Computational Physics - Project 4

Johannes Scheller (candidate no. 71), Vincent Noculak (candidate no. 22)
Lukas Powalla (candidate no. 67), Richard Asbah (candidate no. 50)

November 13, 2015

Contents

Introduction

In project 4 we are dealing with the Ising model in two dimensions without an external magnetic field. We are looking at a lattice of L times L particles, where each particle has a spin value of ± 1. In order to compute different interesting properties of the system, we want to use the metropolis algorithm. With our computations, we want to calculate the expectation value of the Energy, the expectation value of the absolute value of the magnetisation, the heat capacity and susceptibility of the system as a function of temperature in order to study phase transitions of the system. In our case, we want to analyse, weather a phase transition from a ordered phase for low temperatures to a disordered phase for high temperatures above the critical Temperature T_c takes place. (in a canonical ensemble this means a second order phase transition and therefore a divergence of the heat capacity at the critical Temperature). In order to get familiar with all the quantities, we first wan to study the case of a 2 times 2 lattice. For This lattice-size, we want to find a analytical expression for all interesting physical properties of the system and compare them with the computed values. However, researcher have solved the two dimensional Ising model a long time ago for any and even for infinite size. The Ising model in two dimensions has been solved first for any "fixed" size by Kaufman in 1949 and in the end for a infinite size by Onsager in 1944. (compare with "Statistical Mechanics: Algorithms and Computations; Werner Krauth; published 2006; page 237") This project may also show the link from statistical physics to macroscopic properties of a given physical system, which is a interesting relation.

1 Theory

1.1 General properties of physical systems and their link to statistical physics

1.1.1 physical ensembles

Let us now look at a physical system and its surroundings. In principle, it is necessary to describe the relation of the physical system and its surroundings in order to determine the properties of the system.(sometimes this relations are related to physical boundary conditions) It is necessary to know whether we want to allow for instance particle/heat exchange or not. How we set up our system also defines us the thermodynamic potential, which can be used to describe the system.(e.g. Entropy, Helmholtz,Gibbs) All in all, we have the Micro canonical ensemble, the canonical ensemble, the Grand canonical ensemble and the pressure canonical ensemble. (in this case an ensemble means a collection of microscopic systems, compare with the lecture notes Computational physics 2015 at University of Oslo by Morten Hjorth-Jensen page 417) In the following, we will always deal with the canonical ensemble. This means that we don't allow particle exchange from the system with its surroundings, but we allow exchange of heat with the environment. Fixed variables are in this case the temperature, the total volume and the total particle number. The total energy of the canonical ensemble is not constant, because there can be heat exchange with the surroundings. The system, which does not allow heat exchange and does not allow particle exchange is called micro-canonical ensemble. (compare "Statistical Mechanics An Intermediate Course; 2nd Edition; G. Morandi/F. Napoli/ E. Ercolessi; page 94ff")

1.1.2 General properties of canonical ensembles

The canonical ensemble can be expressed by Helmholtz' free energy. The system strives to a minimum of Helmholtz' free energy, which is defined as follows:

$$F = -k_B T lnZ =< E > -TS \tag{1}$$

where the entropy S is given by

$$S = -k_B lnZ + k_B T \frac{\partial lnZ}{\partial T} \tag{2}$$

We can see that F depends on the expectation value of the Energy and on $-TS$. Hence, the canonical ensemble pursuits towards an energy minimum and higher entropy. This can be interpreted as a "struggle between two important principles in physics"(lecture notes Computational physics 2015 at University of Oslo by Morten Hjorth-Jensen page 419) The probability distribution for a canonical ensemble is given by the Boltzmann distribution.

$$P_i(\beta) = \frac{e^{-\beta E_i}}{Z} \tag{3}$$

$\beta = 1/k_B T$ where T is the temperature, k_B is the Boltzmann constant,E_i is the energy of micro state i and Z is the partition function for the canonical ensemble is the sum over all the micro states M.

$$Z = \sum_{i=1}^{M} e^{-\beta E_i} \tag{4}$$

After running the system for long time the canonical ensemble is uniquely determined and does not depend on the arbitrary choices of the initial temperature. The system uncertainty due the Energy fluctuations in the canonical

ensemble gives the variance of the energy.

from equation 1, 2 and probability distribution P_i

$$< E >= k_b T^2 \frac{\partial lnZ}{\partial T} = \sum_{i=1}^{M} E_i P_i(\beta) = \frac{1}{Z} \sum_{i=1}^{M} E_i e^{-\beta E_i} \tag{5}$$

The heat capacity is how much the energy change due to the change in the temperature. The heat capacity C_V can be defined as

$$C_V = \frac{\partial E}{\partial T} \tag{6}$$

$$\frac{\partial}{\partial T}\frac{1}{Z} = \frac{\partial}{\partial T}\frac{1}{\sum_{i=1}^{M} e^{-\frac{1}{k_B T}E_i}} = -\frac{1}{k_B T^2}\frac{\sum_{i=1}^{M} E_i e^{-E_i \frac{1}{k_B T}}}{\left(\sum_{i=1}^{M} e^{-E_i \frac{1}{k_B T}}\right)^2} = -\frac{1}{k_B T^2}\frac{\sum_{i=1}^{M} E_i e^{-E_i \frac{1}{k_B T}}}{(Z)^2} \tag{7}$$

$$\frac{\partial}{\partial T}\sum_{i=1}^{M} E_i e^{-\frac{1}{k_B T}E_i} = \frac{1}{k_B T^2}\sum_{i=1}^{M} E_i^2 e^{-\frac{1}{k_B T}E_i} \tag{8}$$

$$C_V = \frac{\partial < E >}{\partial T} = \frac{\partial}{\partial T}\left(\frac{1}{Z}\sum_{i=1}^{M} E_i e^{-\frac{1}{k_B T}E_i}\right) = -\frac{1}{k_B T^2}\frac{\sum_{i=1}^{M} E_i e^{-E_i \frac{1}{k_B T}}}{Z^2}\sum_{i=1}^{M} E_i e^{-\frac{1}{k_B T}E_i} + \frac{1}{Z}\frac{1}{k_B T^2}\sum_{i=1}^{M} E_i^2 e^{-\frac{1}{k_B T}E_i} \tag{9}$$

$$= -\frac{1}{k_B T^2}\left(\frac{\sum_{i=1}^{M} E_i e^{-E_i \frac{1}{k_B T}}}{Z}\right)^2 + \frac{1}{Z}\frac{1}{k_B T^2}\sum_{i=1}^{M} E_i^2 e^{-\frac{1}{k_B T}E_i} = \frac{1}{k_B T^2}\left(< E >^2 - < E^2 >\right) \tag{10}$$

The magnetic susceptibility is a measurable quantity, which indicates if the material is attracted or repelled of a magnetic field. Magnetic materials can be classified as paramagnetic, diamagnetic or ferromagnetic based on their susceptibility.

$$\chi = \frac{\partial < M >}{\partial H} \tag{11}$$

We can evaluate the mean magnetization through:

$$< M >= \sum_{i}^{M} M_i P_i(\beta) = \frac{1}{Z}\sum_{i}^{M} M_i e^{-\frac{E_i}{k_B T}} \tag{12}$$

