Simulating classical spin systems using the Fixed Point Corner Method

by

RAYMOND VAN DER WERFF

10531688

July 25, 2018

60 ECTS
Research carried out between:
01-09-2017 and 01-07-2018

Supervisor: Dr. Philippe Corboz
Assessor: Dr. Edan Lerner

INSTITUTE FOR THEORETICAL PHYSICS
Abstract

We use the Corner Transfer Matrix Renormalization Group (CTMRG) method and the Fixed Point Corner Method (FPCM) in order to simulate the partition function of the two-dimensional square-lattice Ising model. Thermodynamic quantities, like the magnetization per site, free energy per site and correlation length, are calculated with both methods. A finite bond dimension scaling analysis is performed in order to determine the critical temperature. We use the exact solutions of the Ising model to compare the accuracy and speed of both methods. It is concluded that FPCM suffers less from finite-size effects and is faster than CTMRG near the critical temperature.

Furthermore, we use FPCM to investigate the $q$-state clock model for $q = 5$ and $q = 6$. For both choices of $q$, we find a phase diagram with an ordered, massless and disordered phase, that are separated by two transition temperatures. We show that the transitions are of Berezinskii-Kosterlitz-Thouless type, and use a finite bond dimension scaling analysis and correlation length extrapolation scheme in order to determine the values of the transition temperatures.
Acknowledgements

Firstly, I want to thank my supervisor Philippe Corboz for helping me during the project. You always gave good suggestions and feedback, which were very useful for understanding the theory and getting to these results.

Then I want to thank my second assessor Edan Lerner, for taking the time to look at my presentation, read my thesis and perform an assessment.

Also thanks to Ido Niesen, Alex van der Werff and Patrick Vlaar, who helped me to learn how to use Linux in order to run simulations on the computer on science park. With the use of my computer only the results would have never been this great.

Finally, I want to thank my friends Jasper Dingerink, Maarten Hammer, Danny Wernik, Patrick Verhagen and Melvin van den Bout for supporting me not only during this project, but during my whole bachelor and master trajectory.
Contents

1 Introduction 5

2 The tensor network representation of quantum states 7
  2.1 Diagrammatic notation 4
  2.2 Matrix product state 8
    2.2.1 Left/right/mixed canonical form 10
    2.2.2 Uniform matrix product state 11

3 The tensor network representation of the partition function 13
  3.1 Tensor network representation 14
  3.2 Relation to thermodynamic variables 16
    3.2.1 Local operators 16
    3.2.2 Correlation length 17
  3.3 The ansatz for the partition function 18
    3.3.1 Calculation of thermodynamic variables 18

4 Two methods to calculate the partition function 20
  4.1 The Corner Transfer Matrix Renormalization Group Method 20
    4.1.1 Growth scheme 20
    4.1.2 Renormalization scheme 21
    4.1.3 Overview CTMRG 22
  4.2 The Fixed Point Corner Method 24
    4.2.1 Left orthonormalization scheme 25
  4.3 Technical details 27
    4.3.1 Starting environment tensors 27
    4.3.2 Convergence criterium 29
    4.3.3 Determination of fixed points with FPCM 29

5 Results for the Ising model 32
  5.1 Definition of the Ising model 32
    5.1.1 Tensor network representation 33
    5.1.2 Phase transition 34
    5.1.3 Exact solutions 35
  5.2 Analytic results 35
    5.2.1 Singular value spectrum 36
    5.2.2 Magnetization 37
    5.2.3 Correlation length 42
    5.2.4 Determination of critical temperature by finite-χ scaling 43
6 Results for the 6-state clock model

6.1 Definition of the q-state clock model

6.1.1 Tensor network representation

6.1.2 The \( q \to 2 \) and \( q \to \infty \) limits

6.1.3 The Berezinskii-Kosterlitz-Thouless transition

6.1.4 Phase diagram

6.1.5 Transition temperatures from literature

6.2 Analytic results

6.2.1 Singular value spectrum

6.2.2 Magnetization

6.2.3 Correlation length

6.2.4 Determination of transition temperatures by finite-\( \chi \) scaling

6.2.5 Determination of transition temperatures by \( \xi \)-extrapolation

7 Results for the 5-state clock model

7.1 Definition of the 5-state clock model

7.1.1 Tensor network representation

7.1.2 Transition temperatures from literature

7.2 Analytic results

7.2.1 Singular value spectrum

7.2.2 Magnetization and correlation length

7.2.3 Determination of transition temperatures by finite-\( \chi \) scaling

7.2.4 Determination of transition temperatures by \( \xi \)-extrapolation

8 Conclusions
1 Introduction

Many-body systems are very interesting, as simple interactions between particles can result in interesting phases of matter. Exotic phenomena can emerge, such as magnetism and superconductivity. Changing the temperature can lead to phase transitions, like the forming of ice or the melting of iron [1]. To understand phases and phase transitions, we would like to simulate those systems and get numerical solutions. But, as the number of degrees of freedom of many-body systems grows exponentially with the system size, it is in most cases impossible to get those solutions. Tensor networks can be used to overcome this problem, and the use of them as a tool in the field of many-body physics has grown tremendously over the last decades [2]. Along with renormalization techniques, tensor networks can very efficiently approximate systems with infinitely many degrees of freedom, by keeping only the most relevant states. For example, ground states of one-dimensional quantum systems can be represented by Matrix Product States (MPS’s) [3]. These can be optimized by the Density Matrix Renormalization Group (DMRG) method [4], introduced by White in 1992 [5].

Tensor networks have also been useful in representing partition functions of classical spin systems. In 1995, Nishino and Okunishi formulated the Corner Transfer Matrix Renormalization Group (CTMRG) method [6], which was based on DMRG, and could be used to approximately represent partition functions with high accuracy. This approximation can be controlled by the bond dimension of the tensors in the tensor network, which represents the number of relevant states that are kept. Using the same ansatz as for CTMRG, Fishman et al. recently formulated another method to find the representation of the partition function in the thermodynamic limit [7], which they call the Fixed Point Corner Method (FPCM).

The goal of this thesis is to get results for several two-dimensional square-lattice classical spin systems, using FPCM, and see how they compare with results obtained with CTMRG. Before we apply the methods, I first give some theory about tensor networks and how we can efficiently represent them in diagrammatic notation. Then I describe how tensor networks can be used to represent quantum ground states and partition functions of classical spin systems. For the latter, I give some background of statistical mechanics, which is needed in order to describe emergent phenomena in many-body systems. Then I describe how CTMRG and FPCM can be used to calculate the partition function, and apply them to several classical spin systems.
The first model that we study is the Ising model [8]. Despite its simplicity, this model undergoes a phase transition at a critical temperature $T_c$, from a phase in which the spins are ordered to a phase in which the spins are disordered. This makes the Ising model very interesting, as it can be used to understand magnetism. Because exact solutions are known [9], this model is a good candidate to test and compare both methods. We calculate several thermodynamic quantities, determine the value of $T_c$, and look at the accuracy and speed of both methods.

After investigating the Ising model, we turn to the more general $q$-state clock model. Instead of the $q = 2$ possible spin directions in the Ising model, each spin in the clock model can point in $q$ possible directions. The origin of this model goes back to the formulation of the Potts model by Domb [10]. It happens that when $q$ is bigger or equal than some $q_c$, the system cannot only be in the ordered and disordered phase, but also in an intervening massless phase. When going to the massless phase, the system undergoes an interesting Berezinskii-Kosterlitz-Thouless (BKT) transition [11, 12]. The discovery of this transition led to the Nobel Prize in Physics in 2016 [13]. Instead of one critical temperature, the massless phase describes a finite critical region, which is bounded by transition temperatures $T_1$ and $T_2$. The singularities in thermodynamic quantities makes the BKT-transition difficult to study [14].

In literature, there is some discussion about the value of $q_c$ [15], which is believed to be either 5 or 6. We first investigate the 6-state clock model, where it is commonly agreed that the BKT-transitions take place. We calculate several thermodynamic quantities in order to show the presence of the three different phases, and show that the phase transitions are of BKT-type. We also determine the values of $T_1$ and $T_2$ by using a scaling analysis in the bond dimension, and by a recently designed extrapolation scheme of the correlation length [16]. We then use the same analyses for the 5-state clock model in order to see if a massless phase is present, and if the phase transitions are of BKT-type.
2 The tensor network representation of quantum states

Tensors are multi-dimensional arrays, which are generalizations of vectors and matrices. The number of indices that is needed to pick an element of the tensor, is called the rank of the tensor. Thus, a scalar, vector and matrix are a rank-0, rank-1 and rank-2 tensor respectively. Each tensor index has a certain dimension, which is the number of values the index can take. Just like matrices can be multiplied together to form another matrix, tensors can be multiplied together to form another tensor. Tensors are very useful in physics, as they, for example, can represent ground states of quantum many-body systems or partition functions of classical systems. In order to show this, I will first introduce an efficient diagrammatic notation to visualize tensors and tensor networks. The information in this chapter is taken from Ref. [2].

2.1 Diagrammatic notation

A rank-$n$ tensor can be visualized by a shape with $n$ lines attached to it. These lines are also called legs. Each leg represents an index of the tensor, and has the dimension of that index. As depicted in Fig. 1a, a scalar is then represented as a ball without any legs, a vector as a ball with one leg, and a matrix as a ball with two legs. Multiple legs can be regrouped to form a new leg, which has then the dimension equal to the product of the dimensions of the regrouped legs, as shown in Fig. 1b. When two tensors have a leg with the same dimension, these legs can be connected to each other, which means that you sum over the corresponding indices. Performing this sum is also called tensor multiplication or contraction of indices, and is shown in Fig. 1c. The legs that do not connect tensors are called open legs. Multiple tensors can be connected to each other, even in a very complicated way, in order to form a tensor network. By multiplying all tensors along the connected legs, it is possible to contract the tensor network to a single tensor, as shown in Fig. 1d. This remaining tensor has the same open legs as the original tensor network. We see that, opposed to the use of summation symbols and index notation that give long and complicated expressions, the diagrammatic notation gives us a very clear overview of how to contract tensors in a tensor network.
Figure 1: Diagrammatic notation of a tensor network. (a) Diagram of a scalar, vector, matrix and a rank-3 tensor. (b) Diagram of a grouping of indices. (c) Diagram of a multiplication of tensors, or equivalently, a contraction of indices. (d) Diagram of the contraction of a tensor network.

2.2 Matrix product state

Tensor networks can be very useful for representing ground states of quantum systems. Take, for example, a one-dimensional lattice, consisting of \( N \) sites. If site \( i \) is described by a local Hilbert space \( \mathcal{H}_i \), then the total Hilbert space is given by \( \mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \cdots \otimes \mathcal{H}_N \).

If we consider a spin-1/2 system where each site can either be in state up \( (s_i = |\uparrow\rangle) \) or down \( (s_i = |\downarrow\rangle) \), then the dimension of \( \mathcal{H}_i \) equals 2 and the dimension of \( \mathcal{H} \) equals \( 2^N \).

The ground state of the system is in general given by

\[
|\psi\rangle = \sum_{s_1, s_2, \ldots, s_N} \psi_{s_1, s_2, \ldots, s_N} |s_1 \otimes s_2 \otimes \cdots \otimes s_N\rangle, \tag{1}
\]

where the sum goes over the possible spin states of each site; \( s_i \in \{ |\uparrow\rangle, |\downarrow\rangle \} \). The coefficients of the states are stored in the rank-\( N \) tensor \( \psi_{s_1, s_2, \ldots, s_N} \). The number of elements of this tensor is \( 2^N \), so this number grows exponentially with the system size. This is a major problem when we want to study many-body systems, where \( N \to \infty \). To solve this problem, we need a way to approximate this coefficient tensor, which can be done by introducing Matrix Product States (MPS’s).

In this ansatz, the big tensor \( \psi_{s_1, s_2, \ldots, s_N} \) is replaced by a product of tensors, where each site has its own tensor. This decomposition can be done in different ways. In Fig. 2a, we see a decomposition for the case that the lattice consists of four sites \( (N = 4) \) with open boundary conditions. The boundary tensors have two indices, so these are matrices.
The tensors in the bulk are rank-3 tensors. The open legs in this network still have the dimension of the local Hilbert space, which is called the physical dimension and is denoted here by \( q \). In the spin-1/2 system we have \( q = 2 \). The legs that connect the tensors have the dimension of an auxiliary space. I call this dimension the bond dimension, and denote it by \( \chi \). We can control the accuracy of the approximation by changing \( \chi \). The calculation of a coefficient of a particular state boils down to a taking a product of matrices with vectors on the ends, as shown in Fig. 2c. This is why the ansatz for the ground state is called a matrix product state.

Figure 2: The matrix product state ansatz for the ground state of a one-dimensional four-site quantum system. The big tensor \( \psi \) is decomposed into four tensors; one for each site. This is displayed in diagrammatic notation (a) and index notation (b). The accuracy of the approximation is controlled by the bond dimension \( \chi \) of the legs connecting the tensors. Calculating the coefficient of a particular state boils down to taking a matrix product (c).

The number of elements of the boundary and bulk tensors are \( q\chi \) and \( q\chi^2 \) respectively. The advantage of the elements of the boundary and bulk tensors is that the number of parameters involved is now polynomial in the bond dimension \( \chi \) and system size \( N \), instead of exponential in \( N \). So the MPS is a very efficient representation. Now we have to think about how big \( \chi \) needs to be in order to accurately represent the ground state. It is known that \( \chi \) should be proportional to \( 2^N \) in order to describe a random state in the Hilbert space accurately, so at first glance it seems that we did not solve the problem. But we are in most cases only interested in ground states, and these states are special. These states are much less entangled than a random state, and need states from a tiny corner of the Hilbert space in order to describe them accurately. This means that \( \chi \) does not have to increase exponentially with \( N \), but at most only polynomially, which solves our problem.
2.2.1 Left/right/mixed canonical form

A useful method in order to decompose the initial tensor into a matrix product state is to use a singular value decomposition (SVD). Performing an SVD on a matrix \( A \) means that this matrix is written as the following unique expression: \( A = U \cdot s \cdot V^\dagger \), where \( s \) is a diagonal matrix containing the singular values of \( A \) in descending order on the diagonal, and \( U \) and \( V \) are unitary matrices, i.e. \( U \cdot U^\dagger = V \cdot V^\dagger = 1 \). The singular value matrix \( s \) can be absorbed either in \( U \) or \( V^\dagger \) in order to decompose \( A \) into two separate tensors.

An SVD can also be performed on a higher-rank tensor by first regrouping the legs such that the tensor is reshaped into a matrix. After performing the SVD, the open legs of \( U \) and \( V^\dagger \) can be reshaped back to the size of the open legs of the original tensor. In Fig. 3, I show how repeated SVD’s can be used in order to get a matrix product state for a four-site quantum state with open boundary conditions. The bond dimension of the MPS can be controlled in this process by keeping only the largest \( \chi \) singular values in each SVD, which corresponds to keeping the first \( \chi \) columns of \( U \) and the first \( \chi \) rows of \( V^\dagger \).

