Monte Carlo Simulation of Ising Model

Eppenga, M.D., Faasse, G.M., Tirband Dastgerdi, C.

Computational Physics
TU Delft
Faculty of Applied Sciences
MSc program Applied Physics

Delft, May 9, 2018

ABSTRACT

The Ising model describes a system of interacting spins. In this report the Ising model is simulated using various Monte Carlo algorithms, like the metropolis, the heat-bath, the Wolff and the checkerboard algorithm. From the microscopic behaviour of the spin lattice, several macroscopic quantities, like specific heat, magnetisation and magnetic susceptibility are computed. These quantities have good qualitative agreement with respect to literature. The critical exponents in our simulation, describing the behaviour of the quantities near the phase transition are $0.128 \pm 6 \cdot 10^{-6}$ for the specific heat (theory predicts 0.125) and $0.63 \pm 5 \cdot 10^{-3}$ (theory predicts 1.75). Furthermore, the different algorithms are compared in terms of stability of the macroscopic quantities and the correlation time, stabilising time and running-time. Although the checkerboard algorithm runs the fastest in terms of MCS/s, the Wolff algorithm stabilises quicker and produces the most stable results. Lastly, the dependence of the simulation on grid-size is qualitatively comparable with theory and literature, due to the changing ratio of the grid-size and the correlation length

Contents

1 INTRODUCTION

The Ising model is one of the most well-known ways of describing a ferromagnetic system. Originating from the field of statistical physics, it models a system of interacting spins. Furthermore, this model predicts the notion of a phase transition, at which the system's properties change drastically.

In this report, Monte-Carlo methods are used to implement the Ising model, to compute several macroscopic quantities of ferromagnetic systems. Examples are the specific heat, the magnetic susceptibility and the magnetisation. Properties like the stabilisation times and macroscopic behaviour in the vicinity of the phase-transition are studied as well. Furthermore, performance and stability of the different algorithms are studied.

In this report, first a general description of Monte-Carlo methods is given. Then this method is applied on the Ising model and various numerical methods are discussed. After that, the results are given and discussed and lastly conclusions are posed.

2 THEORY

A famous model to simulate (anti)-ferromagnetism is the Ising model. The model consist of interacting spin particles sitting on a lattice. These spins can be in two different positions: Spin-up, denoted by $+1$ and Spin-down denoted by -1. The model was first solved analytically by Ising in one dimension, and later it was solved by Onsager in two dimension on a square lattice. The model is later generalised for more than two spins values in Potts model. In this section, the theoretical background on Monte Carlo models and their application to the different algorithms used in this study on the Ising model are described. In the section 'Methods', this theory is applied to the literal computational implementation of the different algorithms.

2.1 Ising model

In the Ising model a systems of spins is described by a Hamiltonian which considers the interaction between spins and their nearest neighbour spins together with interaction between the spins and an external magnetic fields. The Hamiltonian is given by equation the following equation:

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

in which s_i is the value of the i^{th} spin (which is ±1), h is the external magnetic field in the direction of the spins and J resembles the interaction constant.

A shortcoming of the Ising model is that it does not describe magnetism satisfactory [1] however it can describe atom absorbed on a lattice quite accurately [1]. The model, however, is relatively simple to simulate. Furthermore the Ising model on a two dimensional square lattice was the first model found that exhibits a phase transition [1].

The phase transition is clearly visible in the magnetisation: At the critical temperature a phase transition occurs from a ferromagnetic state to a non magnetic state. Onsager has showed that there is an exact solution for the critical temperature at $\frac{J}{k_b T_{crit}} \approx 0.44$ [2].

Another interesting effect of this model is that in the ferromagnetic state the magnetisation can be positive as well as negative due to the spin reversal symmetry of the model [1]. However, the symmetry can be broken by switching on a magnetic field denoted by h in the Hamiltonian.