The total energy of the system in addition of external magnetic field H can be described with:

$$E = -\sum_{i,j} J_{s_i s_j} - H\sum_{i} s_i \tag{13}$$

The magnetization is the sum of all spin for a given configuration:

$$\frac{\partial E_i}{\partial H} = -\sum_{i} s_i = -M_i \tag{14}$$

$$\frac{\partial}{\partial H}\frac{1}{Z} = \frac{\partial}{\partial H}\frac{1}{\sum_{i=1}^{M} e^{-\frac{1}{k_B T}E_i}} = -\frac{1}{k_B T}\frac{\sum_{i=1}^{M} M_i e^{-E_i \frac{1}{k_B T}}}{\left(\sum_{i=1}^{M} e^{-E_i \frac{1}{k_B T}}\right)^2} = -\frac{1}{k_B T}\frac{\sum_{i=1}^{M} M_i e^{-E_i \frac{1}{k_B T}}}{(Z)^2} \tag{15}$$

$$\frac{\partial}{\partial H}\sum_{i=1}^{M} M_i e^{-\frac{1}{k_B T}E_i} = \frac{1}{k_B T}\sum_{i=1}^{M} M_i^2 e^{-\frac{1}{k_B T}E_i} \tag{16}$$

$$\chi = \frac{\partial < M >}{\partial H} = \frac{\partial}{\partial H}\left(\frac{1}{Z}\sum_{i=1}^{M} M_i e^{-\frac{1}{k_B T}E_i}\right) = -\frac{1}{k_B T}\frac{\sum_{i=1}^{M} M_i e^{-E_i \frac{1}{k_B T}}}{Z^2}\sum_{i=1}^{M} M_i e^{-\frac{1}{k_B T}E_i} + \frac{1}{Z}\frac{1}{k_B T}\sum_{i=1}^{M} M_i^2 e^{-\frac{1}{k_B T}E_i} \tag{17}$$

$$= -\frac{1}{k_B T}\left(\frac{\sum_{i=1}^{M} M_i e^{-E_i \frac{1}{k_B T}}}{Z}\right)^2 + \frac{1}{Z}\frac{1}{k_B T}\sum_{i=1}^{M} M_i^2 e^{-\frac{1}{k_B T}E_i} = \frac{1}{k_B T}\left(< M^2 > - < M >^2\right) \tag{18}$$

1.1.3 Ferromagnetic order

A ferromagnet has a spontaneous magnetic moment even with the absence of an external magnetic field. Due the existence of a spontaneous moment the electron spin and magnetic moments must be arranged in a regular manner. In a ferromagnet most of the spins are aligned and in an anti ferromagnet the most of neighbouring spins are pointing in opposite directions. (compare "Introduction to Solid States Physics- 8th edition, by Charles kittel, page 323) In general -in a ferromagnet- spins up want to be next to spin up and spin downs want to be next to spin downs. At low temperature, the spin system is magnetised (either mostly up or mostly down). However, at high temperatures above the critical temperature, up and down spins are equally likely and the spins will in total cancel each other out. (total magnetisations is zero)

1.1.4 link from the Macroscopic values to statistical physics

In chapter 1.1.2, we derived the expression of the heat capacity and of the magnetic susceptibility of a canonical ensemble. We didn't care about statistical properties. However, what strikes the eye is that in the expression of the heat capacity as well as in the expression of the magnetic susceptibility, we see that they both depend on the variance of the energy of the magnetisation. this means that they can be written as:

$$C_v = \frac{1}{k_B T^2} \cdot \mathrm{Var(E)} \tag{19}$$

$$\chi = \frac{1}{k_B T} \cdot \mathrm{Var(M)} \tag{20}$$

The variance measures how far away a set of numbers is spread out. A huge variance means then that the values of the quantity fluctuate a lot around the expectation value. Now, we have linked a statistical property to a thermodynamic and macroscopic such as heat capacity.

1.2 theoretical numerical solutions

1.2.1 Ising model

Ising model is a mathematical model for ferromagnetism studies of phase transitions for magnetic system at given a temperature. The model consists the interaction between two neighbouring spins is related by the interaction energy

$$-Js_k s_l \tag{21}$$

where the sin s can be in two states +1 or -1,where s_k and s_l are the nearest neighbors. Which give a low energy (-J) if the two spin aligned and high energy (j) for spin pointing in opposite direction. The total energy to a system with N number of spins and with the absence of magnetic field can be expressed as

$$E = -J \sum_{<kl>}^{N} s_k s_l \tag{22}$$

1.2.2 Periodic boundary conditions

Periodic boundary conditions is used for approximating a large or infinite system by using smaller repeating system, we will impose PBCs on our spin lattice in x and y directions.

s(L+1,y) = s(1,y)
s(x,L+1) = s(X,1)

1.2.3 Metropolis algorithm in the two dimensional Ising model

The Ising model with Metropolis algorithm generates a sequence of states with Monte Carlo path, where the transition between states depends on the transition probability between the next and current state. The probability distribution is given by the Boltzmann distribution which is the probability for finding the system in a state s.

$$Ps = \frac{e^{-\beta E_i}}{\sum_{i=1}^{M} e^{-\beta E_i}} \tag{23}$$

It is difficult to compute since we need the sum over all states. If we have a 10 x 10 spin lattice interacting in our Ising model, there are 2^{100} possible states. Computing the sum seems to be not that efficient, but luckily the Metropolis algorithm needs only the ratios between the state probabilities and we do not need to compute the sum of all the states after all. The Metropolis algorithm in this case can be implement by establishing two dimensional Ising model with random lattice configuration. Then we flip a randomly chosen spin and compute the energy difference ΔE. If

$\Delta E \leq 0$ we accept the flip, otherwise we compute the transition probability $w = e^{-}\beta\Delta E$ and compare with a random number r.
If $r \leq w$ we accept the flip otherwise we keep the old configuration. We can keep choosing new random spins until we are satisfied with a good representation of the states.(compare to"lecture notes Computational physics 2015 at University of Oslo by Morten Hjorth-Jensen page 435)"

1.2.4 Critical temperature (Lars Onsager)

in 1944 the Norwegian chemist Lars Onsager made very important discovery in theoretical physics, namely the exact solution of the Ising spin model in two dimensions. His work is up to now a valid theoretical description of the two dimensional Ising model. Onsager's solutions achieved the thermodynamic properties of interaction systems and phase transitions at T_c. However in 1942 Lars Onsager solved the two dimensional model for zero field energy, which has been published two years after. In 1948, he wrote the solution for the zero field magnetization in a conference at Cornell. Onsager showed how to derive the partition function for the canonical ensemble with zero external magnetic field Z(B = 0,T)with N spins.

$$Z_N = \left(2\cosh(\beta J)e^I\right)^N \tag{24}$$

where $I = \frac{1}{2\pi}\int_0^\pi d\phi ln\left[\frac{1}{2}\left(1+\sqrt{1-\kappa^2\sin^2\phi}\right)\right]$ where $\kappa = \frac{2\sinh(2\beta J)}{\cosh^2(2\beta J)}$ and the energy is given by

$$< E >= -J\coth(2\beta J)\left[1+\frac{2}{\pi}(2\tanh^2(2\beta J)-1)K_1(q)\right] \tag{25}$$

where $q = \sinh(2\beta J)/\cosh^2(2\beta J)$ and the complete elliptic integral of the first kind is:

$$k_1(q) = \int_0^{\frac{\pi}{2}} \frac{d\phi}{\sqrt{1-q^2\sin^2\phi}} \tag{26}$$

and differentiating the energy with the respect to temperature we obtain the specific heat:

$$C_v = \frac{\partial < E >}{\partial T} = \frac{4K_B}{\pi}(\beta J\coth(2\beta J))^2\left\{K_1(q)-K_2(q)-(1-\tanh^2(2\beta J))\left[\frac{\pi}{2}+(2\tanh^2(2\beta J)-1)K_1(q)\right]\right\} \tag{27}$$

where

$$k_2(q) = \int_0^{\frac{\pi}{2}} d\phi\sqrt{1-q^2\sin^2\phi} \tag{28}$$

Near the critical temperature T_c the specific heat behaves as:

$$C_v \approx -\frac{2}{\pi}\left(\frac{2J}{K_B T_c}\right)^2 ln\left[1-\frac{T}{T_c}\right] + const. \tag{29}$$

$$C_v \sim \left|1-\frac{T}{T_c}\right|^\alpha \tag{30}$$

the limiting form of the function

$$\lim_{\alpha\to 0}\frac{1}{\alpha}(\mathrm{Y}^{-\alpha}-1) = -ln\mathrm{Y} \tag{31}$$

can be used to infer that closed-form result in low singularity with $\alpha = 0$. We do not want t make a complete derivation of Onsager's result, however we want to limit ourselves to his final result for the expectation value of the magnetisation:

$$< \frac{M(T)}{N} >= \left[1-\frac{(1-\tanh^2(\beta J))^4}{16\tanh^4(\beta J)}\right]^{\frac{1}{8}} \tag{32}$$

for $T < T_c$. otherwise the magnetization is zero "lecture notes Computational physics 2015 at University of Oslo by Morten Hjorth-Jensen page 435)"