![Figure 3: Procedure for decomposing the ground state of a one-dimensional four-site quantum system. Singular value decompositions (SVD’s) are performed repeatedly in order to build a matrix product state in left canonical form.](image)

In this example, I started performing SVD’s from the left, and at each step I absorbed the singular value matrix \( s \) into \( V^\dagger \). As a result, the tensor network has only unitary tensors, except from the tensor on the far right. This form of the MPS is called the left canonical form, and then the MPS is said to be left orthonormalized. I could also have started from the right, and have absorbed \( s \) into \( U \) at each step. Then the MPS would be in right canonical form and would then be right orthonormalized. It is also possible to orthonormalize the MPS both from the left and the right, which would lead to the absorption of \( s \) into a \( U \) or \( V^\dagger \) of a site somewhere in the middle. This form is called the mixed canonical form with respect to that specific site. The canonical form of an MPS is useful in the calculation of norms and expectations values, where we can use the unitarity of the tensors for simplifications of the network. For example, if you want to calculate the expectation value for an operator that acts on site 3, then the calculation becomes simpler when the MPS is in mixed canonical form with respect to that site, as shown in Fig. 4.
2.2.2 Uniform matrix product state

So far, we have looked to finite $N$-site quantum chains, but we will now move on to MPS’s for infinite chains with translational invariance. These states are called uniform matrix product states (uMPS’s) \[17\]. Such states can be described by a single rank-3 tensor, which is repeated infinitely many times as shown in Fig. 5. I will denote this tensor as $T^{s_i}_{jk}$, or simply as $T^{s_i}$ or $T$, where $s_i$ is the physical index that can take $q$ different values corresponding to $q$ different spin states. The indices $j$ and $k$ are used to connect the tensors into a chain in auxiliary space, and can have again bond dimension $\chi$.

\[
\psi(T) = \cdots T T T T T T T \cdots
\]

Figure 5: The uniform matrix product state describes the ground state for an infinite quantum chain. Because of translational invariance this state can be described by a single tensor $T$.

When looking at this representation of the ground state, you can see that it remains invariant when doing the local gauge transformation: $T^{s_i} \rightarrow X \cdot T^{s_i} \cdot X^{-1}$, where $X$ is an invertible, $\chi \times \chi$ matrix. By using a suitable gauge transformation and a proper renormalization, we can bring the uMPS into the left or right canonical form. In these representations, we have the following isometric constraints respectively, which are also shown in diagrammatic notation in Fig. 6.

\[
\sum_{s_i=1}^{q} (T^{s_i}_L)^\dagger T^{s_i}_L = \mathbf{1}, \quad \sum_{s_i=1}^{q} (T^{s_i}_R)^\dagger T^{s_i}_R = \mathbf{1}
\] (2)
Figure 6: Isometric constraints for the uniform matrix product state in (left) the left canonical form and (right) the right canonical form.

These constraints are related through the isometries $C_1$ and $C_2$ by the following equations respectively:

\begin{align}
C_1 T^s_i \propto T^s_L C_1 \\
T^s C_2 \propto C_2 T^s_R
\end{align}

In Fig. 7a and 7b, these equations are shown in diagrammatic notation, and it allows us to represent the ground state in mixed canonical form as in Fig. 7c. In this form, there is a tensor $T^s_C \equiv C_1 T^s C_2$ at one site of the chain, and all tensors on the left/right of this site are left/right orthonormalized. There are different ways to find the left and right canonical tensors and isometry tensors that obey Eqs. 3 and 4. In section 4.2.1, I describe the method that I use for left orthonormalizing a uMPS.

Figure 7: Isometry relations for the uniform matrix product state in (a) left canonical form and (b) right canonical form. The isometry matrices $C_1$ and $C_2$ can be used to represent the ground state in the mixed canonical form by defining $T^s_C \equiv C_1 T^s C_2$ (c).
3 The tensor network representation of the partition function

Tensor networks are not only useful in describing quantum states; they can also efficiently represent partition functions of classical spin systems. In this thesis, I investigate infinite two-dimensional square-lattice classical spin systems, with sites that can be in \( q \) possible spin states. I restrict to models with rotational and translational invariance, and where each site has only interactions with its nearest neighbors and an external field \( h \).

The energy of such a model is dependent on the spin configuration of the system \( \{s_i\} \), and is measured by the Hamiltonian of the system. The Hamiltonian characterizes the model, and can be written as

\[
H(\{s_i\}) = \sum_{\langle i,j \rangle} g(s_i, s_j) + h \sum_i s_i,
\]

where the first sum goes over all pairs of nearest neighbors, and \( g \) is a function that is symmetric in the variables \( s_i \) and \( s_j \), which are the spin values of the neighboring sites \( i \) and \( j \) respectively. This sum accounts for the nearest-neighbor interactions. The second sum goes over all lattice sites, and accounts for the coupling of each site to the external field \( h \), which is dependent on the spin state on that site. We can write the second sum as

\[
h \sum_i s_i = \sum_{\langle i,j \rangle} h \cdot (s_i + s_j)/4.
\]

The sum now goes over all nearest neighbors. Because each site has 4 nearest neighbors, we have added a factor 1/4 to avoid overcounting of sites. We can write Eq. 5 more generic by defining \( f(s_i, s_j) \equiv g(s_i, s_j) + h \cdot (s_i + s_j)/4 \), which is symmetric in \( s_i \) and \( s_j \):

\[
H(\{s_i\}) = \sum_{\langle i,j \rangle} f(s_i, s_j).
\]

Since the system is a collection of infinitely many spin sites, we need statistical mechanics to get information about it. An important function that describes statistical properties of many-body systems, is the partition function \[\text{18}\]. This function is given by

\[
Z(\beta) = \sum_{\{s_i\}} e^{-\beta H(\{s_i\})},
\]

where the sum goes over all possible spin configurations, and \( e^{-\beta H(\{s_i\})} \) is the Boltzmann weight corresponding to spin configuration \( \{s_i\} \). \( \beta \equiv \frac{1}{k_B T} \) is the inverse temperature, with
the Boltzmann’s constant and $T$ the temperature. In this thesis, I will set $k_B = 1$ for simplicity, such that $\beta = 1/T$. Plugging Eq. 5 into Eq. 8 results in the following expression for the partition function:

$$Z(\beta) = \sum_{\{s_i\}} e^{-\beta \sum_{\langle i,j \rangle} f(s_i, s_j)} = \prod_{\{s_i\} \langle i,j \rangle} e^{-\beta f(s_i, s_j)}$$

In this expression, the Boltzmann weight is replaced by a product of local Boltzmann weights.

### 3.1 Tensor network representation

The local Boltzmann weight in Eq. 9 can take $q^2$ possible values, as $s_i$ and $s_j$ can each take $q$ different values $\sigma_n$ corresponding to the $q$ different spin states; $s_i, s_j \in \{\sigma_1, \sigma_2, ..., \sigma_q\}$. Therefore, it is possible to rearrange the values of the Boltzmann weight into the Boltzmann weight matrix $Q$, with elements $Q_{s_i, s_j}$:

$$Q \equiv \begin{pmatrix}
  e^{-\beta f(\sigma_1, \sigma_1)} & e^{-\beta f(\sigma_1, \sigma_2)} & \cdots & e^{-\beta f(\sigma_1, \sigma_q)} \\
  e^{-\beta f(\sigma_2, \sigma_1)} & e^{-\beta f(\sigma_2, \sigma_2)} & \cdots & e^{-\beta f(\sigma_2, \sigma_q)} \\
  \vdots & \vdots & \ddots & \vdots \\
  e^{-\beta f(\sigma_q, \sigma_1)} & e^{-\beta f(\sigma_q, \sigma_2)} & \cdots & e^{-\beta f(\sigma_q, \sigma_q)}
\end{pmatrix}$$  \hspace{1cm} (10)

The partition function can then simply be written as

$$Z = \sum_{\{s_i\} \langle i,j \rangle} \prod_{\langle i,j \rangle} Q_{s_i, s_j},$$

from which we can see that it can be represented by a tensor network, consisting of the matrices $Q_{s_i, s_j}$ and rank-4 tensors $\delta_{ijkl}$. The $\delta$ tensor is a Kronecker delta tensor with 4 indices, and has the following elements:

$$\delta_{ijkl} = 1 \quad \text{if} \quad i = j = k = l, \quad \text{and} \quad \delta_{ijkl} = 0 \quad \text{otherwise.} \hspace{1cm} (12)$$

On each lattice site, such a tensor is present, and it determines how a site is connected to other sites. In Eq. 11, the $\delta$-tensors are hidden in the product over the nearest neighbor pairs. Figure 8 shows how the tensor network looks like in diagrammatic notation. I also show in this figure how the partition function can be represented by a tensor network consisting of identical tensors only, by introducing the following tensor:

$$a_{ijkl} = \sum_{n=1}^{q} \left( \sqrt{Q} \right)_{in} \left( \sqrt{Q} \right)_{jn} \left( \sqrt{Q} \right)_{kn} \left( \sqrt{Q} \right)_{ln},$$  \hspace{1cm} (14)
Figure 8: The tensor network representation of the partition function. The network consists of two types of tensors: Boltzmann weight matrix $Q$ and rank-4 Kronecker delta tensor $\delta$. The network can be made more uniform by replacing $Q$ by $\sqrt{Q} \cdot \sqrt{Q}$ and contracting $\delta$ on four indices with $\sqrt{Q}$ to form tensor $a$. 
3.2 Relation to thermodynamic variables

Once the partition function is calculated, it is possible to derive thermodynamic variables from it. Because we consider infinite systems, we are not interested in variables describing quantities of the total system, which are often infinite. For example, the magnitude of the total energy of the system is infinite, but the energy per site is finite. So we are interested in quantities per site, and therefore we have to use the partition function per site $\kappa$:

$$\kappa \equiv \lim_{N \to \infty} Z_{1/N}.$$  \hspace{1cm} (15)

The partition function per site is related to the free energy per site and specific heat per site by the following equations respectively:

$$f = -T \log(\kappa)$$ \hspace{1cm} (16)

$$c_v = -T \frac{\partial^2 f}{\partial T^2} = T \frac{\partial^2}{\partial T^2}(T \log(\kappa))$$ \hspace{1cm} (17)

3.2.1 Local operators

We can calculate the expectation value for a local operator $O_r$, acting on site $r$, by

$$\langle O_r \rangle = \frac{1}{Z} \sum_{\{s_i\}} O_r e^{-\beta H(\{s_i\})}. \hspace{1cm} (18)$$

In this expression we see that the partition function acts as a normalization constant. We also see that the sum is almost the same as the partition function; the only difference is the presence of the operator $O_r$. This sum can be represented by the tensor network as shown before in Fig. 8 but with the $\delta$-tensor on site $r$ replaced by the operator $O_r$. For example, if we want to calculate the expectation value for the magnetization per site, we put the operator $s_r$ on site $r$ that measures its spin value; $m = \langle s_r \rangle$. In the uniform representation of the partition function, this boils down to replacing the $a$-tensor on site $r$ by the tensor

$$b_{ijkl} = \sum_{n=1}^{q} \sigma_n^{(r)} \left( \sqrt{Q} \right)_i^n \left( \sqrt{Q} \right)_j^n \left( \sqrt{Q} \right)_k^n \left( \sqrt{Q} \right)_l^n. \hspace{1cm} (19)$$

Then we have to contract the tensor network and divide by the partition function, as shown in Fig. 9.

Figure 9: Diagrammatic notation of the calculation of the magnetization per site, which is a division of two tensor networks. The denominator is the partition function, consisting only of tensor type $a$. The numerator is the same, but one tensor is replaced by tensor $b$, which measures the spin value on that site.
3.2.2 Correlation length

In classical spin models, the spins are correlated to each other, because of the interactions between the sites. For example, in ferromagnetic models, where neighboring sites like to be in the same spin state, one site can influence the spin state of sites that are far away. The function that measures the correlation between site $i$ and site $j$ in a spin system is called the correlation function, and is given by

$$C(i, j) = \langle s_i s_j \rangle - \langle s_i \rangle \langle s_j \rangle$$ (20)

$$= \langle s_i s_j \rangle - \langle s_i \rangle^2$$ (21)

$$= \langle s_i s_j \rangle - m^2.$$ (22)

Here I used the fact that $\langle s_i \rangle = \langle s_j \rangle = m$, which follows from the translational invariance in the lattice. In almost all situations, the correlation function decays exponentially to zero with distance $|i - j|$ between site $i$ and $j$, according to

$$C(i, j) \sim \exp(-|i - j|/\xi),$$ (23)

where $\xi$ is the correlation length. This $\xi$ is a length scale that determines how far two spins can be separated and still influence each other. Baxter showed that the correlation length of the system can be calculated from the two largest eigenvalues of the tensor network that represents one (infinite) row of the system [19]. This string of tensors is called the row-to-row transfer matrix, and is shown in Fig. 10. The relation between the correlation length and the largest and second largest eigenvalue of the row-to-row transfer matrix, denoted by $\lambda_1$ and $\lambda_2$ respectively, is given by

$$\xi = \frac{1}{\log(\lambda_1/\lambda_2)}$$ (24)

As the row-to-row transfer matrix is dependent on temperature, the correlation length is dependent on temperature as well. I use the correlation length in order to describe how the phases of the Ising and clock model change if the temperature is changed. I give the results in Chapter 5, 6 and 7.

Figure 10: The row-to-row transfer matrix, which represents the partition function of one (infinite) row of the system.
3.3 The ansatz for the partition function

I showed that the partition function can be represented by a tensor network, but this network contains an infinite number of tensors. In order to get numerical results, I use the following ansatz for the partition function \[7\]:

\[
Z(\beta) = \sum_{\text{states}} \prod_{\text{tensors} a} \langle a | \langle a | \prod_{\text{tensors} C} \langle C | \prod_{\text{tensors} T} \langle T | \rangle \rangle
\]

Figure 11: Approximation for the partition function. The environment around one tensor \(a\) is represented by tensors \(C\) and \(T\), which represent an infinite quadrant and infinite half-row/column respectively. These environment tensors have bond dimension \(\chi\), which controls the accuracy of the approximation.

In this ansatz, the partition function is represented by one particular site, described by tensor \(a\), surrounded by an environment, described by tensors \(C_{ij}\) and \(T_{ij}^s\). These environment tensors \(C\) and \(T\) account for one infinite quadrant and one infinite half-row/column of the system respectively. We can use the same tensor \(C\) for all corners and the same tensor \(T\) for all edges because of the rotational symmetry in the system. These tensors are symmetric, i.e. \(C_{ij} = C_{ji}\) and \(T_{ij}^s = T_{ji}^s\), because of the reflection symmetry in the system. The bond dimension of the transfer matrices is \(\chi\), and this number controls the accuracy of the approximation. This is very similar to how \(\chi\) controls the accuracy of an MPS, as described in section 2.2. The ansatz approaches the exact representation in the \(\chi \to \infty\) limit. In Chapter 4, I describe two possible methods to find \(C\) and \(T\).

