh(\beta J)}$. All the remaining factors cancel up to the overall factor $\frac{1}{\sqrt{2}}$.

B.2 Rescaling of the External Field

The discussion up till now was mathematically correct, but equation (B.1) is in fact *only in 1D* the right transfer matrix. The Ising Hamiltonian (4.1) consists of interaction terms - the J-terms - and one-site terms - the h-terms. In the TN description, a matrix T is placed on each bond index, in the first instance to describe the interaction between neighbouring sites.
If one wants to add an external field, it is straightforward to see that terms similar to $\exp(\pm\beta h)$ need to be added to the diagonal of T. Indeed, this was shown for the one-dimensional case in section 4.2.3. However, care should be taken that exactly one unit h of energy is assigned to every site. In 1D, there are as many Ts as there are sites (ignoring potential variations at the boundaries), so each T needs $\exp(\pm\beta h)$.
When going to 2D, there are twice as many Ts as sites, so each T is only responsible for an energy amount of $\frac{h}{2}$. Therefore, in the previous results, h needs to be replaced by $\frac{h}{2}$ for the 2D case. Similarly, in 3D $h \to \frac{h}{3}$ because there are six nearest-neighbours per site.

Bibliography

[1] R. J. Baxter. *Exactly Solved Models in Statistical Mechanics*. Academic Press, 1982.

[2] K. Binder and E. Luijten. Monte Carlo tests of Renormalization-Group predictions for critical phenomena in Ising models. *Physics Reports*, 344:179–253, 2001.

[3] C. Bonati. The Peierls argument for the higher dimensional Ising models. *European Journal of Physics*, 35(3):035002, 2014.

[4] S. Bornholdt and F. Wagner. Stability of money: phase transitions in an Ising economy. *Physica A: Statistical Mechanics and its Applications*, 316(1-4):453–468, 2002.

[5] P. Calabrese and J. Cardy. Entanglement entropy and quantum field theory. *Journal of Statistical Mechanics: Theory and Experiment*, page 06002, 2004.

[6] J. I. Cirac. Entanglement in many-body quantum systems. *arXiv:1205.3742 [quant-ph]*, 2012.

[7] N. Van den Bergh. Wiskundige ingenieurstechnieken. *Lecture notes UGent*, 2010.

[8] R. L. Dobrushin. Existence of a phase transition in two-dimensional and three-dimensional Ising models. *Theory of Probability and its Applications*, 10(2):193–213, 1965.

[9] J. Eisert. Entanglement and tensor network states. *arXiv:1308.3318v2 [quant-ph]*, 2013.

[10] A. M. Ferrenberg and D. P. Landau. Critical behavior of the three-dimensional Ising model: A high-resolution Monte Carlo study. *Physical Review B*, 44(10):5081–5091, 1991.

[11] M. E. Fisher. Critical temperatures of anisotropic Ising lattices. II. General upper bounds. *Physical Review*, 162(2):480–485, 1967.

[12] M. E. Fisher and M. F. Sykes. Excluded-volume problem and the Ising model of ferromagnetism. *Physical Review*, 114(1):45–58, 1959.

[13] J. Fröhlich, B. Simon, and T. Spencer. Infrared bounds, phase transitions and continuous symmetry breaking. *Communications in Mathematical Physics*, 50(1):79–95, 1976.

[14] G. Galilei. *Dialogo sopra i due Massimi Sistemi del Mondo Tolemaico e Copernicano*. 1632.

[15] A. García-Sáez and J. I. Latorre. Renormalization Group contraction of tensor networks in three dimensions. *Physical Review B*, 87(8):085130, 2013.

[16] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins University Press, 3rd edition, 1996.

[17] R. B. Griffiths. Correlations in Ising ferromagnets. III. A mean-field bound for binary correlations. *Communications in Mathematical Physics*, 6:121–127, 1967.

[18] J. Haegeman. Variational Renormalization Group methods for extended quantum systems. *PhD Thesis*, 2011.

[19] M. B. Hastings. Solving gapped Hamiltonians locally. *Physical Review B*, 73(8):085115, 2006.

[20] M. B. Hastings. An area law for one-dimensional quantum systems. *Journal of Statistical Mechanics: Theory and Experiment*, page 08024, 2007.

[21] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[22] A. Irbäck, C. Peterson, and F. Potthast. Evidence for nonrandom hydrophobicity structures in protein chains. *Proceedings of the National Academy of Sciences, USA*, 93:9533–9538, 1996.

[23] E. Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik*, 1925.

[24] J. Jordan, R. Orús, G. Vidal, F. Verstraete, and J. I. Cirac. Classical simulation of infinite-size quantum lattice systems in two spatial dimensions. *Physical Review Letters*, 101(25):250602, 2008.