From Onsager's result, we get <M(t)/N>. T_c is the temperature, where we start getting a non zero magnetization. When we heat up the system, we start with non-zero magnetisation until we pas the critical Temperature. From there

on, we have zero magnetisation. If we want now to calculate the critical Temperature, we can set the equation 32 to zero. The temperature belonging to this is the critical temperature:

$$< M(T)/N >= \left[1 - \frac{(1-\tanh^2(\beta J))^4}{16\tanh^4(\beta J)}\right]^{\frac{1}{8}} = 0 \tag{33}$$

the only way to obtain <M>=0 is when equation 34 is valid. We assume that the magnetisation over the temperature is continuous.

$$\frac{(1-\tanh^2(\beta J))^4}{16\tanh^4(\beta J)} = 1 \tag{34}$$

This gives us:

$$0 = \tanh(\beta J)^2 + 2\cdot\tanh(\beta J) - 1 \tag{35}$$

$$\tanh(J\beta) = \frac{-2 \pm \sqrt{4+4}}{2} = -1 \pm \sqrt{2} \tag{36}$$

$$\beta J = \operatorname{arctanh}(-1 \pm \sqrt{2}) \tag{37}$$

$$= \frac{1}{2} ln(1+\sqrt{2}) \tag{38}$$

$$\Rightarrow \frac{k_B T_c}{J} = \frac{2}{ln(1+\sqrt{2})} \approx 2.2692 \tag{39}$$

Now, we found the critical temperature of 2.2692 (dimensionless unit) Near the critical temperature, we can describe the behaviour of the different quantities through the so called "power law". We get the so called power law through a taylor-expansion at the critical temperature. However, we want to spend no time on deriving the critical exponents of the power laws. What is interesting is that with this power lows, a few systems with the same properties behave close to the critical temperature the same. No matter, which lattice-size we choose, we get the same behaviour close to the critical temperature.

$$< M(T) > \propto (T - T_c)^{\frac{1}{8}} \tag{40}$$

$$C_v \propto |T_c - T|^{\alpha} \tag{41}$$

$$\chi(t) \propto |T_c - T|^{\gamma} \tag{42}$$

$$\tag{43}$$

with $\alpha = 0, \gamma = \frac{7}{4}$. In addition to that, we can describe the correlation length of the system by $\xi(T)$:

$$\xi(T) \propto |T_c - T|^{-\nu} \tag{44}$$

All in all, the critical temperature scales than as follows:

$$T_c(L) - T_c(L=\infty) = a \cdot L^{\frac{-1}{\nu}} \tag{45}$$

In this case, a is a constant and ν is defined through the previous equation 44.

1.3 Closed solution for a 2 dimensional 2 x 2 lattice

We want now to look at a 2 x 2 lattice and we want to calculate the partition function, the energy, magnetisation, heat capacity and susceptibility of the system dependent of T. The partition function for a canonical ensemble with periodic boundary conditions can be computed by:

$$Z = \sum_{i=1}^{M} e^{-\beta E_i} \tag{46}$$

Here, β is $\frac{1}{k_b \cdot T}$, where k_b is the Bolzmann constant. In this expression we sum over all microstates m. The Energy of the system in configuration i is then:

$$E_i = -J \sum_{<kl>}^{N} s_k s_l \tag{47}$$

The sum over $< kl >$ means that we only sum over nearest neighbours. In our 2 x 2 case, we have for each "particle" two possible values ± 1. This means that we have all in all $2^{2\cdot 2} = 2^4 = 16$ micro states. We have to compute the Energy

of the micro states in order to compute the partition function. We also want to introduce the magnetisation, which is simply the sum over all the spins of the system:

$$M_i = \sum_{j=1}^{N} s_j \tag{48}$$

We want also to introduce the so called degeneracy, which counts the number of micro states for a given micro energy. We get the following table: We can now write the expression of the partition function as in equation 51. We used the

Figure 1: Energy of the different micro states

Number of spins up (+1)	Degeneracy	Energy	Magnetization
4	1	$-8J$	4
3	4	0	2
2	4	0	0
2	2	$8J$	0
1	4	0	-2
0	1	$-8J$	-4

Table 1 to calculate the sum over the micro states.

$$Z = \sum_{i=1}^{M} e^{-\beta E_i} = 12 \cdot e^{-\beta \cdot 0} + 2 \cdot e^{-8J\beta} + 1 \cdot e^{8J\beta} + 1 \cdot e^{8J\beta} \tag{49}$$

$$= 12 + 2 \cdot e^{-8J\beta} + 2 \cdot e^{8J\beta} \tag{50}$$

$$= 12 + 4 \cdot \cosh\left(8J\beta\right) \tag{51}$$

We can now calculate the expectation value of the energy. There are two possible ways of calculating it. the first way of calculating the expectation value of the energy can be seen in equation 53.

$$< E > = -\frac{\partial ln(Z)}{\partial \beta} = -\frac{1}{Z} \cdot 32J \cdot \sinh(8J\beta) \tag{52}$$

$$= -\frac{32J \cdot \sinh((8J\beta)}{Z} \tag{53}$$

$$= -\frac{8 \cdot J \cdot \sinh(8J\beta)}{3 + \cosh(8J\beta)} \tag{54}$$

Alternatively, we can calculate the expectation value of the Energy by looking at the micro states:

$$< E >= \frac{1}{Z} \sum_{i=1}^{M} E_i e^{-\beta E_i} = -\frac{8 \cdot J \cdot \sinh(8J\beta)}{3 + \cosh(8J\beta)} \tag{55}$$

Both expressions are equal. Next, we want to determine the expectation value of the magnetisation. We use the formula 57. We can see that we get 0 for the expectation value of the magnetisation.

$$< M > = \frac{1}{Z} \sum_{i}^{M} M_i \cdot e^{-\beta E_i} \tag{56}$$

$$= \frac{1}{Z} \cdot \left(4 \cdot 1 \cdot e^{-8J\beta} + 2 \cdot 4 + (-2) \cdot 4 + (-4) \cdot 1 \cdot e^{-8J\beta}\right) \tag{57}$$

$$= 0 \tag{58}$$

However, we are interested in the expectation value of the absolute value of magnetisation, which is $< |M| >$. This expression can be determined as follows:

$$< |M| > = \frac{1}{Z} \sum_{i}^{M} |M_i| \cdot e^{-\beta E_i} \tag{59}$$

$$= \frac{1}{Z} \cdot \left(|4| \cdot 1 \cdot e^{8J\beta} + |2| \cdot 4 + |(-2)| \cdot 4 + |(-4)| \cdot 1 \cdot e^{8J\beta}\right) \tag{60}$$

$$= \frac{1}{Z} \cdot \left(8 \cdot e^{8J\beta} + 16\right) \tag{61}$$

$$= \frac{2 \cdot e^{8J\beta} + 4}{3 + \cosh(8J\beta)} \tag{62}$$

In order to describe how the temperature will change when thermal energy is added to the system, we want to look at a quantity called heap capacity. (C_v) The bigger this quantity is the less heats the system up by a given amount of thermal energy, which is added to the system.