3.3.1 Calculation of thermodynamic variables

When using this ansatz, the calculation of expectation values for local one-site operators becomes easy. Figure 12 shows how the calculation of the expectation value for the magnetization per site is represented. Baxter showed that the partition function per site \(\kappa\) can be calculated in a special way, as shown in Fig. 12. From \(\kappa\) we can calculate for example the free energy and specific heat per site, using Eq. 16 and 17. Figure 12 shows how the row-to-row transfer matrix is represented; it is just the contraction of two \(T\) tensors along their physical index. The correlation length \(\xi\) can be calculated from the two largest eigenvalues of this row-to-row matrix, with the use of Eq. 24.
\[ \langle s_r \rangle = \frac{1}{Z} \sum_{\{s_i\}} s_r e^{-\beta H(\{s_i\})} \approx \]

(a)

\[ \kappa = \lim_{N \to \infty} Z^{1/N} \approx \]

(b)

\[ (\text{C C T T})^2 \]

(c)

Figure 12: Diagrammatic notation of (a) the magnetization per site, (b) the partition function per site, and (c) the row-to-row transfer matrix of the partition function. Tensors \( C \) and \( T \) are the environment tensors from the ansatz in Fig. 11. Tensors \( a \) and \( b \) are defined by Eq. 14 and 19 respectively.
4 Two methods to calculate the partition function

We now have an ansatz (Fig. 11), which can be used to calculate the partition function numerically. We still need, however, a method in order to determine the environment tensors $C$ and $T$. In this chapter, I give two numerical methods that can be used to determine these tensors. The first method is the Corner Transfer Matrix Renormalization Group (CTMRG) method. The second method, which is designed more recently, is the Fixed Point Corner Method (FPCM). Both methods are iterative; a certain procedure is repeated and in each iteration the environment tensors are updated. The procedure stops iterating when the tensors are not changing anymore, which means that they have reached convergence. Below I describe the two different methods and their technical details.

4.1 The Corner Transfer Matrix Renormalization Group Method

The origin of CTMRG goes back to the works of Baxter [21], Wilson [22] and White [5]. In 1976, Baxter found out that the partition function of two-dimensional classical systems could be formulated in terms of transfer matrices. Based on Wilson’s renormalization group method, White formulated in 1992 the Density Matrix Renormalization Group (DMRG) method, which was used for describing one-dimensional quantum systems. In 1995, Nishino and Okunishi showed that, in the thermodynamic limit, the work of Baxter was equivalent to DMRG. They unified these concepts into a single method; the Corner Transfer Matrix Renormalization Group method [6]. This method determines the environment tensors of the partition function by iterating a growth and renormalization scheme. Below, I will describe what the steps in the growth scheme are, and why a renormalization scheme is needed.

4.1.1 Growth scheme

CTMRG determines the environment tensors as follows. We start with initial (e.g. random) tensors $C$ and $T$, which have dimensions $\chi \times \chi$ and $\chi \times q \times \chi$ respectively. Then we repeat the following procedure, in which we first grow the size of the system from $N \times N$ to $(N + 2) \times (N + 2)$ by adding two rows and columns to the system. The system is then divided into nine pieces; one center $a$-tensor, four corner pieces and four edge pieces. Because of the rotational symmetry in the lattice, the four contracted corner pieces are the same, and we call these $M$. The four contracted edge pieces are the same for the same reason, and we call these $L$. These steps are shown in Fig. 13.
Figure 13: The system growth scheme in one CTMRG iteration. The lattice size is grown from $N \times N$ to $(N + 2) \times (N + 2)$ by adding site tensors $a$ and edge tensors $T$ to the system. Then the system is divided into nine pieces; one $a$-tensor, and new corner and edge environment tensors, which are denoted by $M$ and $L$ respectively.

In principle, we can properly regroup the indices of $M$ and $L$ such that $M$ becomes a matrix and $L$ a rank-3 tensor, and then use these tensors as $C$ and $T$ for the growth scheme in the next iteration. But a difficulty appears then, because the bond dimension of $C$ and $T$ will grow with factor $q$ in each iteration when the $a$-tensors are absorbed. This means that the computation time will grow exponentially with iteration number, which is a major problem when we want to get results in a finite amount of time. In order to avoid this, we need a renormalization scheme.

4.1.2 Renormalization scheme

To keep the bond dimension of $C$ and $T$ constant, instead of growing with factor $q$ per iteration, we perform a singular value decomposition on $M$, as shown in Fig. 14. Because $M$ is symmetric, we have $M = U \cdot s \cdot V^\dagger = U \cdot s \cdot U^\dagger$. Also, as we only look at real systems, $U^\dagger = U^T$. In order to perform an SVD on $M$, we first have to regroup the horizontal and vertical indices, such that $M$ becomes a $q \chi \times q \chi$ matrix. The main idea behind the renormalization is that we can approximate the exact matrix $M$ by a $\chi \times \chi$ matrix $\tilde{M}$. We do this by performing an SVD on $M$ and keeping only the $\chi$ largest singular values;

$$M \approx \tilde{M} = \tilde{U} \cdot \tilde{s} \cdot \tilde{U}^T,$$

(25)

where $\tilde{s}$ is a $\chi \times \chi$ diagonal matrix with the $\chi$ largest singular values of $s$, and $\tilde{U}$ is a $q \chi \times \chi$ matrix consisting of the first $\chi$ columns of $U$. The matrix $\tilde{U}$ is reshaped to a tensor of size $\chi \times q \times \chi$. The approximation in Eq. 25 minimizes the Frobenius norm $\|M - \tilde{M}\|$, and is therefore the best lower rank approximation.
First, a singular value decomposition is performed: $M = U \cdot s \cdot V^\dagger = U \cdot s \cdot U^T$. Then, this decomposition is approximated by keeping only the $\chi$ largest singular values: $M \approx \tilde{U} \cdot \tilde{s} \cdot \tilde{U}^T$.

We can now find the updated environment tensors $C$ and $T$ by sandwiching the tensors $M$ and $L$ between $\tilde{U}$ and $\tilde{U}^T$, as shown in Fig. 15. What we are effectively doing is putting approximated identities $\tilde{U} \cdot \tilde{U}^T \approx \mathbb{1}$ on each link between the tensors $M$ and $L$, and then absorbing $\tilde{U}$ and $\tilde{U}^T$ to different sides of the link. The updated environment tensors have the same dimensions as the ones we started with, and are used for the next iteration.

4.1.3 Overview CTMRG

An overview of the CTMRG method is shown in Fig. 16. We start with the ansatz in Fig. 11 which represents the partition function of a $N \times N$ system. Then we grow the system size to $(N+2) \times (N+2)$ by introducing bulk and edge tensors, according to the growth scheme. The corner and edge networks are contracted to tensor $M$ and $L$, and resolutions of the identity $\tilde{U} \cdot \tilde{U}^T$ are placed between them, according to the renormalization scheme. The new corner and edge tensors $C$ and $T$ are created by absorbing $\tilde{U}$ and $\tilde{U}^T$ into $M$ and $L$. Eventually, $C$ and $T$ will converge according to some given accuracy. They then can be used to calculate the partition function and other thermodynamic quantities.
Figure 16: Overview of the CTMRG iteration. We start with the partition function of a $N \times N$ lattice, given by the ansatz in Fig. 11 with environment tensors $C$ and $T$, and bulk tensor $a$. Then the system size is grown to $(N + 2) \times (N + 2)$ by inserting new $a$- and $T$-tensors. Then the edges and corners are contracted to tensors $L$ and $M$ respectively. Then approximated identities $\tilde{U} \cdot \tilde{U}^T \approx 1$ are placed between $M$ and $L$, where $U$ is obtained from an SVD on $M$, and keeping only the $\chi$ largest singular values; $M = U \cdot s \cdot U^T \approx \tilde{U} \cdot \tilde{s} \cdot \tilde{U}^T$. Then $M$ and $L$ are contracted with $\tilde{U}$ and $\tilde{U}^T$ in order to become the corner and edge tensors $C$ and $T$ for the new iteration respectively. This iteration is repeated until $C$ and $T$ are converged.
4.2 The Fixed Point Corner Method

Over the last years, CTMRG has been the most used method to contract two-dimensional tensor networks. Fishman et al., however, recently formulated a new method that uses the same ansatz (Fig. 11), but better exploits the translational invariance of the system [17]. This method, called the Fixed Point Corner Method, uses eigenvalue solvers to determine the environment tensors, as opposed to the power method used in CTMRG. According to the power method, the eigenvector with the largest eigenvalue, also called the fixed point or leading eigenvector $v_{leading}$, of a matrix $A$ can be found by multiplying a random initial vector $v_0$ by $A$ infinitely many times, and normalizing it in each iteration:

$$v_{leading} = \lim_{k \to \infty} \frac{A^k v_0}{||A^k v_0||}$$ (26)

In Eq. 26 is assumed that the starting vector $v_0$ is not orthogonal to $v_{leading}$. In CTMRG we see this method, as the environment tensors are multiplied by new sites in each iteration. The power method becomes less efficient when the difference between the eigenvalues of the two leading eigenvectors becomes smaller. FPCM makes use of direct eigenvalue solvers, so it is expected that the environment tensors will converge faster than when the power method is used.

In order to use FPCM, we have to note that the environment of the ansatz in Fig. 11 can be treated as an MPS. Because the ansatz describes an infinity system, we can add infinitely many rows and columns. So the top and bottom row of the ansatz can actually be written as an uMPS when we insert infinitely many columns. Using Eq. 4, we can left orthonormalize this uMPS as shown in Fig. 17.

---

Figure 17: uMPS representation of the ansatz in Fig. 11. First, an infinite number of columns are added to the (infinite) system. Then, the uMPS is left orthonormalized. The tensor $T$ on the far left side is multiplied infinitely many times by transfer matrix $K$. 

24
In Fig. 17 we see that the tensor \( T \) on the left end of the tensor network is multiplied infinitely many times by the same tensor, call it \( K \). Just like in the power method, this tensor \( K \) projects \( T \) onto the leading eigenvector of \( K \). Instead of multiplying \( K \) infinitely many times to \( T \), we can obtain \( T \) directly by solving the eigenvalue equation in Fig. 18.

\[
\text{Figure 18: The environment tensor } T \text{ can be determined by calculating the leading eigenvector of the transfer matrix comprised of left orthonormal tensors } T_L \text{ and tensor } a, \text{ which is defined by Eq. 14.}
\]

FPCM uses the above observations in order to determine the environment tensors as follows. We start with an initial (e.g. random) tensor \( T \) of size \( \chi \times q \times \chi \). Then we iterate the following steps:

1. Left orthonormalize the uMPS consisting of tensor \( T \) in order to find isometry tensor \( C \) and left orthonormal tensor \( T_L \), that satisfy: \( CT \propto T_L C \) and \( \sum_{s_i=1}^{q} (T_{s_i}^L)^T T_{s_i}^L = 1 \).
2. Using \( C \) and \( T_L \) from step 1, find \( T \) by solving the eigenvalue equation in Fig. 18.

The tensors \( C \) and \( T \) are updated in each iteration, and will eventually converge according to some given accuracy. We only have to determine the fixed points of one side of the system, because rotational and reflection invariance in the system makes all four sides equal to each other. We have chosen the left side, so therefore we need a left orthonormalization scheme. There are different ways how to left orthonormalize a uMPS, and below I describe the scheme I am using.

4.2.1 Left orthonormalization scheme

The goal of left orthonormalizing our uMPS, comprised of real symmetric tensors \( T \), is to find tensors \( C \) and \( T_L \), that satisfy the relations in Fig. 19.

\[
\text{Figure 19: Relations that has to be satisfied for a left orthonormalized uMPS consisting of tensor } T. \text{ (a) When the isometry matrix } C \text{ is pulled through tensor } T, \text{ it changes } T \text{ to } T_L. \text{ (b) This tensor } T_L \text{ is equal to identity when it is contracted with the left two indices of its transpose.}
\]
In order to do this we use the procedure described in [7]. We start with calculating the leading left eigenvector $C^2$ of the MPS transfer matrix, see Fig. 20.

![Figure 20: Eigenvalue equation that determines the initial environment tensor $C$. This tensor is the square root of the left fixed point $C^2$ of the MPS transfer matrix comprised of tensor $T$.](image)

The MPS transfer matrix is real and symmetric, and therefore we can write $C^2$, after performing an SVD, as:

$$C^2 = U s U^T = U \sqrt{s} \sqrt{s} U^T = U \sqrt{s} U^T U \sqrt{s} U^T. \quad (27)$$

So then we obtain $C = U \sqrt{s} U^T$. We can find $A_L$ and the new $C'$ by the uniqueness of the left polar decomposition. This is done with the following steps:

1. Multiply $C$ and $T$: $T_{C_T}^{s_i} \equiv C T^{s_i}$
2. Perform an SVD: $T_{C_T}^{s_i} = U^{s_i} s V^T = U^{s_i} V^T V s V^T$
3. Define $T_L$ and new $C'$: $T_{L_T}^{s_i} = U^{s_i} V^T$ and $C' = V s V^T$

Our initial error in this calculation is $||C - C'||$, which is limited to the square root of machine precision, i.e. $O(10^{-8})$, because we calculated the square root of the singular values of the leading eigenvector $C^2$. In order to get higher accuracy, we repeat the following steps until convergence:

4. Replace $C'$ by the left leading eigenvector $C$ of the mixed transfer matrix of $T^{s_i}$ and $(T_{L_T}^{s_i})^T$, by solving the eigenvalue equation in Fig. 21
5. Perform a left polar decomposition of $C = W C'$ such that $C'$ is positive and symmetric, and change the gauge of $T_L$ by $W$, i.e. $T_{L_T}^{s_i} \rightarrow W^T \cdot T_{L_T}^{s_i} \cdot W$.
6. Perform a left polar decomposition on $C' T^{s_i}$ as before to determine the new $T_L$ and $C'$, i.e. do step 1 through 3 with $C = C'$.

![Figure 21: Eigenvalue equation to calculate the improved environment tensor $C$. This equation is derived from Eq. 3 by multiplying both sides by $(T_{L_T}^{s_i})^T$, and using the orthonormality of $T_L$.](image)
I described the CTMRG and FPCM methods above, but in order to implement them into a working program, we have to add some small steps. First of all, because of round-off errors in the tensor multiplications, the tensors $C$ and $T$ will lose their symmetry. Therefore we have to symmetrize them at the end of each iteration:

\[
C_{ij} \rightarrow \frac{(C_{ij} + C_{ji})}{2} \quad (28)
\]
\[
T_{si}^{\alpha} \rightarrow \frac{(T_{si}^{\alpha} + T_{si}^{\alpha})}{2} \quad (29)
\]

Also, the values of the elements of $C$ and $T$ will grow when the program iterates. As computers cannot handle arbitrary large numbers, we have to keep these values bounded. This is done by dividing $C$ and $T$ by their largest magnitude element at the end of each iteration:

\[
C \rightarrow C / \max(|C|) \quad (30)
\]
\[
T \rightarrow T / \max(|T|) \quad (31)
\]

This is possible, because we can multiply $C$ and $T$ each by an arbitrary constant. In the end, these constant do not matter when we calculate thermodynamic variables, which can be seen from the equations in Fig. 12a and Fig. 12b, in which the expectation value of a local operator and the partition function per site are calculated. In these equations, we see the same number of $C$ and $T$ tensors in the numerator as in the denominator, so the arbitrary constants will drop out. The correlation length is unaffected as well, because the arbitrary constants will drop out when we divide the two largest eigenvalues in Eq. 24.