2.2 Metropolis algorithm

Following the procedure in [1], the theory behind the Metropolis algorithm can be explained starting from the detailed balance equation.

$$T(X \to X')\rho(X) = T(X' \to X)\rho(X') \tag{2}$$

Here, $T(X \to X')$ is the transition probability from state X to X' and $\rho(X)$ is the probability of state X, given by the Boltzmann probability:

$$\rho(X) = \frac{1}{Z} e^{-\beta H(X)} \tag{3}$$

$T(X \to X')$ will be rewritten in terms of the trial step probability $\omega_{XX'}$ and the acceptance probability $A_{XX'}$. If $\omega_{XX'}$ and $\omega_{X'X}$ are equal, the detailed balance Equation 2 becomes:

$$\frac{A_{XX'}}{A_{X'X}} = \frac{\rho(X')}{\rho(X)} \tag{4}$$

The Metropolis algorithm is the solution where $A_{XX'}$ is chosen as equal to 1 if $\rho(X') \geq \rho X$ and equal to $\rho(X')/\rho(X)$ if $\rho(X') < \rho(X)$.

Using the probabilities as given in 3 it is seen that in the implementation of the Ising model the acceptance probability is 1 if $H(X) \geq H(X')$ and equal to $e^{\beta(H(X')-H(X))}$ if $H(X) < H(X')$. Since only the relative Boltzmann probabilities are of importance, computing the partition function Z is not necessary.

2.3 Heat-bath algorithm

The Heat-bath algorithm is a variation on the Metropolis algorithm. When a random spin is chosen, its orientation after the time step is independent of its spin before. The probability of a spin-up or spin-down state, P_+ and P_- respectively, are given by equation 5 [1].

$$\begin{aligned} P_+ &= \frac{e^{(2n_+-4)\beta J}}{e^{(2n_+-4)\beta J} + e^{-(2n_+-4)\beta J}} \\ P_- &= \frac{e^{-(2n_+-4)\beta J}}{e^{(2n_+-4)\beta J} + e^{-(2n_+-4)\beta J}} \end{aligned} \tag{5}$$

Here, n_+ is the number of neighbouring +-spins. The difference with the Metropolis algorithm mainly appears when the energy difference due to the spin-flip is close to zero. In Metropolis, a spin is always accepted when $\Delta E = 0$. With the Heat-bath algorithm, a spin flip is accepted with probability 0.5 when $\Delta E = 0$. This is equivalent to applying infinite Metropolis steps to the spin, which would result in it being up half of the time, and down the other half of the time. It is expected that implementation of the Heat-bath algorithm reduces the correlation time.

2.4 Checker-board algorithm

The checker-board algorithm is a more advanced version of the metropolis algorithm. It uses the same criteria as the metropolis algorithm to determine if a spin should be flipped or not. However, instead of flipping one random spin, the checker-board algorithm flips half of all the spins in the lattice at the same time. This is possible since each spin interacts only with its nearest neighbour, and therefore there are many spins on the lattice that do not interact with each other. To ensure that non of the spins that are simultaneous flipped interact witch is each other the lattice is divided in a checker board pattern of alternating colour as can be seen in figure 1. Then all the squares with the same colour can be updated simultaneously since the spins on these squares do not interact with each other.

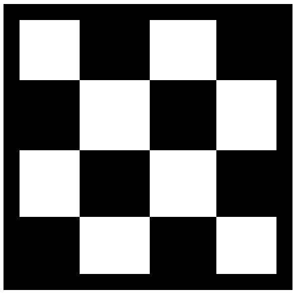


Figure 1: Checker-board configuration of an 10 by 10 lattice. The black or white squares represent the spins that can be updated simultaneous

This simultaneous updating of the spins comes with one constraint on the size of the lattice. As can be seen from figure 1 it is necessary that the lattice consists of an even number of spins along each axis. Otherwise the spins on one boundary side of the lattice would interact with the spins of the same colour at the other boundary side.

2.5 Wolff algorithm

In the Wolff algorithm, instead of flipping single spins, clusters of spins are flipped on each iteration. A random spin is selected, after which neighbouring spins with the same orientation are added to the cluster with a probability P_{add}. For the transition probabilities we then have:

$$\frac{\omega_{XX'}}{\omega_{X'X}} = (1 - P_{add})^{n-m}, \tag{6}$$

where n is the number of bonds between spins inside and outside the cluster in state X and m is the number of same-spin bonds between spins inside and outside the cluster in X'. With the knowledge that the energy difference between states X and X' equals $2(n-m)J$, and using Equations 2 and 3, we get for the acceptance probabilities:

$$\frac{A_{XX'}}{A_{X'X}} = ((1 - P_{add})e^{2\beta J})^{m-n} \tag{7}$$

In the Wolff algorithm we choose as a solution the one where P_{add} is equal to $1 - e^{-2\beta J}$ and the acceptance probability always equal to 1. That is, the cluster is built using P_{add} and always flipped.