$$C_v = \frac{1}{k_b T^2}\left(\frac{1}{Z}\sum_{i=1}^{M} E_i^2 e^{-\beta E_i} - \left(\frac{1}{Z}\sum_{i=1}^{M} E_i e^{-\beta E_i}\right)^2\right) \tag{63}$$

$$= \frac{1}{k_b T^2}\left(\frac{1}{Z}\left(2\cdot(8J)^2\cdot e^{8J\beta} + 2\cdot(-8J)^2\cdot e^{-8J\beta}\right) - \left(\frac{8\cdot J\cdot\sinh(8J\beta)}{3+\cosh(8J\beta)}\right)^2\right) \tag{64}$$

$$= \frac{1}{k_b T^2}\left(\frac{64\cdot J\cdot\cosh(8J\beta)}{3+\cosh(8J\beta)} - \left(\frac{8\cdot J\cdot\sinh(8J\beta)}{3+\cosh(8J\beta)}\right)^2\right) \tag{65}$$

$$= \frac{1}{k_b T^2}\left(\frac{64\cdot J + 3\cdot J\cdot 64\cosh(8J\beta)}{\left(3+\cosh(8J\beta)\right)^2}\right) \tag{66}$$

$$= \frac{64}{k_b T^2}\left(\frac{J + 3J\cdot\cosh(8J\beta)}{\left(3+\cosh(8J\beta)\right)^2}\right) \tag{67}$$

$$\tag{68}$$

At last, we want to have a look at the magnetic susceptibility. This quantity is a magnetic property of the material. The magnetic susceptibility describes the response of the material to an applied magnetic field.

$$\chi_{abs} = \frac{1}{k_b T}\cdot\left(\frac{1}{Z}\sum_{i=1}^{M} M_i^2 e^{-\beta E_i} - \left(\frac{1}{Z}\sum_{i=1}^{M} |M_i| e^{-\beta E_i}\right)^2\right) \tag{69}$$

$$= \frac{1}{k_b T}\cdot\left(\frac{1}{Z}\cdot\left(4^2\cdot 1\cdot e^{8J\beta} + 2^2\cdot 4 + (-2)^2\cdot 4 + (-4)^2\cdot 1\cdot e^{8J\beta}\right) - \left(\frac{2\cdot e^{8J\beta}+4}{3+\cosh(8J\beta)}\right)^2\right) \tag{70}$$

$$= \frac{1}{k_b T}\cdot\left(\frac{32e^{8J\beta}+32}{12+4\cdot\cosh\left(8J\beta\right)} - \left(\frac{2\cdot e^{8J\beta}+4}{3+\cosh(8J\beta)}\right)^2\right) \tag{71}$$

$$= \frac{1}{k_b T}\cdot\left(\frac{8e^{8J\beta}+8}{3+\cdot\cosh\left(8J\beta\right)} - \left(\frac{2\cdot e^{8J\beta}+4}{3+\cosh(8J\beta)}\right)^2\right) \tag{72}$$

$$\tag{73}$$

2 Execution

2.1 Implementing the Algorithm

In all the calculations we did with the programm, we assume that $J = 1$ and $k_B = 1$, meaning that $\beta = \frac{1}{T}$. Therefore, all values of the temperature T in this part will be in dimensionless units! Our program is split into three parts: in the first part, we prepare the calculations by declaring variables and initializing the grid. We call a function *readINput* that reads input from the screen the desired temperature range, the temperature step size, the lattice size and preferences concerning the initial spin setup. Next, we call a function *initialize* that initializes the array that contains all the spins. Depending on the user's choice, the lattice will be set up randomly or with all spins pointing in one direction. We also prepare the output file and the seed for the random number generator in the first part.

In the main part, we perform a loop over all desired temperatures. For every temperature, we set up an array which saves the probabilities for every possible energy change that can occur when a spin is flipped. As these are only 5 different values, it is much easier and better in matters of computation time to compute and save them before we start with the Metropolis algorithm instead of calculating them every time a spin is flipped. Next, the program can perform a function to start the process of thermalization. We will discuss this function later in part 2.3. The call of this function should be commented out for studying the thermalization in more detail.

In the following loop, the program performs the main Metropolis algorithm: the loop runs over the number of *MCcycles* that was set by the user. It will first call the function *monteCarlo* every time and thereafter update the array containing the expectation values of $< E >$, $< E^2 >$, $< M >$, $< M^2 >$ and $< |M| >$. Please note that we included three output statements for each cycle that are normally commented out. These commands are only used only for some parts of the exercises in this project where we want to study the development of the system over the time!

```
for (temp=tempStart; temp<=tempMax; temp+=tempStep){
        acceptedmoves=0;
//reset Energy and magnetization (averages)
```

```
for(int i=0; i<5; i++){
        averages[i]=0;
}
//Set array with possible energy changes according to temperature
for(int i=0; i<5; i++){
        double delEnergy = (4*i)-8;
        energyChanges[i] = exp(((double)-delEnergy)/((double)temp));
}
//thermalization - comment out for exercices
//where thermalization behaviour should be studied!
thermalization(spinArray, size, idum, energyChanges, M, E, acceptedmoves);
//actual Monte Carlo happens here
for(int i=0; i<mccycles; i++){
        monteCarlo(spinArray, size, idum, energyChanges, M, E, acceptedmoves);
        averages[0]+=E;
        averages[1]+=E*E;
        averages[2]+=M;
        averages[3]+=M*M;
        averages[4]+=abs((int)M);
//Only for c), otherwise comment next line out (very slow!)
//output(size, i+1, temp, averages);
//only for c) (second part); comment out if not used!
//ofile << i << "\t" << acceptedmoves << endl;
//This is for part d, can be commented out else
//ofile << E << endl;
}
//output of data for this temperature (comment out if not used!)
output(size, mccycles, temp, averages);}
```

The main function that is called in this loop is *monteCarlo*. Every time we call this function, we perform another loop over the number of spins in the system. In each loop, we pick one spin randomly and try to flip it. This means that we calculate the change in energy that is caused by flipping this spin and compare the corresponding probability from the array that was set up before to a random number between 0 and 1. By doing this, we ensure that every move that lowers the overall energy is accepted, but also some of the moves that will result in a higher energy.

Instead of performing the loop over the number of spins and picking one spin every time, we could also try to flip all spins at the same time. However, this would mean that we had to calculate very complicated probabilities for the change in energies, resulting in a higher computation time. By using the loop described above, only one spin is affected at one time, meaning that only five different values for the change in overall energy can occur. This means that we can just pick the previously calculated probabilities for energy changes and still flip the same amount of spins as when trying to flip all of them at the same time.

```
//This is the actual Monte Carlo method as described in the report!
void monteCarlo(int** spinArray, int size, long& idum,
        double* energyDeltas, double& M, double& E, int& accept){
int count;
for(count=0; count<=(size*size); count++){
        //Pick random position
        int x, y;
        x=(int)(ran3(&idum)*size);
        y=(int)(ran3(&idum)*size);
        //check energy difference
        double deltaE =(double) 2*spinArray[x][y]*(spinArray[(size+x+1)%size][y]
                +spinArray[(size+x-1)%size][y]+spinArray[x][(size+y+1)%size]
                +spinArray[x][(size+y-1)%size]);
        //compare it to random number
        if(ran3(&idum)<=energyDeltas[(int)(deltaE+8)/4]){
        //change spin
        spinArray[x][y]*=-1;
        //update energy and magnetization
        M+=2*spinArray[x][y];
        E+=deltaE;
        //count accepted move! (needed for part c)
```

```
            accept++;
}}
return;}
```

In the default mode, our program will write the expectation values of $< E >$, $< |M| >$, $< c_v >$ and $< \chi >$ as well as the number of Monte Carlo cycles and the temperature to an output file. However, by using the output statements mentioned above, we can also write the current expectation values, the current energy or the number of accepted moves to this file if this needed. To avoid conflicts in the output file, only one output statement should be used at the same time, whilst the others should be commented out!

In the end, our program frees allocated memory and closes the output file before finishing.

2.2 Parallelization of the code

```
//C++/Omp section
//ising modle with omp
S1 = tempStart;
E1 = S2 = tempStart+(tempMax- tempStart)/4;
E2 = S3 = tempStart+(tempMax- tempStart)/2;
E3 = S4 =  tempStart+ 3*(tempMax- tempStart)/4;
E4 = tempMax;

// paralizing the 4 threads each thread run specfic temprature part.
#pragma omp parallel sections shared (tempStep, size, isRandomSetup,mccycles)
{
    #pragma omp section
    {
     JohansenCode(S1, E1, tempStep, size, mccycles, isRandomSetup);
    }
    #pragma omp section
    {
     JohansenCode(S2, E2, tempStep, size, mccycles, isRandomSetup);
    }
    #pragma omp section
    {
     JohansenCode(S3, E3, tempStep, size, mccycles, isRandomSetup);
    }
    #pragma omp section
    {
     JohansenCode(S4, E4, tempStep, size, mccycles, isRandomSetup);
    }
}
```

We used omp to parallelize our code by arranging 4 equal sub sections of the temperature region, each subsection is calculated by one thread at the time. The tested computer has 4 threads and gave a speedup factor by 3.