### 4.3.1 Starting environment tensors

As described before, we can start the simulations with random $C$ and $T$, with size $\chi \times \chi$ and $\chi \times q \times \chi$ respectively. These starting tensors can be regarded as boundary conditions of the system, which are not important if we let $C$ and $T$ converge, i.e. when we let them reach their representation in the thermodynamic limit. In this limit, the boundary is infinitely far away from the bulk, and therefore we expect them to have no influence on the physics. There are, however, reasons why we should not take random boundary conditions. The speed of the simulations depends on the initial conditions of the tensors; the tensors converge faster when you provide a better initial guess.
Instead of choosing random starting tensors, we could choose the following initial guesses for $C$ and $T$:

$$C_{ij} = \sum_{n=1}^{q} \left( \sqrt{Q} \right)_{in} \left( \sqrt{Q} \right)_{jn} = Q_{ij} \quad \text{and} \quad T_{ijk} = \sum_{n=1}^{q} \left( \sqrt{Q} \right)_{in} \left( \sqrt{Q} \right)_{jn} \left( \sqrt{Q} \right)_{kn}. \quad (32)$$

The matrix $Q$ is again the Boltzmann weight matrix, defined in [10]. With these guesses, the environment tensors are just like the rank-4 tensor $a$, which is defined in Eq. [14] and represents the partition function of one site in the bulk of the system. So the initial $C$ and $T$ represent now the partition function of a site on the boundary corner and boundary edge of the lattice respectively. Let us call these guesses the *free boundary conditions*. This choice is natural, as we then start our simulation with the partition function of a $3 \times 3$ lattice. We can see from these guesses that we start with a bond dimension $\chi = q$. If we want to start with a higher bond dimension, we use $n$ CTMRG iterations without the renormalization scheme in order to increase the bond dimension to $\chi' = q^{n+1}$. If the bond dimension $\chi'$ exceeds the desired one $\chi$ (which is the case when there exist no integer $n$ such that $\chi = q^{n+1}$), we use the renormalization scheme of CTMRG to reduce the bond dimension to the desired one $\chi' \to \chi$.

We can also start the simulations with a $3 \times 3$ lattice in which the boundary sites are prepared in a particular spin state $\sigma_n$. For example, when the boundary sites are prepared in the first spin state $\sigma_1$ of the set of spin states, we have the following initial guesses:

$$C_{ij} = \sum_{n=1}^{q} \delta_{1n} \left( \sqrt{Q} \right)_{in} \left( \sqrt{Q} \right)_{jn} \quad \text{and} \quad T_{ijk} = \sum_{n=1}^{q} \delta_{1n} \left( \sqrt{Q} \right)_{in} \left( \sqrt{Q} \right)_{jn} \left( \sqrt{Q} \right)_{kn}. \quad (34)$$

The $\delta$-tensor in these equations is defined as follows:

$$\delta_{1n} = 1 \quad \text{if} \ n = 1, \ \text{and} \quad (36)$$

$$\delta_{1n} = 0 \quad \text{otherwise.} \quad (37)$$

Again, the starting bond dimension is in this case $\chi = q$. If we want to start with a higher bond dimension, we can use the same procedure that I described for the free boundary conditions. These guesses turns out to speed up the simulations, and become useful in calculations of the magnetization. Let us call these guesses the *fixed boundary conditions*, because the boundary sites are fixed to a particular spin state.

For the FPCM simulations, a good initial guess would be environment tensors with fixed boundary conditions, which are followed by $\chi$ CTMRG iterations. With the fixed point equations in FPCM, the tensors go directly to the thermodynamic limit, so it makes
more sense to start with tensors that already represent a larger system. The choice of $\chi$
CTMRG iterations appears to be optimal for the computation time in the $\chi$-range that
I will use in my simulations. In order to make a fair comparison between CTMRG and
FPCM, I use the same initial tensors for CTMRG.

### 4.3.2 Convergence criterium

When the simulations are run, the environment tensors are improved in each iteration
and converge to their representation in the thermodynamic limit. But if we want to get
results within a given accuracy, the program has to stop iterating at some point. I use
the change of the singular value spectrum of $C$ as a measure of convergence. So at the
end of each iteration $n$, I determine the singular values of $C$ by performing an SVD, and
compute the following quantity:

$$\epsilon(n) = \sum_{k=1}^{\chi} |s_k^{(n)} - s_k^{(n-1)}|,$$

where $s_k^{(n)}$ is the $k^{th}$ singular value of environment tensor $C$ in the $n^{th}$ iteration. These
singular values are normalized such that the largest singular value is one. So $\epsilon$ is basically
the difference between the singular value spectrums of $C$ from the current iteration and
previous iteration. When $\epsilon$ drops below a given tolerance (e.g. $\epsilon < 10^{-6}$), the simulation
stops iterating and $C$ and $T$ can be used to calculate thermodynamic variables within a
certain accuracy. When the tolerance is chosen smaller, the accuracy of the results will
be higher. This is, however, at the expense of a longer computation time.

The thermodynamic limit representation of the environment tensors is reached when
we take the limits where the bond dimension and the number of iterations go to infinity.
This representation will never be reached as we have to use a finite bond dimension and
a finite number of iterations in order to get results in a finite amount of time. We are
also limited with the machine precision, so arbitrary high accuracies are not obtainable.
Therefore, we expect to see some finite-size effects in our results.

### 4.3.3 Determination of fixed points with FPCM

In one FPCM iteration step, several eigenvalue equations have to be solved to find $C$ and
$T$ in terms of fixed points. In my Matlab simulations, I use the build-in `eigs` function to
do this. This function uses the (iterative) Lanczos method. When a matrix $A$ is passed
to this function, it is applied multiple times to an initial vector in order to determine the
eigenvectors with the largest eigenvalues. The eigenvalue equations in Fig. 18, 20 and 21
can be solved by reshaping eigenvectors $C$ and $T$ into vectors, and reshaping the applied
tensor networks into matrices. For example, the eigenvalue equation in Fig. 21 can be
calculated as shown in Fig. 22.
This is computationally not preferable, because the applied matrix will then have size $\chi^2$ by $\chi^2$, which is a problem when we want to do simulations with large $\chi$. Fortunately, the `eigs` function has the possibility to pass as an input a function instead of a matrix. This function determines which tensors are applied to the initial tensor. So in the example above, `eigs` will apply two tensors of size $\chi \times q \times \chi$, instead of one matrix of size $\chi^2 \times \chi^2$, which avoids computation and storage problems in the large-$\chi$ regime.

The order in which the tensors are applied to the initial tensor has also an important effect on the computation time. When two tensors are contracted, the computational cost is determined by the product of the dimensions of all their indices, where the indices that connect the two tensors are counted only once. Figure 23 shows the computational cost for two possible contraction sequences in the example above.

The contraction sequence that minimizes the computational cost is chosen in each eigenvalue equation of the FPCM iteration. The leading computational cost of FPCM is then of order $O(q^2 \chi^3)$. This scaling is better than for a CTMRG iteration, where the leading computational cost is of order $O(q^3 \chi^3)$. 

Figure 22: (a) Eigenvalue equation of Fig. 21. (b) Tensor $(T_L)^T$ and $T$ are multiplied, and their product is called $A$. (c) Tensor $C$ is reshaped into a vector and tensor $A$ into a matrix.

Figure 23: Computational cost of the two possible contractions (a) and (b) of the tensor network of Fig. 21. The red numbers on the tensor legs indicate which contraction sequence is used. The red expressions on the side of the diagrams indicate the corresponding computational cost. Contraction sequence (a) is favored in the large-$\chi$ regime because of the lower computational cost.
The `eigs` function uses random initial tensors as default for guessing the fixed points. To speed up the determination of the fixed points, I use the $C$ or $T$ from the previous iteration as initial tensors for the eigenvalue equations in Fig. 21 and 18 respectively. For the eigenvalue equation in Fig. 20, I use the following initial tensor: 

$$C^2 = \sum_{s_i=1}^{q} (T_{s_i})^T T_{s_i},$$

with the $T$ from the previous iteration. In the options of `eigs`, I also change the number of Lanczos basis vectors to 12, and the tolerance of convergence to $10^{-3}$. These changes have negligible effect on the accuracy of the results, and speed up the algorithm significantly.
5 Results for the Ising model

To test and compare CTMRG and FPCM, I first investigate the Ising model. The Hamiltonian of this model was formulated in 1920 by Wilhelm Lenz. In 1925, his student Ernst Ising solved this model, defined on a one-dimensional spin chain [8]. The two-dimensional version is more interesting, as this system undergoes a phase transition when the temperature is decreased below a certain critical point. In the absence of an external field, there are exact solutions known, which makes this model a good candidate for testing our methods. In this chapter, I first give the definition of the Ising model, its properties and the exact solutions we can use. Then I give the results of simulations in which I calculate thermodynamic variables using the two different methods.

5.1 Definition of the Ising model

I investigate the infinite two-dimensional square-lattice Ising model. In this lattice, each lattice site \( i \) can be in two possible spin states \( (q = 2) \), with spin values \( s_i \in \{+1, -1\} \). We can also say that on each site sits a spin, which is represented by an arrow that can either point up \(|\uparrow\rangle\) or down \(|\downarrow\rangle\). The Ising model is defined by the following Hamiltonian:

\[
H(\{s_i\}) = -\sum_{\langle i,j \rangle} J_{ij} s_i s_j - \sum_i h_i s_i. \tag{39}
\]

The first sum in Eq. (39) goes over all nearest-neighboring sites, and accounts for the interaction between each pair of neighboring sites \( i \) and \( j \), which has coupling strength \( J_{ij} \). The second sum goes over all sites, and accounts for the interaction of each site \( i \) with an external field, which has coupling strength \( h_i \). I will assume that the interactions between all neighboring sites have the same strength, and that each site has the same coupling strength to the external field. The Hamiltonian can then be written as

\[
H(\{s_i\}) = -J \sum_{\langle i,j \rangle} s_i s_j - h \sum_i s_i. \tag{40}
\]

The sign of \( J \) and \( h \) have physical significance. From the Hamiltonian we see that, when \( J < 0 \), the energy is lowered when \( s_i \) and \( s_j \) have opposite spin values. Since lower energies are favored in physical systems, we conclude that neighboring sites prefer to be in different spin states. The spin configuration with the lowest energy is shown in Fig. 24a, in which we can see anti-ferromagnetic behavior. That is why the \( J < 0 \) case is called the anti-ferromagnetic Ising model. Following the same reasoning, neighboring sites prefer to have the same spin values when \( J > 0 \). The lowest-energy configuration is shown in Fig. 24b, from which we can see why this case is called the ferromagnetic Ising model.
The sign of \( h \) determines which spin state is favored. If \( h < 0 \), a site gives a lower energy contribution to the total energy when it is in the spin down state, so these states are favored. In the same manner, if \( h < 0 \), the spin up states are favored. Therefore, we can regard \( h \) as the strength of an external magnetic field, and the sign of \( h \) as its direction. The relative magnitude of \( J \) and \( h \) determines how the nearest-neighbor interactions and external field interactions are competing. I investigate the ferromagnetic Ising model in the absence of an external field (\( h = 0 \)). I will work in units where I can set \( J = 1 \).

### 5.1.1 Tensor network representation

In order to write the partition function of the Ising model as a tensor network, we have to use the Boltzmann weight matrix from Eq. 10. For the Ising model, the function \( f \) in this matrix has the following form:

\[
f(s_i, s_j) = -J s_i s_j - h \cdot (s_i + s_j)/4.
\] (41)

The Boltzmann weight matrix has then the following representation:

\[
Q = \begin{pmatrix} e^{{\beta}(J+h/2)} & e^{-\beta J} \\
 e^{-\beta J} & e^{{\beta}(J-h/2)} \end{pmatrix}.
\] (42)

In the simulations, I use \( Q \) with \( J = 1 \) and \( h = 0 \). With this \( Q \) we can represent the partition function as explained in section 3.1.
5.1.2 Phase transition

A very interesting feature of the ferromagnetic Ising model is that it can undergo a phase transition when the lattice dimension is bigger than one. At each temperature, the system is in the spin configuration that minimizes the total free energy;

\[ F = E - TS, \tag{43} \]

The first term of the free energy is the total energy \( E \) of the system, and the second term, which is proportional to the temperature, is the total entropy \( S \) of the system. At zero temperature, the system is in the configuration with the lowest total energy. This is the configuration in which all spins are aligned in the same direction; all pointing up or all pointing down, like in Fig. 24b. The system is then ordered, because the majority of sites are in the same spin state. When the temperature is non-zero, there is a competition between the total energy and the entropy of the system. At high temperatures, the system is in the configuration with the highest entropy, which are the configurations with the same number of up spins as down spins. The system is then disordered. There is one point at which the free energy of the ordered and disordered phase are equal. This point separates the ordered phase from the disorder phase, and is called the critical temperature \( T_c \).

If you look at the Hamiltonian of the Ising model in Eq. 40 in the case where \( h = 0 \), you can see that it has a discrete \( \mathbb{Z}_2 \) symmetry. The total energy of the system does not change when all the spins are flipped: \( \{s_i\} \rightarrow -\{s_i\} \). When the temperature drops below \( T_c \), this symmetry is spontaneously broken, because the ferromagnetic configuration in Fig. 24b is not invariant under a total spin flip. Also when the magnetic field strength \( h \) becomes nonzero, the symmetry is broken.

So there are two possible phases in the Ising model: the ordered and disordered phase. A quantity that can indicated in which phase the system is in, is called an order parameter. This quantity ranges continuously from 0 to 1, and the assigned value describes the order in the system. This value is 0 if the system is fully disordered, so when the number of up spins is equal to the number of down spins. It is 1 if the system is fully disordered, so when all sites are in the same spin state. For the Ising model, we can use the absolute value of the magnetization per site \( |m| \) as an order parameter. The magnetization goes continuously from 1 to 0 when \( T_c \) is approached from below, which is typical for a second-order phase transition.

All thermodynamic quantities show critical behavior when the temperature approaches \( T_c \). For example, the correlation length, specific heat \( c_v \), slope of the magnetization and magnetic susceptibility \( \chi_m \) diverge when \( T_c \) is approached. The divergence of these
quantities go with a power-law, each with its own critical exponent \((\nu, \alpha, \beta, \gamma)\):

\[
\begin{align*}
\xi & \sim |t|^{-\nu} \quad (44) \\
c_v & \sim |t|^{-\alpha} \quad (45) \\
m & \sim t^\beta \quad (46) \\
\chi_m & \sim |t|^{-\gamma}, \text{ where } t = (T - T_c)/T_c. \quad (47)
\end{align*}
\]

This critical exponent \(\beta\) should not be confused with the inverse temperature. These critical exponents are related to each other by the following scaling law:

\[
\alpha + 2\beta + \gamma = 2. \quad (48)
\]

The values of these critical exponents are dependent on the dimensions and symmetry of the systems. They are universal, i.e. they do not depend on the microscopic details of the model. For the two-dimensional square-lattice Ising model, the specific heat diverges logarithmically \((\alpha = 0)\), and the other exponents have the following values [24]:

\[
\beta = \frac{1}{8}, \quad \gamma = \frac{7}{4} \quad \text{and} \quad \nu = 1. \quad (49)
\]

### 5.1.3 Exact solutions

In 1925, Ernst Ising solved the one-dimensional Ising model, but finding exact solutions for the two-dimensional Ising model was a lot more difficult. It was almost 20 years later, when Lars Onsager found exact solutions for the two-dimensional Ising model in the absence of a magnetic field \((h = 0)\), which was groundbreaking work in theoretical physics [9, 25]. He found solutions for the free energy and magnetization per site. Although he did not derive the solution for the magnetization, it was three years later derived by Yang [26].