Contrary to the Metropolis algorithm, the size of the time step is not constant for each loop iteration. Instead, the time size of an iteration is decided by the number of time a neighbouring spin is attempted to add to a cluster. In the low temperature limit, the time step per iteration equals 1 MCS.

3 METHODS

In the simulations, several methods of spin-flipping were used: the metropolis algorithm and its faster brother, the checker-board algorithm, the Wolff algorithm and lastly the heat-bath algorithm. For all the methods, a grid of $N \cdot N$ spins is used. To approximate an infinite grid of electrons, periodic boundary conditions are used.

3.1 Algorithms

In this section the implementation of the different algorithms is discussed. For all the algorithms a square grid of N by N spins is used with periodic boundary conditions.

3.1.1 Metropolis algorithm

The most straight-forward Monte-Carlo method used is the metropolis algorithm. This is an iterative method represented by the recipe below (based on the theory discussed before):

1. A random spin out of the grid is flipped
2. The energy difference (ΔE) between the old and the new state is computed
3. ΔE is checked:
   - If the energy difference is negative, the flip is accepted
   - If the energy difference is positive, the flip is accepted with probability $p_{acc} = e^{-\Delta E/(k_b T)}$ and is rejected with probability $1 - p_{acc}$
4. Return to step 1

Unfortunately this method is fairly slow in both stabilisation of the system and in computing time. Another drawback of this method is that grid can split up into two regions, with one region having the spins up and the other region having the spins down, when the grid is rapidly cooled down form a hot phase (random spin distribution). The other methods used tackle this problem mostly by parallel computing (checker-board) or by using the notion of spin clusters (Wolff).

3.1.2 Heat-bath algorithm

The Heat-bath algorithm is a variation on the metropolis algorithm, where the main difference is the acceptance probability for flipping a spin. Therefore the computational procedure is the same as the one described by the metropolis algorithm, only with a modified probability, as described in the theory section.

3.1.3 Checker-board algorithm

The checker-board algorithm is a sped up version of the metropolis algorithm by updating many spins at the same time. The iterative method is represented by the recipe bellow:

1. Look at all the spins on the white sections on the checker-board
2. Compute the energy difference (ΔE) between the old and new states
3. Check (ΔE) for each of the selected spins
    - If the energy difference is negative, the flip is accepted
    - If the energy difference is positive, the flip is accepted with probability $p_{acc} = e^{-\Delta E/(k_b T)}$ and is rejected with probability $1 - p_{acc}$
4. Repeat steps 1 to 3 for the black sections on the checker-board
5. Return to step 1

This algorithm is much faster than the metropolis algorithm since the updating of the spins can be performed with parallel computing. However, the checker-board algorithm still suffers from the same drawback as the metropolis algorithm of forming two regions of spin-up and spin-down in the lattice, when the lattice is rapidly cooled down form a hot phase to a cold phase.

3.1.4 Wolff-algorithm

The Wolff algorithm has a different approach to determining which spins should be flipped. In this case, the probability distribution used in other algorithms is not used to determine whether to flip spins, but whether to form bonds between spins. When a bond is formed between spins, they form a cluster, which is then flipped with a 50% probability. This is elaborated in the recipe below:

1. Form the bonds between neighbouring spins:
    - If the neighbouring spins are anti-parallel, no bond is formed
    - If the neighbouring spins are parallel, a bond is formed with probability $p_{bond} = e^{-2J/(k_b T)}$ and no bond is formed with probability $1 - p_{bond}$
2. Pick a random spin and check which spins belong to the same cluster as the chosen spin (by checking between which spins bonds are present). This is done using the back-track algorithm as found in the book by Jos Thijssen [1]
3. Flip all the spins in the cluster with probability 1/2
4. Return to step 1

The advantage of this method, is that the system stabilises faster, since entire clusters are flipped, instead of single spins. Furthermore, it stabilises way quicker when going through a phase-transition.