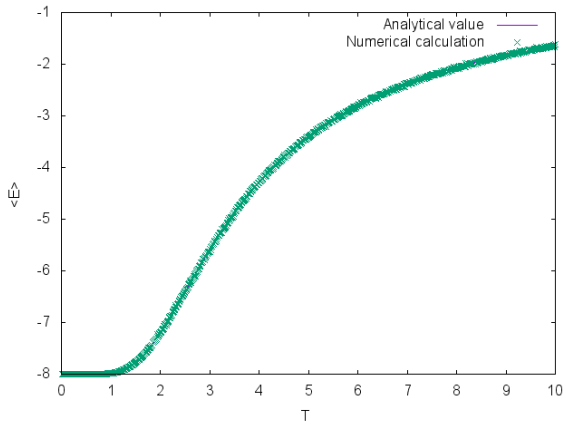
2.3 Results

As a first benchmark test for our program, we calculated the expectation values $< E >$, $< |M| >$, $< C_V >$ and $< \chi >$ of a 2×2-lattice for different temperatures. Those results could be easily compared to the analytical values from the part 1.3. In fig. 2a – 2d, you can see our results for these expectation values compared with the analytical solutions as functions of T. We took 10000 Monte Carlo cycles for each temperature to achieve good results. You can see that the results fit very well to the analytical solutions which means that our program passed this benchmark test and works fine.

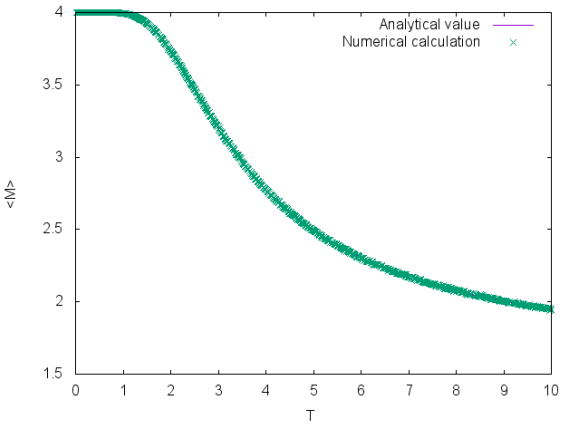
In the following table, we compared the analytical results for the different expectation values for a temperature $T = 1.0$. It shows that all numerical results have a precision of at least two, in most cases of even three leading digits.

In the next step, we took a closer look at the effect of thermalization. This term describes the process of the system slowly reaching the most likely state for a given temperature. When we start with a random setup, it is very unlikely that the system is already in this state at the beginning, but it will need some time (or, in our case, some Monte Carlo cycles) to reach it.

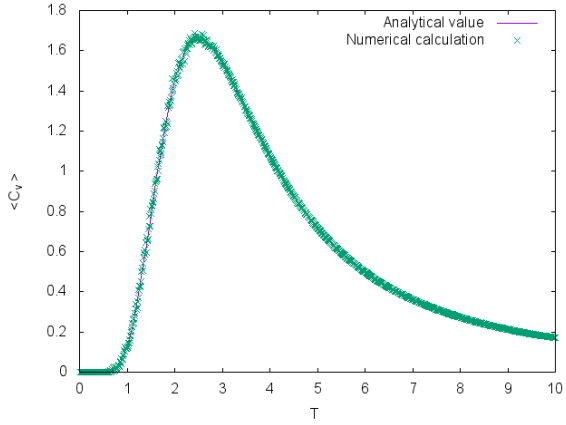
To get more insight in the process of thermalization, we observed the development of $< E >$ and $< |M| >$ of a 20×20-lattice for temperatures of $T = 1.0$ and $T = 2.4$, for both starting with a random setup and a lattice with all



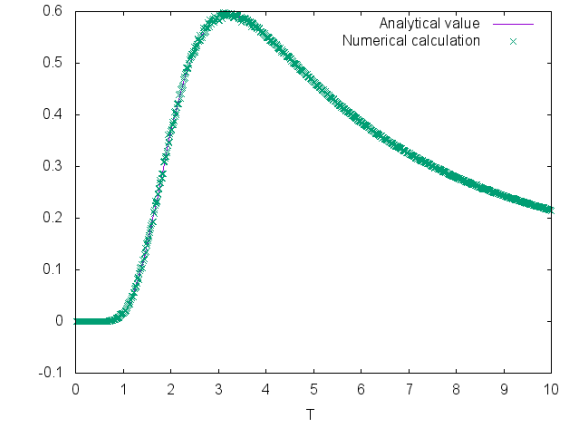
(a) $< E >$ in a 2×2-lattice for different temperatures



(b) $< |M| >$ in a 2×2-lattice for different temperatures



(c) $< c_v >$ in a 2×2-lattice for different temperatures



(d) $< \chi >$ in a 2×2-lattice for different temperatures

Figure 2: different figures for 2x2 lattice

Table 1: Analytical and numerical value of different expectaion values for $T = 1.0$ in a 2×2-lattice

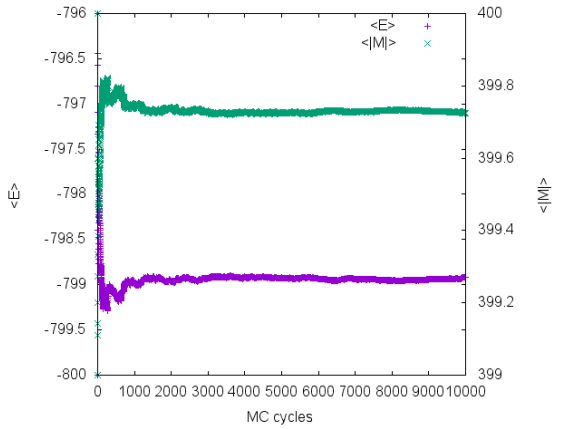
	Analytical value	Numerical results
$< E >$	-7.983928	-7.984080
$< \|M\| >$	3.994642	3.994760
$< c_v >$	0.128329	0.127107
$< \chi >$	0.016043	0.015494

spins pointing in one direction. You can see that development in the figures 3a – 3d, where these expectation values are plotted as function of the number of Monte Carlo cycles. We also plotted how many ‚moves'(flipping of spins) got accepted as a function of the number of cycles in fig. 4a – 4b It is obvious that this value is proportional to the number of cycles and that, the higher the temperature is, the faster the number of accepted moves increases.

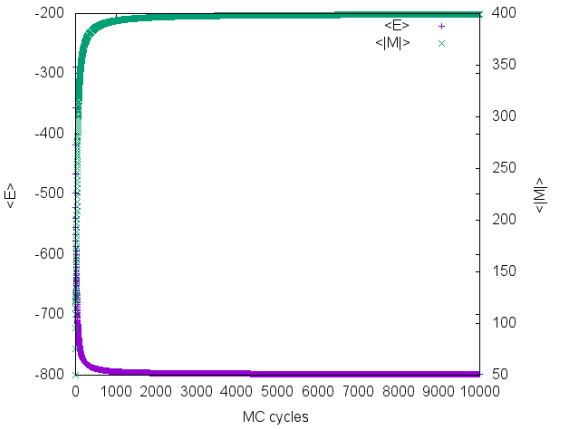
All these plots show that we need a bit more than 1000 cycles in order to reach a stable state. This duration seems not to depend on the temperature, but obviously on the initial set-up: If this is close to a setup corresponding to the given temperature, it will not take long time to reach a stable state.