The exact solutions, with \(J = 1\), are given by:

\[
\begin{align*}
m(T) &= [1 - \sinh^{-4}(2/T)]^{\frac{1}{2}} \quad \text{and} \\
f(T) &= -T \left(\frac{\log(2)}{2} + \frac{1}{2\pi} \int_0^\pi \log \left[\cosh^2(2/T) + \frac{1}{k} \sqrt{1 + k^2 - 2k \cos(2\theta)}\right] \, d\theta\right), \quad (51)
\end{align*}
\]

where \(k = \sinh^{-2}(2/T)\). By equating the magnetization per site in Eq. [50] to zero, we find the following value for the critical temperature:

\[
T_c = \frac{2}{\sinh^{-1}(1)} = 2.26918531. \quad (52)
\]

### 5.2 Analytic results

To compare CTMRG and FPCM, I calculate the magnetization and free energy per site, using fixed boundary conditions. Because we have the exact solutions of these quantities at hand, we can conclude which method performs better. I also calculate the correlation length and the critical temperature, using both methods.
5.2.1 Singular value spectrum

Both methods use the same ansatz in Fig. 11 and their accuracy can be controlled by changing the bond dimension $\chi$. Figure 25 shows the singular value spectrum of the environment tensor $C$ with $\chi = 60$ for three different temperatures. The spectrum is normalized such that the highest singular value is one. The first temperature is in the ordered phase ($T < T_c$), the second temperature is at the critical point ($T = T_c$), and the third temperature is in the disordered phase ($T > T_c$).

![Figure 25: The singular value spectrum of the C tensors with bond dimension $\chi = 60$ for three temperatures ($T < T_c$, $T = T_c$, and $T > T_c$). These results are obtained with CTMRG, using 5000 iterations.]

From Fig. 25 we see that, at the temperatures in the ordered and disordered phase, the singular value spectrum decays rapidly. This means that a small bond dimension $\chi$ can be used to approximate $C$, as most singular values are very small and can be neglected. Because of degenerate singular values, the spectrum decays step-wise at these temperatures. At the critical point, the decay is rather slow, so a large $\chi$ must be chosen to approximate $C$ with high accuracy. We see that the spectrum becomes smoother as the critical point is approached, but we can still identify a step-wise behavior. This has effect on finite-$\chi$ scaling as we will see later.
5.2.2 Magnetization

The magnetization of the Ising model can be calculated by the equation in Fig. 12, with the following representation of $b_{ijkl}$:

$$b_{ijkl} = \sum_{n=1}^{2} \sigma_{n}^{(r)} \left( \sqrt{Q} \right)_{in} \left( \sqrt{Q} \right)_{jn} \left( \sqrt{Q} \right)_{kn} \left( \sqrt{Q} \right)_{ln},$$

(53)

where $\sigma_{1}^{(r)} = +1$ and $\sigma_{2}^{(r)} = -1$. I have calculated the magnetization per site near the critical temperature for different bond dimensions $\chi$. A convergence tolerance of $10^{-6}$ is used for both methods. The results are shown in Fig. 26. From this figure, we see that the exact solution is approached when $\chi$ is increased. This is expected, as we said before that our approximation becomes better when we use a higher bond dimension, and the thermodynamic limit is reached in the $\chi \to \infty$ limit.

![Figure 26: Magnetization per site as a function of temperature. The graphs show $m$ for a convergence tolerance of $10^{-6}$, and for different bond dimensions $\chi$. The results on the left are obtained by CTMRG, and on the right by FPCM. The exact solution is also indicated in both figures.](image)

We also see that the curves do not abruptly go to zero like the exact solution, i.e. they do not have a sharp cut-off, but fall of with a tail. This tail is present because of finite-size effects, which can be reduced by lowering the convergence tolerance. These finite-size effects are studied below. Although the FPCM simulation took more computation time than the CTMRG simulation, the results of FPCM are more accurate as they suffer less from the finite-size effects. In order to compare how the accuracy of the results depend on the convergence tolerance, I have plotted the magnetization per site for several tolerances in Fig. 27.
In Fig. 27 we see that, for a certain $\chi$, the results converge to a curve with a sharp cut-off at a pseudo critical temperature $T^{\star}_c(\chi)$, when the tolerance is lowered. We see that the tail of the magnetization obtained with FPCM is less smooth than the tail of the magnetization obtained with CTMRG. But we also see that, when the tolerance is lowered, the tail vanishes more rapidly for FPCM than for CTMRG. This shaky behavior for FPCM is a direct consequence of the convergence of $\epsilon$, defined by Eq. 38. This parameter converges smoothly for CTMRG, as the singular value spectrum of the corner tensor $C$ changes in the same fashion in each iteration when new sites are absorbed into $C$. For FPCM, fixed point equations are used to determine $C$, and it can happen that the spectrum does not really change much in two consecutive iterations. Therefore, it is possible that $\epsilon$ abruptly drops below the given tolerance and the simulation terminates. This effect happens near the critical point, where the gap between the largest eigenvalues of the environment tensors becomes small. For the eigs function it is then harder to compute the fixed points.

From Fig. 26 and 27 we see the effect of two length scales in the simulations. The first length scale is induced by the bond dimension $\chi$, which determines how many basis states we use to represent the infinite system. We call this length scale $\xi_\chi$, for reasons that become clear when we study the correlation length below. The second one is the physical size $N$ of the system, which is determined by the tolerance. When the tolerance is chosen very small, the finite-size effects will be almost negligible, and the only relevant length scale is $\xi_\chi$. 
From Fig. 27 we can conclude that, for FPCM, a larger tolerance can be used to approach the exact curve within a certain accuracy than for CTMRG. So in simulations where we use a the tolerance of $10^{-10}$ for CTMRG, we can use a tolerance of $10^{-6}$ for FPCM to get results with roughly the same accuracy. In order to compare the speed of both methods, we should therefore not compare the computation time of the simulations where the same tolerance is used for both methods. Instead, we should look at what the computation time would be to get results within a certain accuracy. Figure 28 shows how the magnetization converges in time for both methods for a variety of bond dimensions. This simulation is done near the critical point.

![Figure 28: Convergence of the magnetization per site for different bond dimensions $\chi$ at temperature $T = (1 + 0.001)T_c$, where the exact solution is $m_{\text{exact}} = 0$. The error in the magnetization is defined by Eq. 54. The asterisks show when the magnetization is converged.](image)

From Fig. 28, we see that, in order to get the magnetization per site within a certain accuracy, FPCM needs less computation time than CTMRG. Roughly speaking, FPCM is twice as fast. It takes more time to do one FPCM iteration than one CTMRG iteration, because of the fixed points that have to be determined, but significantly less iterations are needed to get results within a certain accuracy. From the plot we can see that, for $\chi = 50$, our initial tensors are not good guesses for CTMRG, which leads to convergence troubles. Ignoring the results for $\chi = 50$, we see that both methods eventually approach the exact solutions within the same error. This is not only the case for temperatures near the critical point. Figure 29 shows the convergence of the magnetization per site for $\chi = 20$ for three temperatures; $T < T_c$, $T \approx T_c$ and $T > T_c$. The difference between the speed of both methods becomes more significant near the critical point.
Figure 29: Convergence of the magnetization per site for bond dimensions $\chi = 20$, and for three different temperatures. The first temperature is in the ordered phase, the second near the critical point, and the third in the disordered phase. The error in the magnetization is defined by Eq. 54.

Figure 30 shows the error in the magnetization and free energy per site as a function of temperature for $\chi = 20$. These errors are defined by:

$$\delta m = |m - m_{\text{exact}}|,$$
$$\delta f = |f - f_{\text{exact}}|,$$

where $m_{\text{exact}}$ and $f_{\text{exact}}$ are given by Eq. 50 and 51 respectively. If we want to compare both methods, it is not meaningful to use the same convergence tolerance for both methods. I have chosen a tolerance of $10^{-10}$ for CTMRG, and $10^{-6}$ for FPCM, because the results have then the same accuracies in the ordered phase. The plots show, however, that the results of FPCM have a higher accuracy than the results of CTMRG in the disordered phase. Despite the big difference in the tolerance, FPCM suffers less from finite-size effects, which is expected. FPCM uses fixed point equations to determine the environment tensors, which is more direct than the power-method used in CTMRG. The finite-size effects of CTMRG can, however, be reduced by lowering the tolerance even further, like we can see in Fig. 29.

In Fig. 30 I have also plotted the computation time of the simulation as a function of temperature. For temperatures far from the critical point, both methods are fast, but CTMRG performs better than FPCM. In the small interval around the critical temperature, FPCM becomes roughly two times faster than CTMRG.
Figure 30: Accuracies of results obtained with CTMRG and FPCM. From top to bottom: error in magnetization per site (defined in Eq. 54) as a function of temperature, error in free energy per site (defined in Eq. 55) as a function of temperature, and the computation time of the simulations as a function of temperature. The results are shown for $\chi = 20$, and a convergence tolerance of $10^{-10}$ for CTMRG and $10^{-6}$ for FPCM.
5.2.3 Correlation length

The correlation length is plotted as a function of temperature for different bond dimension in Fig. 31. We see that each curve has a narrow peak at a certain temperature. The position and height of this peak are dependent on the bond dimension $\chi$. When $\chi$ is increased, the position of the peak approaches the exact critical temperature, and the height of the peak becomes larger. In the thermodynamic limit, the correlation length goes to infinity at the critical temperature, but in the simulations it stays finite, because of the finite $\chi$ that we are using.

The correlation length indicates how far away two spins can be separated and still influence each other. For finite systems of linear size $N$, $\xi$ can never exceed $N$. The finite-size effects in Fig. 31 are almost negligible because of the low tolerances that we have chosen. We, however, see that the maximum of the correlation length $\xi_\chi$ is still finite, because it is limited by $\chi$. Thus, besides $N$, $\xi_\chi$ is another length scale in the system.

![Figure 31: Correlation length as a function of temperature for different bond dimensions $\chi$. The data on the left is obtained with CTMRG, using a convergence tolerance of $10^{-8}$. The data on the right is obtained with FPCM, using a convergence tolerance of $10^{-6}$. The dashed lines indicate the position of the exact critical temperature.](image)
5.2.4 Determination of critical temperature by finite-\(\chi\) scaling

We can use finite-\(\chi\) scaling in order to calculate the critical temperature with both methods. We do this in a similar way as the finite-\(N\) scaling, which is described in Ref. [31]. Instead of \(N\), we use the other length scale \(\xi\).

In the thermodynamic limit, as described in section 5.1.2, the correlation length diverges near the critical point with a power law, according to

\[
\xi(T) \sim |T - T_c|^{-\nu}.
\]  

(56)

In our simulations, where \(\chi\) is finite, the correlation length \(\xi\) stays finite at the temperature \(T^*_c(\chi)\) at which it has its maximum. We introduce \(T^*_c(\chi)\) as an effective temperature and, given Eq. (56), we expect it to satisfy the following relation:

\[
\xi(\chi(T^*_c(\chi))) \sim |T^*_c(\chi) - T_c|^{-\nu}.
\]  

(57)

We can rewrite this equation as:

\[
T^*_c(\chi) = T_c + c_0 \cdot \xi(\chi(T^*_c(\chi)))^{-1/\nu},
\]  

(58)

where \(c_0\) is a constant. In order to determine the exact critical temperature \(T_c\), I calculate the correlation length near the critical point for bond dimension \(\chi = 20\) to \(\chi = 100\) with steps of 4. For each \(\chi\), I determine the position \(T^*_c(\chi)\) and height \(\xi(\chi(T^*_c(\chi)))\) of the peak of the correlation length. The temperature steps have size \(10^{-4}\), but in order determine \(T^*_c(\chi)\) more accurately, I interpolate the correlation length on a finer grid with temperature steps of size \(10^{-5}\). On the obtained data, I use a fit of the form of Eq. (58), which gives me the critical temperature.

For CTMRG, I used a convergence tolerance of \(10^{-8}\), and for FPCM a tolerance of \(10^{-6}\). The obtained data for the positions and heights of the maximum of the correlation length are shown in Fig. 32. The fits give the following values for the same critical temperature:

\[
T_c = 2.26921(1) \quad \text{for CTMRG, and}
\]

\[
T_c = 2.269194(9) \quad \text{for FPCM,}
\]

(59)  

(60)

which are very close to the exact value of

\[
T_c^{\text{exact}} = 2.26918531...
\]  

(61)

By using the fit of the form of Eq. (58), I assumed that I did not know the exact value of \(\nu\), so I used \(\nu\) as a free parameter. From the fit I obtained the following values for this critical exponent:

\[
\nu = 1.05(8) \quad \text{for CTMRG, and}
\]

\[
\nu = 1.01(6) \quad \text{for FPCM,}
\]

(62)  

(63)

which are in agreement with the exact value \(\nu = 1\).
The pseudo-critical temperature as a function of the inverse of the correlation length for $\chi = 20$ to $\chi = 100$ with steps of 4. The data is fitted with a curve of the form of Eq. 58. The exact critical temperature is indicated by the dashed lines.

Figure 32: The pseudo-critical temperature as a function of the inverse of the correlation length for $\chi = 20$ to $\chi = 100$ with steps of 4. The data is fitted with a curve of the form of Eq. 58. The exact critical temperature is indicated by the dashed lines.
6 Results for the 6-state clock model

After having investigated the Ising model in the previous chapter, I now turn to the more challenging $q$-state clock model, which is a generalization of the Ising model ($q = 2$). This model is very interesting, because it can undergo two phase transition at temperatures $T_1$ and $T_2$, when $q$ is sufficiently large ($q \geq 5$). These transitions are of a special type, and are called Berezinskii-Kosterlitz-Thouless (BKT) transitions. The work on this transition type led to the Nobel Prize in Physics in 2016 [13], which was awarded to Thouless, Kosterlitz and Haldane. In this chapter, I first give the definition and properties of the $q$-state clock model. Then I investigate the clock model in the case of $q = 6$. I give the results of simulations in which I calculate thermodynamic quantities, and I determine the two critical temperatures. In the previous chapter, we have seen that FPCM performs better than CTMRG, as the thermodynamic limit is approached faster within a certain accuracy. Therefore, I use FPCM to get results for the 6-state clock model.