3.2 Macroscopic quantities

Several macroscopic quantities, like magnetisation, susceptibility and specific heat are computed. The mean magnetisation is simply given by:

$$m = (N_\uparrow - N_\downarrow)/N, \tag{8}$$

in which $N_\uparrow$ is the number of spins with spin up and $N_\downarrow$ is the number of spins with spin down and N is the total number of spins.
The specific heat per spin is proportional to the variance of the energy and is given by equation 9.

$$c_v = \frac{k_b \beta^2}{N} \cdot \mathrm{Var}(E), \tag{9}$$

in which $\beta = 1/(k_b T)$ and E is the energy of the system.
Lastly, the susceptibility is given by equation 10

$$\chi = \beta N \cdot \mathrm{Var}(m), \tag{10}$$

in which m is the mean magnetisation as given by equation 8

3.3 Critical exponents

Close to the phase transition, the behaviour of physical quantities, such as the specific heat, the susceptibility and magnetisation is characterised by so-called critical exponents. The different quantities are described as:

$$Q = a|T - T_{crit}|^b, \tag{11}$$

in which Q is the given quantity, T is the temperature, T_{crit} is the temperature of the phase transition, a is a fit parameter and b is the critical exponent.

3.4 Correlation time

The energy and magnetisation of the lattice between subsequent time steps are heavily correlated, since the updated lattice is obtained by flipping some spins of the previous time step. In order for computation of macroscopic quantities such as those in 3.2 to remain valid, the range of time steps must be significantly larger than the correlation time. In order to get an impression of the order of magnitude of the correlation time, the normalised autocorrelation function of the energy will be computed, and averaged over several realisations. Then the autocorrelation function will be fitted to $e^{-t/\tau}$, from which τ, the correlation time, will be determined.

3.5 Measures of time

In Monte-Carlo simulations an important measure of time is the Monte Carlo Step (MCS). In the case of the Ising model, one MCS is defined as the number of loop iterations required to consider each spin in the grid one time. In the case of the metropolis algorithm, an MCS is therefore expressed in the number of loops (N_{loops}) as MCS $= N_{loops}/N_{spins}$, since for every N_{spins} loops, every spins is visited once one average.
For the checker-board algorithm, every loop $N_{spins}/2$ spins are checked. Therefore, the MCS $= N_{loops}/2$, since every two loops, all spins are visited.
For the Wolff algorithm, the definition is dependent on the number of spins visited in one loop (n_{visit}), which is no constant in general. Therefore, the MCS is defined as MCS $= N_{loops} \cdot n_{visit}/N_{spins}$. Note that n_{visit} is dependent on the time-step and has to be calculated separately for each loop.

4 RESULTS AND DISCUSSION

4.1 Correlation time

In Figure 2, the computed energy auto-correlation functions are plotted for different algorithms. They are obtained at a temperature of $k_bT/J = 2$. As expected, the energy auto-correlation function shows a fall-off that resembles an exponential function.

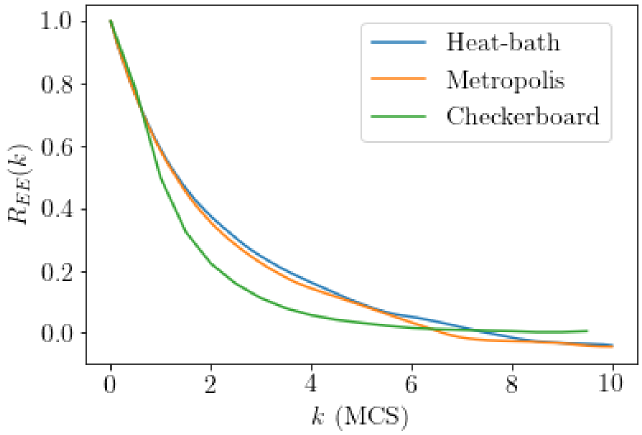


Figure 2: Auto-correlation function of the energy R_{EE} as a function for lags k up to 10 MCS for $k_bT/J = 2$ on a 20x20 grid, averaged over 50 runs of 1000 MCS each.

The figure refutes the expectation that heat-bath could decrease the correlation time, showing no significant improvement over the metropolis algorithm. It may be possible for the heat-bath algorithm to perform better than the regular metropolis algorithm in case of Potts models, in which the spin is allowed to have multiple directions (rather than just up or down) [3].