For later calculation, we set up a function called *thermalization* for our program. This function performs the Metropolis algorithm without collecting data, instead it evaluates the energy after every 10 cycles. When the change between two measurements is less than 1%, the program starts collecting data. This takes in account the process described above and ensures that we reached a stable state before the actual calculations start.

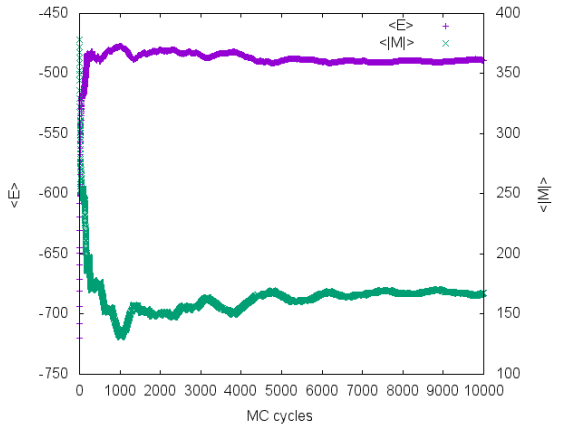
In the next step, we took a closer look at the probability of the single values of E to appear. We observed a 2×2-lattice at a temperature of $T = 1.0$ respectively $T = 2.4$ again and did two histograms of how often an energy value occurred. Obviously, that distribution is expanded for a higher temperature. This correspondents with the fact that the variation of the energy for these temperatures is higher, which means that more values are accessed. Both histograms can be found in fig 4c – 4d.



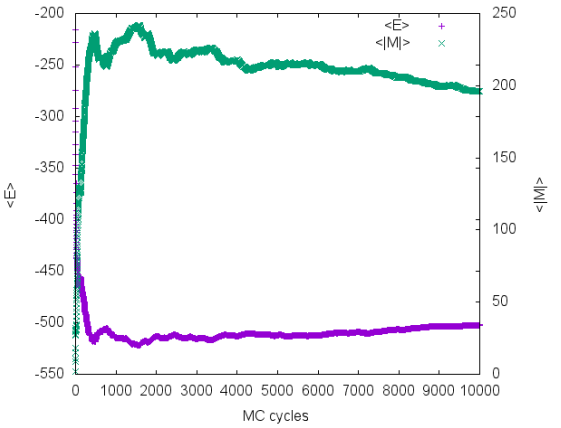
(a) $< E >$ and $< |M| >$ in a 20×20-lattice as a function of the number of Monte Carlo cycles, starting with ordered spins at $T = 1.0$



(b) $< E >$ and $< |M| >$ in a 20×20-lattice as a function of the number of Monte Carlo cycles, starting with random spins at $T = 1.0$

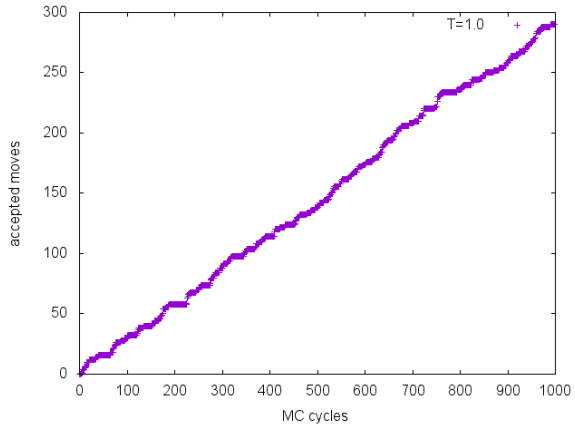


(c) $< E >$ and $< |M| >$ in a 20×20-lattice as a function of the number of Monte Carlo cycles, starting with ordered spins at $T = 2.4$

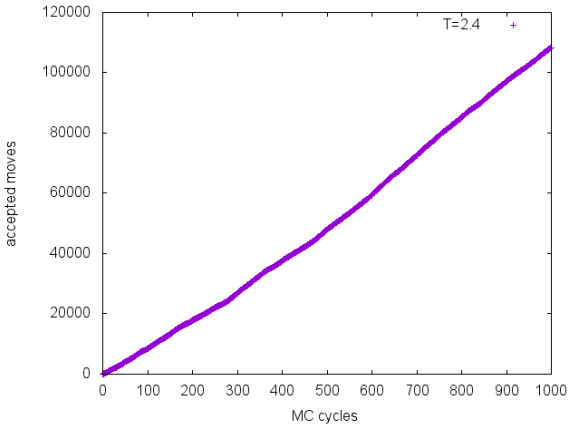


(d) $< E >$ and $< |M| >$ in a 20×20-lattice as a function of the number of Monte Carlo cycles, starting with random spins at $T = 2.4$

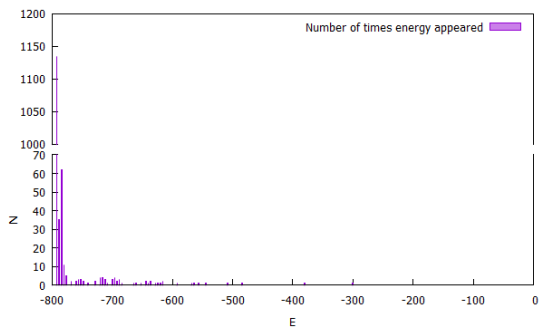
Figure 3: figures for a lattice-size of 20



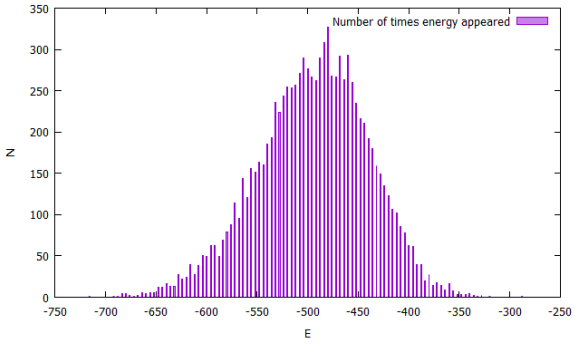
(a) The number of accepted spin flips as function of the number of cycles at $T = 1.0$



(b) The number of accepted spin flips as function of the number of cycles at $T = 2.4$



(c) Frequency of occurrence for different energies at $T = 1.0$



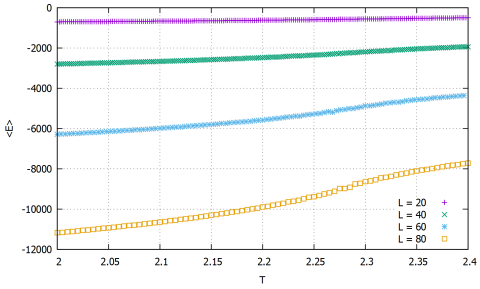
(d) Frequency of occurrence for different energies at $T = 2.4$

Figure 4: general plots

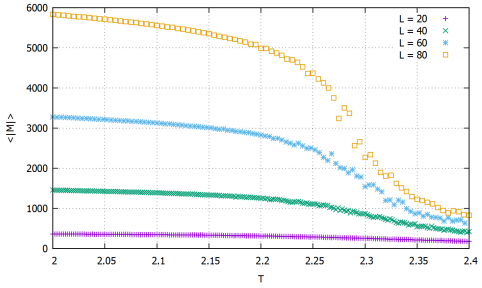
In figure 5a, 5b, 5c and 5d, the expectation values $< E >$ and $< |M| >$, the susceptibility χ and the specific heat C_V for lattice sizes $L = 20, 40, 60, 80$ can be seen as functions of T close to the critical temperature. It is the easiest to estimate the critical temperature by looking at figure 5d, because it is at the temperature where C_V is the highest. In figure 6 and 7 we increased the number of Monte Carlo cycles and decreased the temperature step length and interval to determine the temperature of the maximum of C_V more precisely. In table 2 the the from figure 6 and 7 measured critical temperatures can be seen. While it is possible to measure T_C even more precisely by increasing the Monte Carlo cycles and decreasing the Temperature step length, it is not possible aim at any precision, due to the quickly increasing calculation time.