6.1 Definition of the $q$-state clock model

I investigate the $q$-state clock model, again on a two-dimensional square lattice. On each lattice site $i$ sits a spin, which can be in $q$ different spin states, with spin values values $s_i \in \{1, 2, \ldots, q\}$. The spin states are represented by arrows, with an angle $\theta_{s_i} = 2\pi (s_i - 1)/q$ to the $y$-axis, that are uniformly distributed over a circle. This is shown in Fig. 33 in the case of $q = 6$. I also show in this figure how I choose my coordinate frame in spin space. The spin directions lie in the $xy$-plane, and the first spin state is aligned along the $y$-axis.

Figure 33: The possible spin values $s_i$ for each site $i$ in the 6-state clock model. The spin values are uniformly distributed over a circle, which results to discrete angles $\theta_{s_i}$ to the $y$-axis. On the right is indicated how the coordinate frame in spin space is chosen.
The Hamiltonian of the clock model is defined by

$$H\{\{s_i\}\} = -J \sum_{\langle i,j \rangle} \cos(\theta_{s_i} - \theta_{s_j}) - h \sum_i \cos(\theta_{s_i}). \quad (64)$$

Just like the Hamiltonian of the Ising model in Eq. 40, the first term in Eq. 64 describes the interactions between all neighboring sites, which depend on the relative angle between their spin directions and the coupling strength $J$. We use $J > 0$, so the ground state is ferromagnetic as in Fig. 24b. This ground state is $q$-fold degenerate, as there are $q$ directions in which the system can magnetize. The second term in Eq. 64 describes the interaction between each spin and the external field $h$, which is pointing in the positive $y$-direction. Changing the sign of $h$ is equivalent to rotating the field by 180 degrees in the $xy$-plane. In the simulations, we only look at the case where $J = 1$ and $h = 0$. From now on, we will therefore use the following Hamiltonian for the $q$-state clock model:

$$H\{\{s_i\}\} = -\sum_{\langle i,j \rangle} \cos(\theta_{s_i} - \theta_{s_j}). \quad (65)$$

### 6.1.1 Tensor network representation

The function $f$ in the Boltzmann weight matrix from Eq. 10, that is needed to write the partition function as a tensor network, has the following form for the $q$-state clock model:

$$f(s_i, s_j) = -J \cos(\theta_{s_i} - \theta_{s_j}) \quad (66)$$

$$= -J \cos(2\pi(s_i - s_j)/q). \quad (67)$$

With $q = 6$, the Boltzmann weight matrix has then the following representation:

$$Q = \begin{pmatrix}
    e^{\beta} & e^{-\beta/2} & e^{-\beta/2} & e^{-\beta} & e^{-\beta/2} & e^{\beta/2} \\
    e^{\beta/2} & e^{\beta} & e^{-\beta/2} & e^{-\beta} & e^{-\beta/2} & e^{\beta/2} \\
    e^{-\beta/2} & e^{\beta/2} & e^{\beta} & e^{-\beta/2} & e^{-\beta} & e^{-\beta/2} \\
    e^{-\beta} & e^{-\beta/2} & e^{\beta/2} & e^{\beta} & e^{-\beta/2} & e^{-\beta} \\
    e^{-\beta/2} & e^{-\beta} & e^{-\beta/2} & e^{\beta/2} & e^{\beta} & e^{-\beta/2} \\
    e^{\beta/2} & e^{-\beta/2} & e^{-\beta/2} & e^{\beta/2} & e^{\beta} & e^{\beta} 
\end{pmatrix}. \quad (68)$$

With this $Q$ we can represent the partition function as a tensor network, as explained in section 3.1.
6.1.2 The $q \to 2$ and $q \to \infty$ limits

We see that the Hamiltonian of the clock model reduces to the Hamiltonian of the Ising model when we take the lowest nontrivial limit $q \to 2$. The spins can in this case either point in the positive or negative $y$-direction. Due to its discrete $\mathbb{Z}_2$ symmetry breaking, the Ising model can undergo a phase transition from a disordered to an ordered phase. The Hamiltonian of the $q$-state clock model has a discrete $\mathbb{Z}_q$ symmetry, since we can rotate each spin by the same multiple of $2\pi/q$ without changing the total energy of the system. Therefore, we would expect that the system undergoes a phase transition like in the Ising model, with a $q$-fold degenerate ground state in the ordered phase.

If we, however, take the limit $q \to \infty$, we see that the discrete $\mathbb{Z}_q$ symmetry becomes the continuous $U(1)$ symmetry. Instead of pointing in discrete angles, the spins can point in any direction of the $xy$-plane, so the model approaches the XY-model in this limit. We know from the Mermin-Wagner theorem, that continuous symmetries cannot be spontaneously broken at a finite temperature in systems with short-range interactions and lattice dimension smaller than 3 [27]. That means that the infinite two-dimensional XY-model cannot be ordered in the absence of an external field. An Ising-like phase transition is therefore not possible, but the XY model appears to have another kind of phase transition, which is of topological origin. This special phase transition is called the BKT-transition.

6.1.3 The Berezinskii-Kosterlitz-Thouless transition

The BKT-transition was discovered by Vadim Berezinskii, John M. Kosterlitz and David J. Thouless [12, 11], who investigated two-dimensional systems with a continuous symmetry. Kosterlitz and Thouless showed that it was possible to define a long-ranged topological order in the XY-model, and that a phase transition could happen, characterized by a sudden change of the response of the system to an external perturbation.

The origin of the BKT-transition is the role of vortices. The spin configuration of the two-dimensional square-lattice XY-model can be described by a spin field $\theta(\mathbf{r})$, which gives the spin angle of the spin at position $\mathbf{r}$ [11]. The system is seen now as a continuum, which is accomplished by making the lattice-spacing infinitesimally small. A vortex is a stable spin configuration in which the spin field rotates around a certain point with a (nonzero) winding number. This winding number is called the charge of the vortex. We speak of an anti-vortex, when the charge is negative. The lowest energy vortices and anti-vortices have charge +1 and -1 respectively. These are shown in Fig. 34. Vortices correspond to topological defects in the spin field, because $\theta(\mathbf{r})$ is not defined in the center of the vortices.
A vortex can bind with an anti-vortex to become a pair. The energy of such a configuration is lower than of a configuration in which the vortices are unbound. The free energy of an isolated (anti-)vortex with unit charge is approximately given by

$$F = E - TS \approx 2(\pi - T) \log(L/r_0),$$

where $L$ is the linear size of the system, and $r_0$ is the lattice spacing. We see again a competition between the energy and entropy term. At low temperatures, the energy term dominates, and it more favorable to have bound vortices instead of isolated vortices. But when the temperature is increased to a certain point, the entropy term dominates, and free (anti-)vortices are favorable as it minimized the free energy. So we witness a phase transition at a critical temperature $T_{BKT}$, from a phase with bound vortices to a phase with free vortices.

In the BKT-transition, there is no symmetry breaking, so the magnetization remains zero when the system goes from one phase to the other. There, however, is critical behavior in the correlation length and magnetic susceptibility. Near $T_{BKT}$ the correlation length diverges exponentially, according to:

$$\xi \sim \exp(b|t|^{\frac{1}{2}}) \quad \text{if } t > 0$$

$$= \infty \quad \text{if } t < 0,$$

where $t = (T - T_{BKT})/T_{BKT}$. The constant $b$ is a non-universal constant, which has value $b \approx 1.5$ for the XY model. The phase below $T_{BKT}$, where $\xi$ is infinity, is called the massless phase. The correlation function decays algebraically in that phase. So instead of a single critical temperature, the massless phase is actually a finite critical region.
The magnetic susceptibility diverges, when going to the massless phase, according to:

\[
\chi_m \sim \xi^{2-\eta} \quad \text{if } t > 0 \\
= \infty \quad \text{if } t < 0,
\]

where \( \eta \) is a critical exponent, which has value \( \eta = \frac{1}{4} \) for the XY model. This is the same value as for the two-dimensional Ising model.

For finite \( q \), the spins in the \( q \)-state clock model cannot point in all directions of the \( xy \)-plane like in the XY-model. But the model appears to be able to undergo a BKT-transition for sufficiently large \( q \), so if \( q \geq q_c \). There are discussions about whether \( q_c \) equals 5 or 6, but in this chapter we look at the model where \( q = 6 \), in which case it is commonly agreed that two BKT-transitions occur. Although the Hamiltonian of the 6-state clock model has no continuous symmetry, the system shows a \( U(1) \) symmetry in the massless phase [15].

### 6.1.4 Phase diagram

When \( q = 6 \), the clock model can undergo a transition from a disordered to a massless phase. But, since \( q \) is finite, the Hamiltonian has a discrete symmetry, so the Mermin-Wagner theorem does not prevent the creation of a symmetry-broken phase. Below a certain temperature, we expect the system to magnetize in one of the six possible spin directions. So the 6-state clock model has three different phases; the ordered, massless and disordered phase [30]. These phases are separated by transition temperatures \( T_1 \) and \( T_2 \), which are both of BKT-type. I define \( T_1 \) as the transition temperature from the ordered to the massless phase, and \( T_2 \) as the transition temperature from the massless to the disordered phase. The phase diagram is shown in Fig. 35.

![Phase diagram of the 6-state clock model. The symmetry of the spin configuration in the system is indicated in each phase.](image)

Figure 35: Phase diagram of the 6-state clock model. The symmetry of the spin configuration in the system is indicated in each phase.
6.1.5 Transition temperatures from literature

The 6-state clock model has no exact solutions, but we can compare our results with results from literature. Some values for the transition temperatures $T_1$ and $T_2$ from literature are shown in table 1. These temperatures are roughly $T_1 \sim 0.7$ and $T_2 \sim 0.9$.

The first three of these results are obtained by Monte Carlo simulations. Krčmár et al. [31] used CTMRG with finite-size scaling in order to obtain the transition temperatures. Chen et al. [32] used a tensor renormalization group method, based on higher-order singular value decomposition, (HOTRG) to calculate a gauge invariant quantity that can indicate the transition temperatures.

<table>
<thead>
<tr>
<th>Method</th>
<th>$T_1$</th>
<th>$T_2$</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Challa &amp; Landau(1986)[33]</td>
<td>0.68(2)</td>
<td>0.92(1)</td>
<td>Monte Carlo</td>
</tr>
<tr>
<td>Tomita &amp; Okabe (2002)[34]</td>
<td>0.7014(11)</td>
<td>0.9008(6)</td>
<td>Monte Carlo</td>
</tr>
<tr>
<td>Kumano et al. (2013)[15]</td>
<td>0.700(4)</td>
<td>0.904(5)</td>
<td>Monte Carlo</td>
</tr>
<tr>
<td>Krčmár et al. (2016)[31]</td>
<td>0.70</td>
<td>0.88</td>
<td>CTMRG</td>
</tr>
<tr>
<td>Chen et al. (2017)[32]</td>
<td>0.62658(5)</td>
<td>0.8804(2)</td>
<td>HOTRG</td>
</tr>
<tr>
<td>This work</td>
<td>0.71(1)</td>
<td>0.906(3)</td>
<td>Finite-$\chi$ scaling</td>
</tr>
<tr>
<td>This work</td>
<td>0.73(2)</td>
<td>0.910(8)</td>
<td>$\xi$-extrapolation</td>
</tr>
</tbody>
</table>

Table 1: Values for the transition temperatures of the 6-state clock model from literature.

6.2 Analytic results

6.2.1 Singular value spectrum

In Fig. 36 I plotted the singular value spectrum of the environment tensor $C$ of the 6-state clock model for $\chi = 60$ and three different temperatures. The first temperature is below the critical temperatures $T_1$ and $T_2$, the second one is between them, and the third one above them. The spectra decay slower than the spectra of the Ising model, from which we can conclude that larger bond dimensions are needed to get results within a similar accuracy. Although the singular values are not as degenerate as in the case of the Ising model, we still observe an irregular decay where the singular values lay on plateaus.
6.2.2 Magnetization

The 2-state clock model (Ising model), has one axis along which it can magnetize. When $q > 2$, the magnetization of the $q$-state clock model is not confined to a single line, but to multiple lines in the $xy$-plane. So the system can have a finite magnetization in both the $x$-direction and $y$-direction. The magnetization per site can therefore be described by a vector $\vec{m}$ in the $xy$-plane, and its components in the $x$- and $y$-direction are given by

$$m_x = \langle \sin(\theta_s) \rangle, \quad \text{and}$$

$$m_y = \langle \cos(\theta_s) \rangle,$$

where $s_r$ is the spin value of site $r$ in the lattice. In our simulations we can calculate these quantities with the use of Fig. 12, by defining $b_{ijkl}$ as

$$b_{ijkl}^{(x)} = \sum_{n=1}^{q} \sin \left( \frac{2\pi n}{q} \right) \left( \sqrt{Q} \right)_i \left( \sqrt{Q} \right)_j \left( \sqrt{Q} \right)_k \left( \sqrt{Q} \right)_l,$$

and

$$b_{ijkl}^{(y)} = \sum_{n=1}^{q} \cos \left( \frac{2\pi n}{q} \right) \left( \sqrt{Q} \right)_i \left( \sqrt{Q} \right)_j \left( \sqrt{Q} \right)_k \left( \sqrt{Q} \right)_l,$$

where

Figure 36: The singular value spectrum of the $C$ tensor with bond dimension $\chi = 60$ for three temperatures ($T < T_1$, $T_1 < T < T_2$, and $T > T_2$). These results are obtained by 2000 FPCM iterations.
For the 6-state clock model, I can use the magnitude of the magnetization ($|\vec{m}| = \sqrt{m_x^2 + m_y^2}$) as an order parameter. In Fig. 37, I have plotted the order parameter as a function of temperature for different bond dimensions. We can identify three different phases in this figure. Starting from $T = 0$, we see that the order parameter is 1, so the system is in the ordered phase like in the Ising model. Then, around some temperature $T_1$, the order parameter drops, but remains finite and decreases linearly with temperature. This is the massless phase, in which we expect no finite magnetization at all in the thermodynamic limit. Although the results show a finite magnetization in this phase, we see that the magnitude of the magnetization decreases when $\chi$ is increased. In the thermodynamic limit ($\chi \to \infty$), the magnetization is zero as expected. In the figure we also see that, around some temperature $T_2$, the magnetization abruptly drops to zero and remains zero. From $T_2$, the system is in the disordered phase, where the entropy dominates over the internal energy. The value of $T_2$ is dependent on $\chi$, just like we have seen for the Ising model.