The checker-board algorithm, however, shows a smaller correlation time than the heat-bath and metropolis algorithm, despite simply being a parallelised version of metropolis. This can be made intuitive sense of by realising that in the implementation of metropolis used here, the spins are selected at random. While for the checker-board algorithm, the two checker-boards are visited alternately. When the spins are selected at random, it is possible that the same spin is selected in quick succession, which does not greatly decorrelate the energy.

Each of the curves have been fitted to the exponential function $e^{-t/\tau}$ with τ the fitting parameter indicating the correlation time. The acquired values are 1.41 MCS ±0.05 MCS for checker-board, 1.98 MCS ±0.08 MCS for metropolis, and 2.01 MCS ±0.06 MCS for the heat-bath algorithm. The auto-correlation function for the Wolff algorithm is not shown in this report and can not be computed directly, as the time in MCS between each sample is not constant for the Wolff algorithm. One way of making computation possible is by interpolating the energy of the Wolff algorithm in such a way that time between timesteps is constant, and then computing the auto-correlation as usual.

These results are near the phase transition and show a correlation time in the order of 1-2 MCS. This justifies the computation of thermodynamic quantities such as those in section 3.2 as long as the code is ran for significantly longer than the correlation time. Most results in this report describing thermodynamic quantities are computed across run times of 1000 MCS.

4.2 Macroscopic quantities

The main goal of this project is to simulate the behaviour of macroscopic quantities based on the microscopic Ising model, which includes individual spins. In the next subsections, this behaviour is graphically displayed, while section 4.4 handles quantitative findings.

4.2.1 Magnetisation

Another method to verify the algorithm is by looking at the magnetisation m of the lattice versus the dimensionless number $\frac{k_b T}{J}$. At low temperature the lattice should have a finite magnetisation and at high temperature the magnetisation should be zero. Furthermore a phase transition from the finite magnetisation to the non magnetisation phase should occur at $\frac{k_b T}{J} \approx 2.27$ [2]. Also, when there is no magnetic field present, the lattice can display as well a positive or negative magnetisation. When there is a magnetic field present this symmetry should break and the spins should always align along the magnetic field.

These effects are simulated with the checker-board algorithm. In order to see the effect of symmetry breaking, the spins on the lattice are initialised in a hot phase (i.e. each spins has a random value on the lattice). The results can be seen in figure 3.

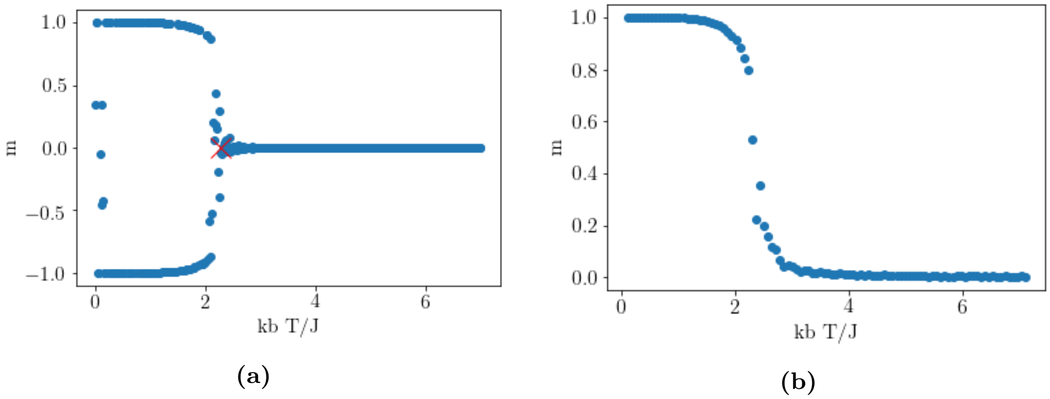


Figure 3: a) Plot of the magnetisation for the checker-board algorithm without external magnetic field. **b)** Plot of the magnetisation for the checker-board algorithm with external magnetic field

From figure 3a the phase transition can clearly be seen at approximately $\frac{k_b T}{J} \approx 2.27$. Also one can see that a few points in the low temperature phase have a quite different magnetisation then one would expect. This can be explained by the splitting in two regions of spin-up and spin-down when the lattice is cooled rapidly from a hot phase to a cold phase (a 'stalemate' situation).
Furthermore, from figure 3a and 3b it can clearly be seen that a magnetic field breaks the spin symmetry.