In figure 5c we can see that for low temperatures the measured values for the susceptibility are nearly zero. At T_C the susceptibilities peak up to at very high value and then decrease to go against zero again. In figure 5a it can be observed that at T_C the slope of the curve $< E(T) >$ starts to decrease again. As already mentioned in figure 5d we can clearly see a peak of the graph at T_C. In figure 5c and 5d it can also be seen, that the critical temperature decreases with bigger lattice sizes.

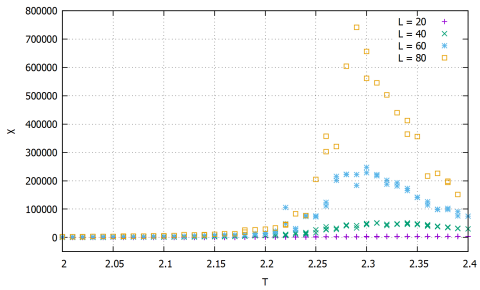
Figure 5b shows, that while $< |M| >$ stays nearly stable at a non-zero value and decreases very slow for temperatures much smaller than T_C, near the critical temperature it decreases quickly and goes against zero. This is a sign that a phase transition is taking place. For a temperature lower than T_C the lattice has a general magnetization. Hence it is ferromagnetic. For temperatures higher than T_C the lattice has nearly no magnetization. Because the susceptibility is positive then, the lattice is paramagnetic.



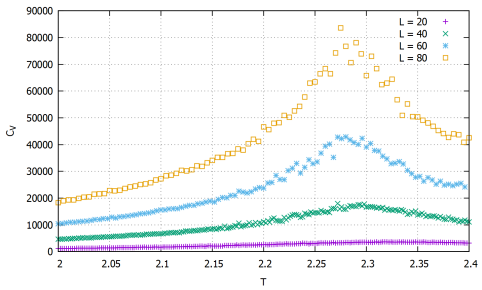
(a) Expectation value of the energy against temperature for different lattice sizes (10^6 Monte Carlo cycles)



(b) Expectation value of the absolute magnetisation against temperature for different lattice sizes (10^6 Monte Carlo cycles)



(c) Susceptibility against temperature for different lattice sizes (10^6 Monte Carlo cycles)



(d) Expectation value of the specific heat against temperature for different lattice sizes (10^6 Monte Carlo cycles)

Figure 5: values of different properties against temperature

L	T_C
20	(2.33 ± 0.01)
40	(2.295 ± 0.007)
60	(2.285 ± 0.005)
80	(2.280 ± 0.007)

Table 2: Critical temperature for different lattice sizes L

The connection between different lattice sizes and the critical temperature is given by the following formula

$$T_C(L) - T_C(L = \infty) = a \cdot L^{-\frac{1}{\nu}} \tag{74}$$

We have $\nu = 1$, so we get

$$T_C(L) = T_C(L = \infty) + a \cdot \frac{1}{L} \tag{75}$$

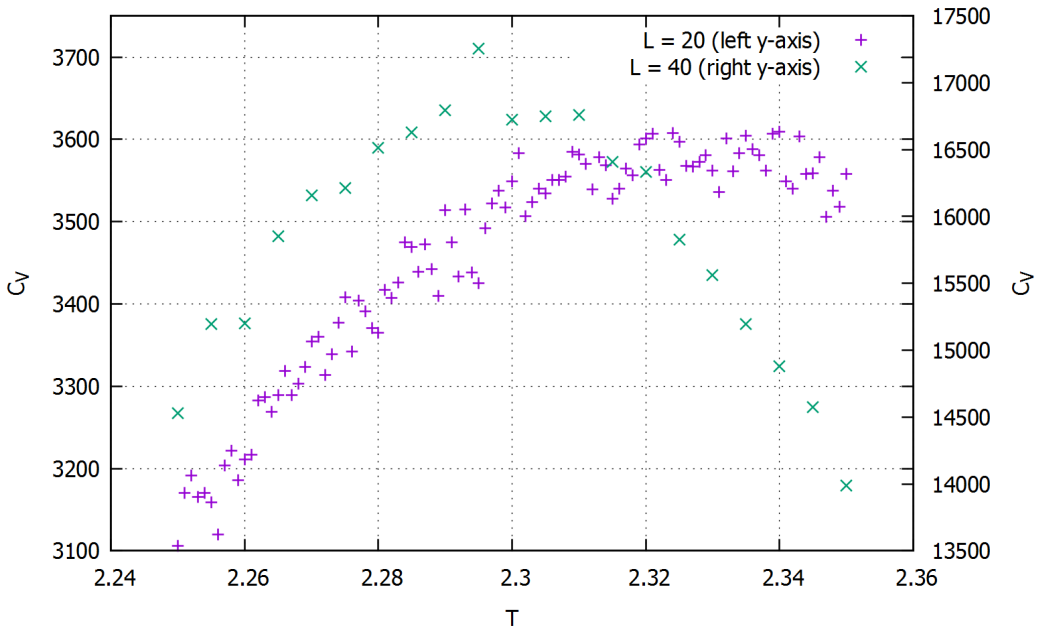


Figure 6: Expectation value of the specific heat against temperature for lattice sizes L = 20 and L = 40 ($5 \cdot 10^6$ Monte Carlo cycles)

If we now plot T_C against $\frac{1}{L}$ we should get a line. By measuring the value at which the line crosses the y-axis we can determine $T_C(L = \infty)$. At figure 8 we have plotted our measured values for the critical temperature (which can be seen in table 2) against $\frac{1}{L}$ and signed in a fit line and an error line. The fit line crosses the y-axis at $T_C = 2.263$ while the error line crosses it at $T_C = 2.273$. Thus our value for the critical temperature for an infinite sized lattice is $T_C(L = \infty) = (2.26 \pm 0.01)$. The analytical solution $T_C^a = 2.269$ is in the first error interval of this value. Hence our value and the analytical solution match.

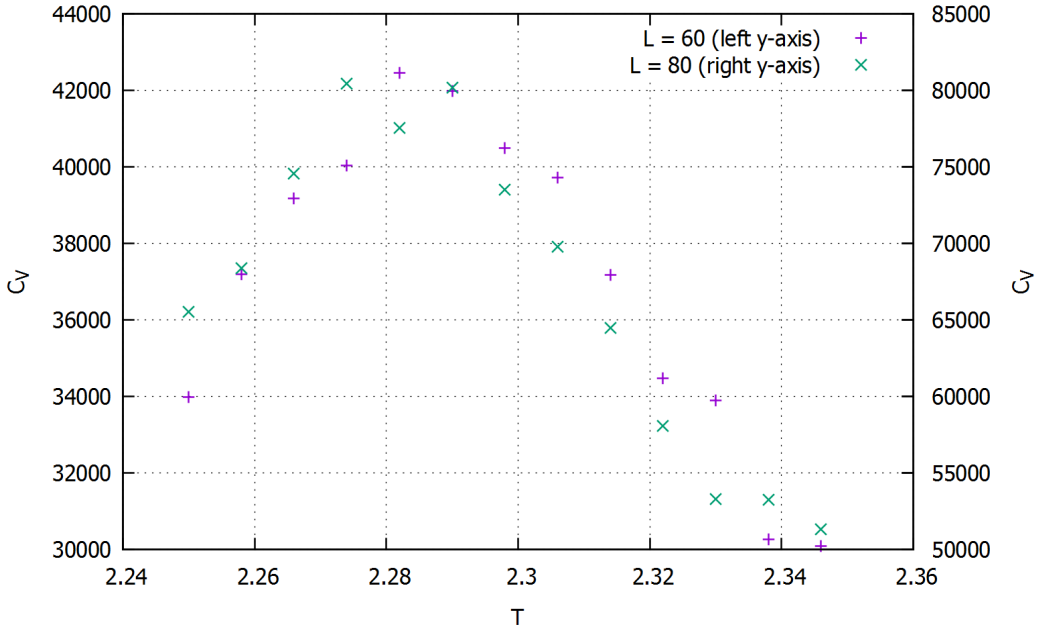


Figure 7: Expectation value of the specific heat against temperature for lattice sizes L = 60 and L = 80 ($5 \cdot 10^6$ Monte Carlo cycles)