![Figure 37: The absolute value of the magnetization per site as a function of temperature for different bond dimensions $\chi$. A convergence tolerance of $10^{-6}$ is used.](image)

With fixed boundary conditions in which all boundary spins are in the first spin state, like in Eq. 34 and 35, we expect that the system only magnetizes in the direction of the first spin state as we have given a preference to this state. The first spin state is in the positive $y$-direction, so we would expect $m_y$ to have a value between 0 and 1, and $m_x$ to be zero in all cases. If this works, we could use $m_y$ as an order parameter, instead of $|\vec{m}|$. It, however, turns out that the magnetization can get a component in the $x$-direction when we run simulations.
To investigate this behavior, I have plotted $m_x$ and $m_y$ as function of temperature in Fig. 38 for $\chi = 20$, but this time I used random initial environment tensors. We expect that for temperatures below $T_1$, the magnetization points in one of the six directions corresponding to each spin state. I use small temperature steps, and at each step one random state of the six states is favored, because of the random boundary conditions. In the massless phase between $T_1$ and $T_2$, we expect the magnetization to point in a random direction in the $xy$-plane, due to the $U(1)$ symmetry. In the disordered phase above $T_2$, we expect the magnetization to be zero. The expectations for the ordered and disordered phase are verified by Fig. 38, but we see something different happening in the massless phase. Instead of pointing in random directions, the magnetization follows some special lines. These lines form a pattern, which is symmetric along the axes parallel to the six spin directions. The magnitude of the magnetization, however, is the same as for fixed boundary conditions.

Figure 38: The magnetization per site as a function of temperature for bond dimension $\chi = 20$, using random boundary conditions. A convergence tolerance of $10^{-6}$ is used. The top left shows the $x$-component, the top right shows the $y$-component, the bottom left shows the absolute value, and the bottom right show the data projection onto the $xy$-plane.
In Fig. [39] I have plotted $m_y$ in the massless phase for bond dimension $\chi = 18$ to $\chi = 23$. We see that the pattern of the magnetization changes quite much when the bond dimension is increased by 1. Therefore it seems that this pattern is an effect, caused by the truncation of the singular values spectrum of the environment tensor $C$, which is regulated by $\chi$. Keeping only the $\chi$ largest singular values instead of $\chi + 1$ singular values can make a difference in which spins states are favored.

![Figure 39: The $y$-component of the magnetization per site as a function of temperature for bond dimensions $\chi = 18$ to $\chi = 23$, using random boundary conditions. A convergence tolerance of $10^{-6}$ and temperature steps of size $10^{-4}$ are used. The five last figures have the same scale as the first figure.](image)
For a temperature in the massless phase, the magnetization points in special directions, that are dependent on $\chi$. In Fig. 40, I have plotted the $x-$ and $y$-components of the magnetization for bond dimensions ranging from $\chi = 10$ to $\chi = 80$ with steps of 10. The simulations are done at temperature $T = 0.8$, which is assumed to be somewhere in the middle of the massless phase. For each value of $\chi$, I calculate the magnetization 200 times with random boundary conditions. From the figure we see that, for all $\chi$, the magnetization points in a limited number of directions that are symmetric along the axes parallel to the six possible spin directions. This number of directions is dependent on $\chi$.

Figure 40: The $x$- and $y$-component of the magnetization per site at $T = 0.8$, for bond dimensions $\chi = 10$ to $\chi = 80$ with steps of 10. For each $\chi$, 200 simulations are done with random boundary conditions and a convergence tolerance of $10^{-5}$. The dash-dotted lines show the 6 possible spin directions and the dotted lines show the axes exactly in between them.
On the left side of Fig. 41, I have shown the same plot as in Fig. 40 but with the bond dimension ranging from $\chi = 10$ to $\chi = 80$ with steps of 1. On the right side of the figure, I projected the data onto the unit circle in order to look at the angle-dependence of the magnetization. From this figure, we can see that for each individual choice of $\chi$ there is no $U(1)$ symmetry, but this symmetry emerges when we look at a collection of different $\chi$. This result confirms the presence of a massless phase.

Figure 41: The left figure shows the $x$- and $y$-component of the magnetization per site at $T = 0.8$, for bond dimensions $\chi = 10$ to $\chi = 80$ with steps of 1. For each $\chi$, 200 simulations are done with random boundary conditions and a convergence tolerance of $10^{-5}$. The dash-dotted lines show the 6 possible spin directions and the dotted lines show the axes exactly in between them. The right figure shows the data of the left figure, projected onto the unit circle.
6.2.3 Correlation length

In Fig. 42 I have plotted the correlation length as a function of temperature for different bond dimensions. Just like in the Ising model, the correlation length is close to zero in the ordered and disordered phase, and diverges at the critical point. In the 6-state clock model, there is not one critical point, but a critical region between $T_1$ and $T_2$, where the correlation length is infinite. Fig. 42 shows no divergence, but the correlation length becomes large in the critical region. The maximum of the correlation length is finite, because of the finite $\chi$, and becomes higher when $\chi$ is increased.

We see that, in the critical region, the correlation length shows some irregular behavior. This effect is caused by the convergence behavior in the massless phase. The measure of convergence $\epsilon$, defined in Eq. 38, does not decrease monotonically as a function of iteration number, but can at some points increase again. Therefore, the simulation can be terminated before the environment tensors are really converged. A way to overcome this problem is to use another measure of convergence. In each iteration, our measure compares the singular value spectrum of the $C$ tensor with the spectrum obtained one iteration before. Instead, we could compare it with the spectrum obtained more (e.g. 10) iterations before. Then, there is a smaller chance for the measure to get stuck on a plateau or in a local minimum.

![Figure 42: Correlation length as a function of temperature for different bond dimensions $\chi$. A convergence tolerance of $10^{-6}$ is used.](image-url)
6.2.4 Determination of transition temperatures by finite-\(\chi\) scaling

In order to determine the transition temperatures \(T_1\) and \(T_2\), we can use finite-\(\chi\) scaling, just like we did for the Ising model. The correlation length diverges at the BKT-transition with the following form:

\[
\xi \sim \exp(b|T - T_c|^{-\frac{1}{2}})
\]  

(78)

Given Eq. [78] we expect that the correlation length scales with a characteristic temperature \(T^*_c(\chi)\) as follows:

\[
\xi(\chi(T^*_c(\chi))) \sim \exp(b|T^*_c(\chi) - T_c|^{-\frac{1}{2}}),
\]  

(79)

which can be rewritten as:

\[
T^*_c(\chi) = T_c + c_1 \log[c_2 \xi(\chi(T^*_c(\chi)))]^{-2},
\]  

(80)

where \(c_1\) and \(c_2\) are constants. In the case of the Ising model, I used the temperature at which the correlation length \(\xi\) has its maximum as the characteristic temperature. In the 6-state clock model, the correlation length has two transition temperatures, but only one maximum in the correlation length, so we have to use another characteristic temperature.

Looking at the absolute value of the magnetization, we see that the slope has a local minimum at two temperatures, which approach the exact transition temperatures in the thermodynamic limit. Therefore I define \(T^*_1(\chi)\) as the temperature where the slope of the magnetization is the steepest when going from the ordered to the massless phase. Likewise, I define \(T^*_2(\chi)\) as the temperature where the slope of the magnetization is the steepest when going from the massless to the disordered phase.

For \(\chi = 20\) to \(\chi = 100\), I have determined \(T^*_1(\chi)\) and \(T^*_2(\chi)\), and the values of the correlation length at those temperatures; \(\xi(\chi(T^*_1(\chi)))\) and \(\xi(\chi(T^*_2(\chi)))\). Then I define the following length variable:

\[
l = \log[c_2 \xi(\chi(T^*_c(\chi)))]^2
\]  

(81)

Using this length variable, \(T^*_c(\chi)\) in Eq. [80] becomes linear in \(1/l\). In Fig. 43 I have plotted \(T^*_c(\chi)\) versus \(1/l\) for both transition temperatures. The data is linear, which confirms the scaling relation in Eq. [79] Using a linear fit, I obtain the following values for the two transition temperatures:

\[
T_1 = 0.71(1), \quad \text{and}
\]

(82)

\[
T_2 = 0.906(3).
\]

(83)
Figure 43: The characteristic temperature as a function of the inverse of length variable $l$, which is defined in Eq. 81. The simulations are done using bond dimensions ranging from $\chi = 20$ to $\chi = 100$, and a convergence tolerance of $10^{-6}$. Using a linear fit on the data, the critical temperature is determined in the limit $1/l \to 0$. 

$T_1^*$ vs. $1/l$ and $T_2^*$ vs. $1/l$ are shown in the figure.
6.2.5 Determination of transition temperatures by ξ-extrapolation

Another method to determine the transition temperatures, is to determine the correlation length $\xi_\chi$ for increasing bond dimensions, and to extrapolate it to get the true correlation length $\xi$ in the thermodynamic limit. On the obtained $\xi$, we can use a fit of the form of Eq. 78 in order to find the transition temperatures.

The problem with this method is that it is not easy to extrapolate the correlation length, especially near the critical temperature. Following the reasoning in Ref. [16], the correlation function, calculated for an MPS with a finite bond dimension, diverges purely exponentially with distance at the critical point:

$$C(r) \sim \exp(-r/\xi_\chi),$$  \hspace{1cm} (84)

where $r$ is the distance between two lattice sites, and $\xi_\chi$ is the correlation length for an MPS with a finite bond dimension $\chi$. It is, however, expected that the true correlation function diverges with an extra algebraic factor:

$$C(r) \sim r^{-\eta} \exp(-r/\xi),$$  \hspace{1cm} (85)

where $\eta$ is a constant. This form is also seen in the Ornstein-Zernike formula for the correlation function in Ising-like models. An MPS with finite bond dimension is not able to capture this algebraic factor, because the eigenvalue value spectrum of the row-to-row transfer matrix has to be continuous. Extrapolating $\xi_\chi$ with $\chi$ as a refinement parameter not only underestimates the true correlation length, but is also inaccurate because of the irregular behavior due to the underlying singular value spectrum.

In Ref. [16], this problem is overcome by using another refinement parameter than $\chi$. If we denote the $k^{(th)}$ largest eigenvalue of the row-to-row transfer matrix of Fig. 12c by $\lambda_k$, we can define the following quantities;

$$\epsilon_k = -\log \lambda_k.$$  \hspace{1cm} (86)

Using Eq. 24 we have the following relation between $\xi_\chi$ and $\epsilon_k$:

$$1/\xi_\chi = \epsilon_2 - \epsilon_1 = \epsilon_2,$$  \hspace{1cm} (87)

where I normalized the spectrum such that the largest eigenvalue is $\lambda_1 = 1$, so $\epsilon_1 = 0.$
Figure 44 shows the idea behind the extrapolation scheme. It shows the dominant part of the transfer matrix spectrum, which is discrete because of the finite bond dimension. The algebraic part of the asymptotic decay of the correlation function is recovered by the continuous band in the spectrum. The exact correlation length is set by the distance between $\epsilon_1 = 0$ and the bottom of the band. Using a finite $\chi$, the discrete spectrum can only approximate the continuous band. In the $\chi \to \infty$ limit, the spectrum becomes continuous, and the difference between the second and third eigenvalue becomes zero. We call this difference $\delta$, which is defined as

$$\delta = \epsilon_3 - \epsilon_2.$$ (88)

Instead of $\chi$, we use $\delta$ as a refinement parameter. If the first eigenvalues of the transfer matrix are degenerate, or become degenerate when $\chi$ is increased, we can use the following refinement parameter:

$$\delta = \epsilon_k - \epsilon_2,$$ (89)

where $\epsilon_2, \ldots, \epsilon_{k-1}$ are degenerate or near degenerate. For the 6-state clock model, I use the refinement parameter in Eq. 89 with $k = 5$.

![Figure 44](image)

Figure 44: The eigenvalue spectrum $\{\epsilon_1, \epsilon_2, \epsilon_3, \ldots\}$ of the transfer matrix in Fig 12c with finite bond dimension $\chi$. The discrete spectrum (indicated by red lines) approximate the continuous band of the exact spectrum (indicated by the blue band). The correlation length $\xi_\chi$ is set by the difference between $\epsilon_1$ and $\epsilon_2$. The exact correlation length is set by the gap between $\epsilon_1$ and the continuous band. The parameter $\delta = \epsilon_3 - \epsilon_2$ becomes zero in the $\chi \to \infty$ limit. This picture is edited from Ref. [16].

I want to plot the exact correlation length $\xi$ as a function of temperature. For each temperature, I determine $\xi_\chi$ and $\delta$ for increasing bond dimensions, ranging from $\chi = 20$ to $\chi = 100$. In Fig. 45 I have plotted $1/\xi_\chi$ as a function of $\delta$ for temperature $T = 0.675$, in which we see a linear relation. Using a linear fit, we find the true $\xi$ in the limit $\delta \to 0$. When doing this for a range of temperatures near the transition temperatures, I can plot $\xi$ versus $T$, as shown in Fig. 46. The obtained data are fitted with a function of the form of Eq. 78, which gives us the values for the transition temperatures.
Figure 45: Extrapolation of the correlation length at temperature $T = 0.675$. For bond dimensions $\chi = 20$ to $\chi = 100$, the inverse correlation length $1/\xi_\chi$ is plotted versus the refinement parameter $\delta$, which is defined in Eq. 89 with $k = 5$. A convergence tolerance of $10^{-6}$ is used. The data is fitted with a linear curve in order to get the extrapolated correlation length at $\delta = 0$.

Figure 46: The extrapolated correlation length as a function of temperature. Each data point is obtained using bond dimension $\chi = 20$ to $\chi = 100$ and a convergence tolerance of $10^{-6}$. The data is fitted with a function of the form of Eq. 78. The transition temperatures $T_1$ and $T_2$, obtained from the fit, are indicated by the dashed lines.
One difficulty arises here, as Eq. 78 is only valid near the critical point. So it not clear how far away from \( T_1 \) and \( T_2 \) we can start our temperature interval and expect Eq. 78 to hold. Figure 47 shows how the values for \( T_1 \) and \( T_2 \) change as we change the start point and endpoint of the temperature interval that we use for the fit respectively.

![Figure 47: Determined values for the transition temperatures \( T_1/T_2 \) as function of the chosen begin/endpoint of the temperature interval used for the fit. In the top figure, the endpoint of the interval is fixed to \( T_{\text{end}} = 0.68 \), and the start point is varied from \( T_{\text{start}} = 0.59 \) to \( T_{\text{start}} = 0.668 \) with steps of \( 10^{-3} \). \( T_1 \) is obtained for each interval like in Fig. 46. In the bottom figure, the start point of the interval is fixed to \( T_{\text{start}} = 0.94 \), and the endpoint is varied from \( T_{\text{end}} = 0.953 \) to \( T_{\text{start}} = 1.02 \) with steps of \( 10^{-3} \). \( T_2 \) is obtained for each interval like in Fig. 46.](image)

From Fig. 47 we see that the values for the transition temperatures become higher when temperatures further away from the critical region are excluded from the fit. The error on these values become bigger as well, which makes sense, as the extrapolated correlation length is less accurate near the critical region and less data are used for the fit. The values for the transition temperatures fall in a certain range, so from the results in Fig. 47 we conclude:

\[
T_1 = 0.73(2), \quad \text{and} \quad (90)
\]

\[
T_2 = 0.910(8). \quad (91)
\]
7 Results for the 5-state clock model

Using FPCM, we have obtained good results for the 6-state clock model. We have confirmed the presence of the massless phase, and determined the two transition temperatures. Now we turn to the 5-state clock model, in which case there is no consensus about the nature of the transitions. Some studies conclude that both transitions are of BKT-type, and other studies conclude that the second one cannot be of BKT-type. In this chapter, we use the same analysis that we used for the 6-state clock model in order to study the transitions of the 5-state clock model. From the results, we conclude that the 5-state clock model has the same phase diagram as the 6-state clock model, and that the two transitions are of BKT-type.