[25] H. G. Katzgraber. Introduction to Monte Carlo methods. *arXiv:0905.1629v3 [cond-mat.stat-mech]*, 2009.

[26] H. A. Kramers and G. H. Wannier. Statistics of the two-dimensional ferromagnet. Part I. *Physical Review*, 60(3):252–262, 1941.

[27] M. Lubasch, J. I. Cirac, and M.-C. Bañuls. Unifying Projected Entangled Pair States contractions. *New Journal of Physics*, 16(3):033014, 2014.

[28] J. Majewski, H. Li, and J. Ott. The Ising model in physics and statistical genetics. *American Journal of Human Genetics*, 69(4):853–862, 2001.

[29] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1092, 1953.

[30] N. Metropolis and S. Ulam. The Monte Carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.

[31] V. Murg, J. I. Cirac, B. Pirvu, and F. Verstraete. Matrix Product Operator representations. *New Journal of Physics*, 12(2):025012, 2009.

[32] L. Onsager. Crystal statistics. I. A two-dimensional model with an order-disorder transition. *Physical Review*, 65(3-4):117–149, 1944.

[33] R. Orús. Exploring corner transfer matrices and corner tensors for the classical simulation of quantum lattice systems. *Physical Review B*, 85(20):205117, 2012.

[34] R. Orús. A practical introduction to tensor networks: Matrix Product States and Projected Entangled Pair States. *arXiv:1306.2164v2 [cond-mat.str-el]*, 2013.

[35] R. Orús and G. Vidal. The iTEBD algorithm beyond unitary evolution. *Physical Review B*, 78(15):155117, 2008.

[36] R. Orús and G. Vidal. Simulation of two dimensional quantum systems on an infinite lattice revisited: corner transfer matrix for tensor contraction. *Physical Review B*, 80(9):094403, 2009.

[37] R. Peierls. On Ising's model of ferromagnetism. *Mathematical Proceedings of the Cambridge Philosophical Society*, 32(3):477–481, 1936.

[38] D. Pérez-García, F. Verstraete, J. I. Cirac, and M. M. Wolf. PEPS as unique ground states of local Hamiltonians. *Quantum Information and Computation*, 8:0650–0663, 2008.

[39] D. Pérez-García, F. Verstraete, M. M. Wolf, and J. I. Cirac. Matrix Product State representations. *Quantum Information and Computation*, 7(5):401–430, 2007.

[40] T. Preis, P. Virnau, W. Paul, and J. J. Schneider. GPU accelerated Monte Carlo simulation of the 2D and 3D Ising model. *Journal of Computational Physics*, 228(12):4468–4477, 2009.

[41] N. Schuch. Condensed matter applications of entanglement theory. *arXiv:1306.5551 [quant-ph]*, 2013.

[42] N. Schuch and J. I. Cirac. Matrix Product State and mean field solutions for one-dimensional systems can be found efficiently. *Physical Review A*, 82(1):012314, 2010.

[43] T. D. Schultz, D. C. Mattis, and E. H. Lieb. Two-dimensional Ising model as a soluble problem of many fermions. *Review of Modern Physics*, 36(3):856–871, 1964.

[44] A. L. Talapov and H. W. J. Blöte. The magnetization of the 3D Ising model. *Journal of Physics A: Mathematical and General*, 29(17):5727, 1996.

[45] F. Verstraete and J. I. Cirac. Renormalization algorithms for quantum-many body systems in two and higher dimensions. *arXiv:cond-mat/0407066v1 [cond-mat.str-el]*, 2004.

[46] F. Verstraete and J. I. Cirac. Matrix Product States represent ground states faithfully. *Physical Review B*, 73(9):094423, 2006.

[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. *Advances in Physics*, 57(2):143–224, 2008.

[48] F. Verstraete, D. Porras, and J. I. Cirac. Density Matrix Renormalization Group and periodic boundary conditions: a quantum information perspective. *Physical Review Letters*, 93(22):227205, 2004.

[49] F. Verstraete, M. M. Wolf, D. Perez-Garcia, and J. I. Cirac. Criticality, the area law, and the computational power of PEPS. *Physical Review Letters*, 96(22):220601, 2006.

[50] J. O. Vigfusson. Upper bound on the critical temperature in the 3D Ising model. *Journal of Physics A: Mathematical and General*, 18(17):3417, 1985.

[51] J.-C. Walter and G. Barkema. An introduction to Monte Carlo methods. *arXiv:1404.0209v1 [cond-mat.stat-mech]*, 2014.

[52] M. M. Wolf, F. Verstraete, M. B. Hastings, and J. I. Cirac. Area laws in quantum systems: mutual information and correlations. *Physical Review Letters*, 100(7):070502, 2008.

[53] C. N. Yang. The spontaneous magnetization of a two-dimensional Ising model. *Physical Review*, 85(5):808–816, 1952.