Another verification is the behaviour of the simulation close to the transition temperature. In figure 4, the magnetisation is plotted for $T = 1/0.44$, which is close to the phase transition.
As seen in figure 4, the magnetisation close to the transition temperature fluctuates a lot between fully positive and fully negative magnetisation, as seen in literature [1], making simulations in this regime with the checker-board algorithm problematic. The solution used in this report to still be

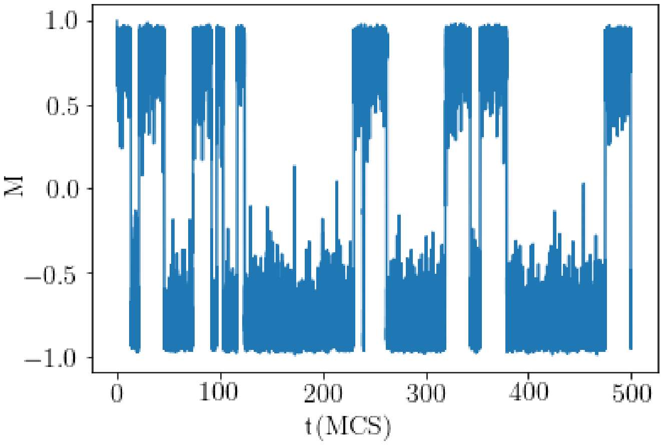


Figure 4: A plot of the magnetisation versus time for the checker-board algorithm close to the phase transition.

able to compute the susceptibility (which depends on the variance of the magnetisation) is to use the variance of the absolute value of the magnetisation. Although this distorts our system [1], no better realisable options were found.

4.2.2 Specific heat and susceptibility

To compare the simulation to known results, the plot in the book by Jos Thijssen [1] is reconstructed for the specific heat. The results for the Wolff and checker-board algorithm are shown in figure 5a. Furthermore, the susceptibility is plotted as function of the scaled temperature in figure 5b

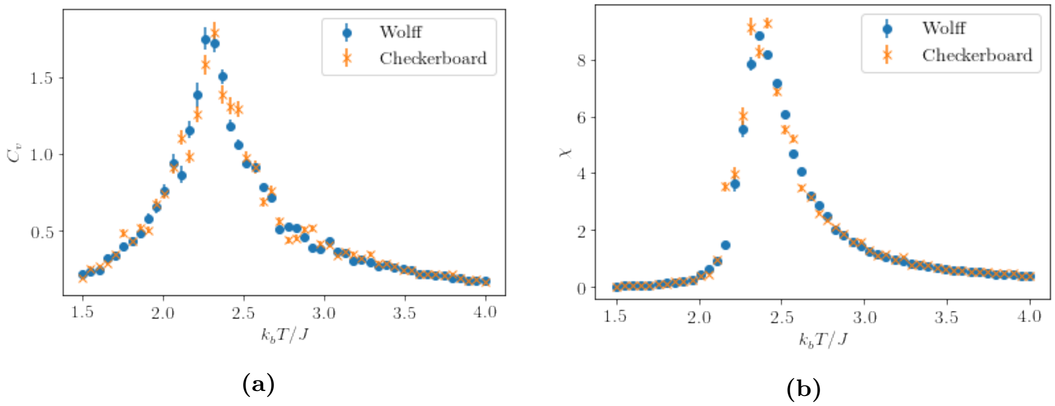


Figure 5: **a)** Plot of the specific heat versus scaled temperature with error-bars for the Wolff and checker-board algorithm. **b)** Plot of the susceptibility (χ) versus scaled temperature with error-bars for the Wolff and checker-board algorithm. In both **a)** and **b)**, the phase transition is clearly visible for both algorithms as a peak. In both cases the algorithms ran for 1000 MCS

When comparing figure 5a to the results in the literature [1], the simulation gives the expected results, with the peak of the C_v at the transition temperature of the system. Also the susceptibility shows the expected peak in the phase transition and the same qualitative behaviour as found in similar projects[4].

Furthermore, it is seen the two algorithms give similar results for both quantities. However, the checker-board algorithm fluctuates more than the Wolff algorithm for the same amount of MCS, especially around the phase transition, as expected.