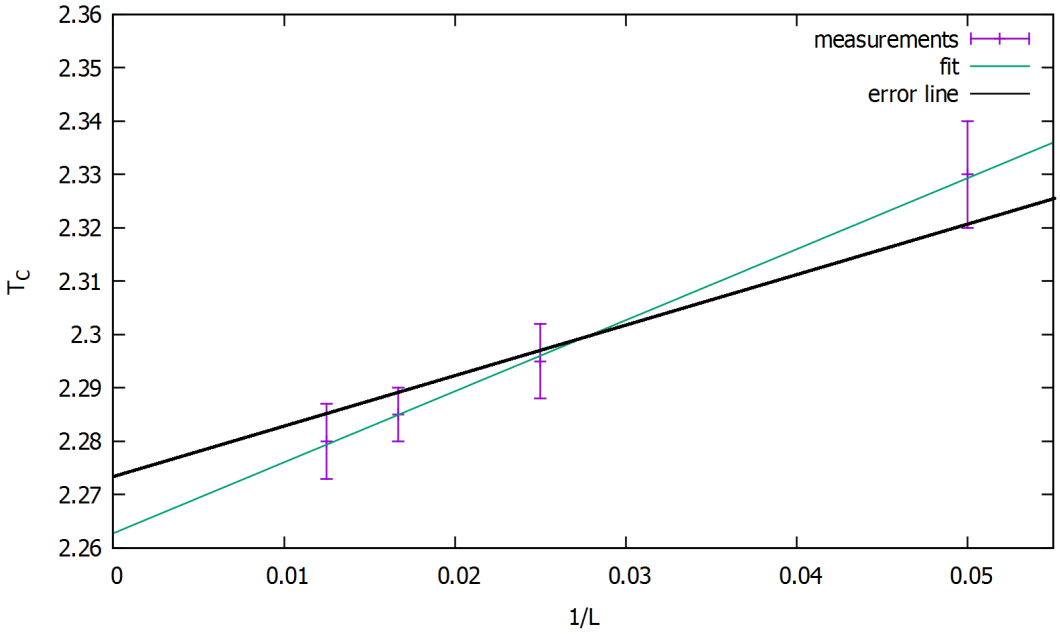
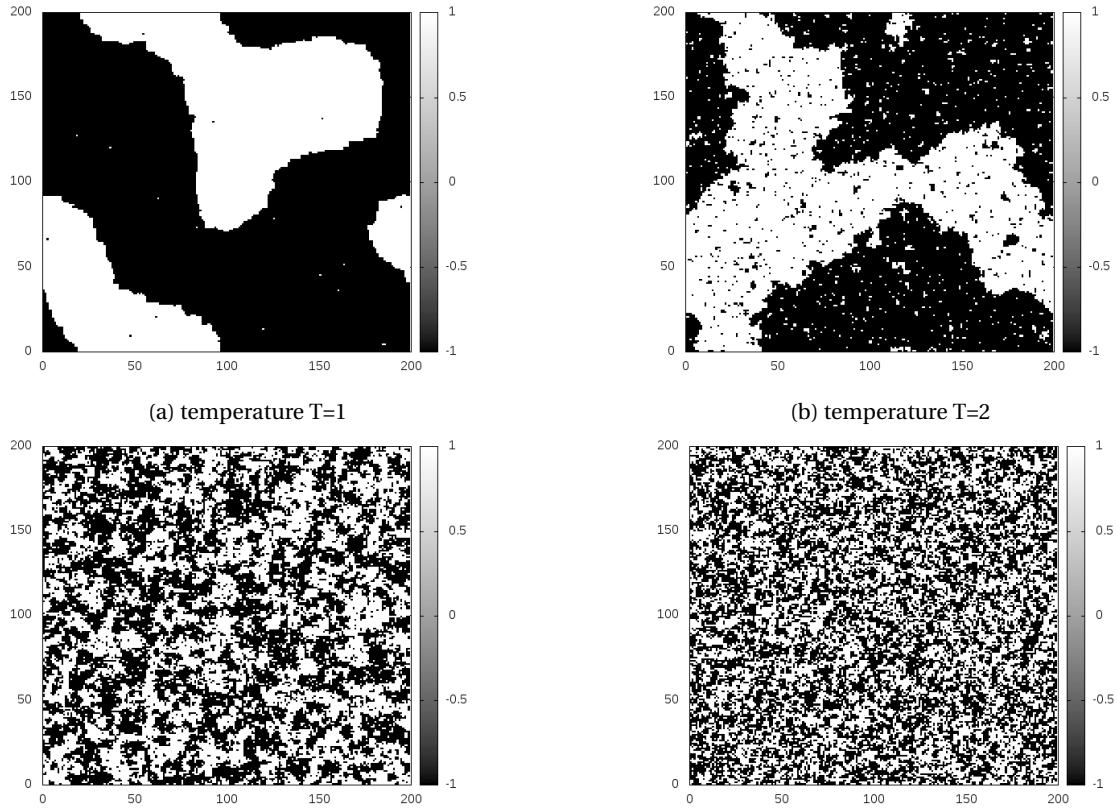


Figure 8: Critical temperature T_C against $\frac{1}{L}$

3 Discussion

All in all, our results for the two times two lattice matched with the theoretical predictions. We have observed that a phase transition is taking place at the critical temperature. There the lattice changes from being ferromagnetic to being paramagnetic. Our numerically calculated value for the critical temperature for an infinite sized lattice which is $T_C(L = \infty) = (2.26 \pm 0.01)$ matches with the analytical solution $T_C^a = 2.269$. By increasing the Monte Carlo cycles it would be possible to get a more accurate result, but the calculation time would increase. Thus a balance between the accuracy of the numerical result and the needed time for the computation of the numerical result needs to be found. We looked at our system and visualized a 200 x 200 spin-lattice for different temperatures after the same am. We visualized the spin-array through a 2D-plot, where the spin-value -1 indicated with black and the spin-value 1 with white pixels. For low temperatures like in figure 9a, we observe a ordered configuration of the spins. At a low temperature and after a infinite number of monte carlo cycles, all spins should be up or all spins should be down. This means that in figure 9a, you would get more or less all spins pointing in one direction after more monte carlo cycles. When we increase the temperature and we come close to the critical temperature, a increase of the so called correlation length kicks in. This means that the influence of particles becomes more and more important. At the critical temperature, "everybody sees everybody". Above the critical temperature, the correlation length drops down to zero. Thermal noise also gets more and more important, when we heat up the system. Comparing the figure 9a to figure 9b, we see more spontaneous flipping of the spins. If we now heat up the system, we observe a transition from ordered to disordered configuration. We can see that for $T = 2$, we get a more disordered spin-array. Above the critical temperature, we can see that the system in in a disordered state. (order decreases from 9a to figure 9d) In addition to that we notice, that for low temperatures, we get districts of only white and districts of only black parts. The transition from ordered to less ordered conformation of the system with increasing of the temperature can be identified with a increase of entropy of the system. To cluster groups with equal behaviour close to the critical temperature (as can be done like with the 2D Ising model; with fixed critical exponents) is quite useful in order to check whether the computed results are correct.(if you know to which group your system belongs, you know the critical exponent of the system). This can be described by the mentioned power law behaviour. We have seen in our project that a classical, fairly simple model gives the qualitatively right physical behaviour. We have also to keep in mind that in reality in our case, we have to deal with quantum systems and we also have to consider interactions between "not nearest neighbours". Nevertheless, such considerations are outside the scope of this project.



(a) temperature T=1

(b) temperature T=2

(c) critical temperature

(d) temperature T=4

Figure 9: plots of a lattice-size 200 at a specific temperature T ; the number of monte carlo cycles is selected in a way that we can assume that we are in the most likely state

4 source code

The source code can be found in out github-repository. The latex-file is in the merged file. `https://github.com/vincentn1/Project-4.git`