7.1 Definition of the 5-state clock model

In the 5-state clock model, each lattice site \( i \) can be in 5 possible spin states, with values \( s_i \in \{1, 2, 3, 4, 5\} \). These are represented by arrows, with an angle \( \theta_{s_i} = 2\pi(s_i - 1)/5 \) to the \( y \)-axis, that are uniformly distributed over a circle, see Fig 48. We again choose the coordinate frame in which the spin directions are confined to the \( xy \)-plane, and the first spin state points along the positive \( y \)-axis. The Hamiltonian of the model is given by Eq. 65 which is

\[
H(\{s_i\}) = -\sum_{\langle i,j \rangle} \cos(\theta_{s_i} - \theta_{s_j}),
\]

where the sum goes over all pairs of nearest neighboring sites \( i \) and \( j \).

![Figure 48: The possible spin values \( s_i \) for each site \( i \) in the 5-state clock model. The spin values are uniformly distributed over a circle, which results in discrete angles \( \theta_{s_i} \) to the \( y \)-axis. On the right is indicated how the coordinate frame in spin space is chosen.](image)
7.1.1 Tensor network representation

In order to represent the partition function of the 5-state clock model as a tensor network, we have to use the following form of the Boltzmann weight matrix:

\[
Q = \begin{pmatrix}
e^{\beta a} & e^{\beta b} & e^{\beta a} & e^{\beta b} \\
e^{\beta a} & e^{\beta} & e^{\beta a} & e^{\beta} \\
e^{\beta b} & e^{\beta a} & e^{\beta} & e^{\beta a} \\
e^{\beta a} & e^{\beta b} & e^{\beta b} & e^{\beta a}
\end{pmatrix},
\]

(93)

where \(a = (\sqrt{5} - 1)/4\) and \(b = -(\sqrt{5} + 1)/4\).

7.1.2 Transition temperatures from literature

There are no exact solutions for the 5-state clock model, but we can compare our results again with results from literature. Some values for the transition temperatures \(T_1\) and \(T_2\) from previous studies are shown in table 2. These values are roughly determined to be \(T_1 \sim 0.9\) and \(T_2 \sim 0.95\).

It is established that the \(q\)-state clock model has a massless phase when \(q \geq 5\) [14], which is bounded by two transition temperatures [35], but Baek and Minnhagen showed that \(T_2\) is not of BKT-type [36]. They used the helicity modulus [37] in order to get to this conclusion, as the helicity modulus should show a universal jump at a BKT-transition [38], which was not found in their research. Kumano et al. claimed, however, that Baek and Minnhagen used the wrong definition for the helicity modulus [15]. They tried a definition that took into account the \(Z_q\) symmetry, and used Monte Carlo simulations to show that there really is a jump at \(T_2\).

<table>
<thead>
<tr>
<th>Method</th>
<th>(T_1)</th>
<th>(T_2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Borisenko et al. (2011) [39]</td>
<td>0.9051(9)</td>
<td>0.9515(9)</td>
</tr>
<tr>
<td>Kumano et al. (2013) [15]</td>
<td>0.908</td>
<td>0.944</td>
</tr>
<tr>
<td>Chatelain (2014) [40]</td>
<td>0.914(12)</td>
<td>0.945(17)</td>
</tr>
<tr>
<td>This work</td>
<td>0.92(1)</td>
<td>0.950(2)</td>
</tr>
<tr>
<td>This work</td>
<td>0.930(9)</td>
<td>0.951(7)</td>
</tr>
</tbody>
</table>

Table 2: Values for the transition temperatures of the 5-state clock model from literature.
7.2 Analytic results

7.2.1 Singular value spectrum

Figure 49 shows the singular value spectrum for the environment tensor $C$ with bond dimension $\chi = 60$. I look again at three different temperatures, where the first is expected to be in the ordered phase, the second in the massless phase, and the third in the disordered phase. The spectra are very similar to the spectra that we have seen for the 6-state clock model. They both decay very slowly near the critical region, so a large bond dimension is needed to get accurate results. We again see that the singular values decay in an irregular fashion, and that the decay becomes smoother (but also slower), when the critical region is approached.

![Figure 49: The singular value spectrum of the $C$ tensor with bond dimension $\chi = 60$ for three temperatures ($T < T_1$, $T_1 < T < T_2$, and $T > T_2$). These results are obtained by 2000 FPCM iterations.](image-url)
7.2.2 Magnetization and correlation length

In the left side of Fig. 50, I have plotted the (absolute value of the) magnetization per site as a function of temperature, for different bond dimensions. We see the presence of the ordered phase at low temperatures, where the magnetization is one. We also see the presence of the disordered phase, where the magnetization is zero. Just as we have seen for the 6-state clock model, there is a region between the ordered and disordered phases, where the magnetization is finite and decreases linearly with temperature. Although this region is small, it indicates the presence of a massless phase.

The right side of Fig. 50 shows the correlation length as a function of temperature. In the ordered and disordered phases, the correlation length goes to zero, and it is finite in between. In the limit $\chi \to \infty$, the correlation length becomes infinite for a finite temperature region, which indicates the presence of a massless phase.

Figure 50: The absolute value of the magnetization per site as a function of temperature for different bond dimensions $\chi$. A convergence tolerance of $10^{-6}$ is used.
In order to study the massless phase, I calculate the $x$- and $y$-components of the magnetization per site, using random boundary conditions. In Fig. 51 I have plotted these components for $\chi = 20$. We again see a strange pattern, as the magnetization points at specific directions at each temperature. We have concluded before that this pattern is probably an effect caused by the truncation of the environment tensors, which is controlled by the bond dimension $\chi$.  

![Figure 51: The magnetization per site as a function of temperature for bond dimension $\chi = 20$, using random boundary conditions and a convergence tolerance of $10^{-6}$. The top left shows the $x$-component, the top right shows the $y$-component, the bottom left shows the absolute value, and the bottom right shows the data projection on the $xy$-plane.](image)
In Fig. 52 I have plotted the magnetization for bond dimension ranging from $\chi = 10$ to $\chi = 80$ with steps of 1. The simulations are done 100 times for each value of $\chi$, at $T = 0.93$, which is expected to be in the middle of the massless phase. After projecting the data onto the unit circle, we can see that the data is uniformly distributed, from which we can conclude that there are no preferred directions for the magnetization. So at this temperature, there is a $U(1)$ symmetry, which confirms the presence of a BKT-phase in the 5-state clock model.

![Figure 52: The left figure shows the x- and y-component of the magnetization per site at $T = 0.93$, for bond dimensions $\chi = 10$ to $\chi = 80$ with steps of 1. For each $\chi$, 100 simulations are done with random boundary conditions and a convergence tolerance of $10^{-5}$. The dash-dotted lines show the 5 possible spin directions and the dotted lines show the axes exactly in between them. The right figure shows the data of the left figure, projected onto the unit circle.](image-url)
7.2.3 Determination of transition temperatures by finite-$\chi$ scaling

If the two transitions in the 5-state clock model are of BKT-type, we can use the same scaling analysis to determine $T_1$ and $T_2$ as we did for the 6-state clock model in section 6.2.4. We define the characteristic temperature again as the temperatures at which the slope of the magnetization is the steepest.

The results are shown in Fig. 53. For both transition, the data show a scaling relation of the form of Eq. 80. So we can conclude that the transitions are of BKT-type. By fitting the data, we find the following values for the transition temperatures:

\[
T_1 = 0.92(1), \text{ and} \\
T_2 = 0.950(2).
\]

7.2.4 Determination of transition temperatures by $\xi$-extrapolation

We can also determine the transition temperatures by extrapolating the correlation length as described in section 6.2.5. As a refinement parameter, I again use $\delta$ in Eq. 89 with $k = 5$. In Fig. 54 I plotted the extrapolated correlation length as a function of temperature. Using a fit on the data, the transition temperatures are determined. As we do not know how far away from the critical region we can include data for the fit, we determine $T_1$ and $T_2$ for different choices for the begin and end-point of the temperature interval respectively. The results are shown in Fig. 55, from which we find the following values for the transition temperatures:

\[
T_1 = 0.930(9), \text{ and} \\
T_2 = 0.951(7).
\]
The characteristic temperature as a function of the inverse of length variable $l$, which is defined in Eq. $\text{[81]}$. The simulations are done using bond dimensions ranging from $\chi = 20$ to $\chi = 100$, and a convergence tolerance of $10^{-6}$. Using a linear fit on the data, the critical temperature is determined in the limit $1/l \to 0$. 

Figure 53: The characteristic temperature as a function of the inverse of length variable $l$, which is defined in Eq. $\text{[81]}$. The simulations are done using bond dimensions ranging from $\chi = 20$ to $\chi = 100$, and a convergence tolerance of $10^{-6}$. Using a linear fit on the data, the critical temperature is determined in the limit $1/l \to 0$. 

71
Figure 54: The extrapolated correlation length as a function of temperature. Each data point is obtained using
bond dimension $\chi = 20$ to $\chi = 100$ and a convergence tolerance of $10^{-6}$. The data is fitted with a function of the
form of Eq. 78. The transition temperatures $T_1$ and $T_2$, obtained from the fit, are indicated by the dashed lines.

Figure 55: Determined values for the transition temperatures $T_1/T_2$ as function of the chosen begin/endpoint of
the temperature interval used for the fit. In the top figure, the endpoint of the interval is fixed to $T_{end} = 0.9$, and
the start point is varied from $T_{start} = 0.8$ to $T_{start} = 0.88$ with steps of $10^{-3}$. $T_1$ is obtained for each interval
like in Fig. 54. In the bottom figure, the start point of the interval is fixed to $T_{start} = 0.98$, and the endpoint is
varied from $T_{end} = 1$ to $T_{start} = 1.06$ with steps of $10^{-3}$. $T_2$ is obtained for each interval like in Fig. 54.
8 Conclusions

In this thesis, we have studied two-dimensional classical spin models, by representing the partition function as a tensor network. To approximate this tensor network and to calculate thermodynamic quantities, we used the well-known Corner Transfer Matrix Renormalization Group (CTMRG) method, and the Fixed Point Corner Method (FPCM), which was designed more recently. Both methods approximated the partition function using the same tensor network ansatz, where the accuracy is controlled by the bond dimension $\chi$. The goals of this research were to look how the results and performance of both methods compare to each other, and to find results for models that do not yet have any rigorous numerical solutions.

In Chapter 5, we applied our methods to the Ising model in order to compare the speed and accuracies of the results obtained with CTMRG and FPCM. We studied the magnetization and free energy per site, and the results were compared to the exact solutions. Both methods are capable of producing results close to the exact solutions within the same accuracy. We conclude that FPCM can produce more accurate results in a less amount of time than CTMRG. Far away from the critical temperature, both methods are fast, but CTMRG is slightly faster than FPCM. Near the critical point, FPCM becomes roughly two times faster than CTMRG when we want to get results within the same accuracy. It is expected that this difference becomes more significant for systems with a larger spin space, as the leading computational cost for FPCM is of order $O(q^3\chi^3)$, as opposed to order $O(q^2\chi^3)$ for CTMRG.

A way to improve the FPCM algorithm could be to use other methods to solve the fixed point equations approximately. Besides the \textit{eigs} function, I also tried the Arnoldi algorithm [41], like Fishman \textit{et al.} used in their simulations [7]. This did, however, not improve the computation time. Another way to improve the performance of FPCM could be to develop a procedure to find better initial guesses for the environment tensors, instead of starting each simulation with environment tensors obtained by CTMRG iterations.

We also determined the critical temperature of the Ising model with both CTMRG and FPCM, by performing a finite-$\chi$ scaling analysis on the correlation length. We found the following values:

$$T_c^{\text{Ising}} = 2.26921(1) \quad \text{for CTMRG, and}$$
$$T_c^{\text{Ising}} = 2.269194(9) \quad \text{for FPCM.}$$

(98)
These values agree with the exact value of $T_c^{exact} = 2.26918531...$. The small error on the obtained values results from the fact that the exact solution for the magnetization is approximated very well for small bond dimensions. In other words, the pseudo critical temperatures $T_c^*(\chi)$ are very close to the exact critical temperature, even for small $\chi$.

After investigating the Ising model, we turned to the $q$-state clock model, which is a generalization of the $q = 2$ Ising model. When the number of possible spin states $q$ is large enough ($q \geq q_c$), the model has a finite critical region instead of a critical point. This region is the massless phase, and the two transitions to this phase are of Berezinskii-Kosterlitz-Thouless (BKT) type. From previous studies, we see that there is no consensus whether $q_c = 5$ or $q_c = 6$.

In Chapter 6, we used FPCM to get results of the 6-state clock model. The magnetization and correlation length showed the presence of three phases. In the massless phase, the magnetization confirmed the $U(1)$ symmetry, and the correlation length showed a divergence for a finite temperature interval. We determined the transition temperatures by performing a finite-$\chi$ scaling analysis on the correlation length, which values are given by:

$$T_{q=6}^1 = 0.71(1), \quad \text{and}$$
$$T_{q=6}^2 = 0.906(3).$$

We also determined the transition temperatures by using a recently introduced extrapolation scheme for the correlation length. With this analysis, we found the following values:

$$T_{q=6}^1 = 0.73(2), \quad \text{and}$$
$$T_{q=6}^2 = 0.910(8).$$

Our results are comparable to the values found in literature. The errors on the values obtained with the $\xi$-extrapolation are, however, determined in a crude way. These errors can possibly be reduced by using another analysis to determine how the temperature interval that we use for the fit affects the value for the transition temperatures.

Then we investigated the 5-state clock model in Chapter 7, where previous studies show no agreement about the nature of the phase transitions. We used the same analyses as for the 6-state clock model, and found similar results. For a finite temperature interval, the magnetization showed a $U(1)$ symmetry and the correlation length diverged, which indicated the massless phase. We studied the correlation length near the massless phase, and found that both phase transition were of BKT type.
With the finite-$\chi$ scaling analysis, we found the following values for the transition temperatures:

\begin{align}
T_1^{\chi=5} &= 0.92(1), \text{ and} \\
T_2^{\chi=5} &= 0.950(2). 
\end{align}

(104) \hspace{1cm} (105)

Using the extrapolation scheme for the correlation length, we found the following values:

\begin{align}
T_1^{\chi=5} &= 0.930(9), \text{ and} \\
T_2^{\chi=5} &= 0.951(7). 
\end{align}

(106) \hspace{1cm} (107)

Our values for the first transition temperature $T_1$ are a little higher than the values found in literature, for unknown reasons. The values for the second transition temperature $T_2$ do agree with the values found in literature.

The obtained values for the transition temperatures of the 5- and 6-state clock model have a significantly bigger error than the ones of the Ising model. We could have expected this, because the singular value spectrum decays more slowly for the 5- and 6-state clock model than for the Ising model. We used the same range of bond dimensions for all three models, which was from $\chi = 20$ to $\chi = 100$. To get more accurate results for the 5- and 6-state clock model, we should use bond dimensions that are even bigger.
References