4.3 Comparing algorithms

Several tests were done regarding the performance of the different algorithms. First a simulation of 10 MCS was done, with the scaled temperature going from 1.5 to 4 in 50 steps. The computation time for a variable number of spins is displayed in figure 6 It is seen that the checker-board algorithm is by far the fastest when looking at raw speed in MCS/s. Furthermore, for the checker-board algorithm the speed depends weakly on the number of spins due to the parallel implementation of the metropolis algorithm. Therefore it is very useful for large simulations. However, as already mentioned, the checker-board algorithm fluctuates a lot around the phase transition (seen in figure 4 and 5). Therefore, to have stable results around these temperatures, long checker-board simulations are necessary. The Wolff algorithm is known to stabilise quicker when going through the phase transition. To verify this claim, figure 7 shows the time-dependency of the magnetisation for the different algorithms. In this case the initial configuration was completely random (resembling a high temperature) and the temperature in the simulation is set to 0.1 k_bT/J, which is well below the phase transition temperature.

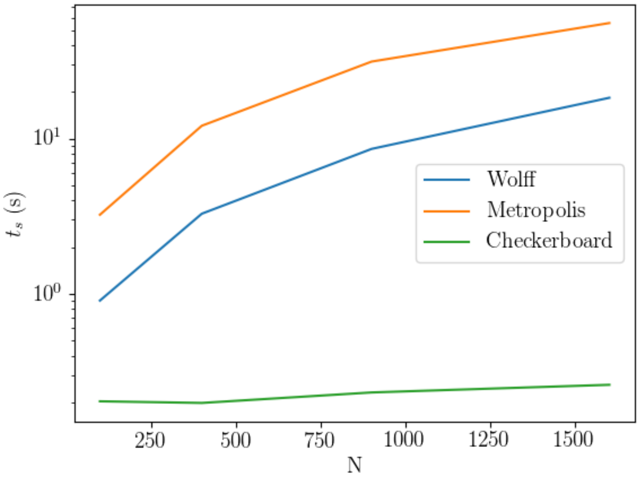


Figure 6: Semi-logarithmic plot of simulation time (t_s) versus the number of spins in the lattice (N) for the different algorithms

As seen in figure 7, the Wolff algorithm stabilises within several MCS, while the metropolis and checker-board algorithm need significantly more simulation time as expected, making the Wolff algorithm the most suitable methods for computing physical quantities around the phase transition.

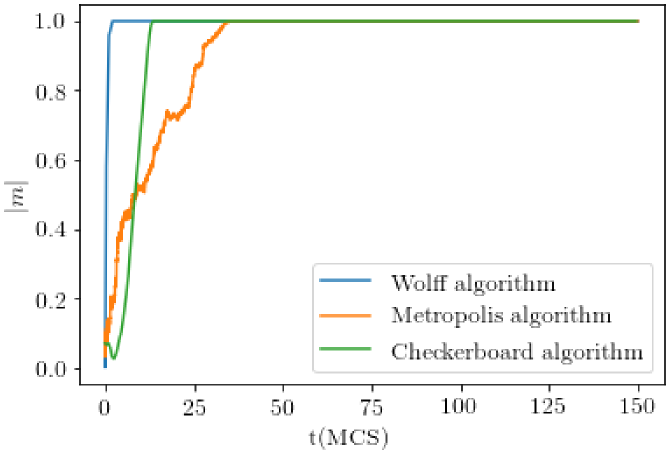


Figure 7: Plot of the magnetisation versus simulation time (MCS) for the different algorithms when transitioning from a high temperature

4.4 Critical Exponents

Other interesting quantities are the critical exponents that describe the behaviour of a given physical quantity near the critical point. The critical exponents are determined form simulation with the Wolff algorithm for a lattice size of 30 by 30 spins.

The critical exponents for the magnetisation (β) and susceptibility (γ) are determined by fitting the function $y = a * |T - T_{crit}|^b$ to the found values of these quantities. A temperature range of $\Delta T = 0.5$ around the critical temperature is used.

The critical exponent of the specific heat is $\alpha = 0$. This means that the specific heat is described by a logarithmic function near the critical temperature. Therefore the critical exponent itself cannot be fitted. However, it is determined how well a logarithmic function can be fitted to the data.

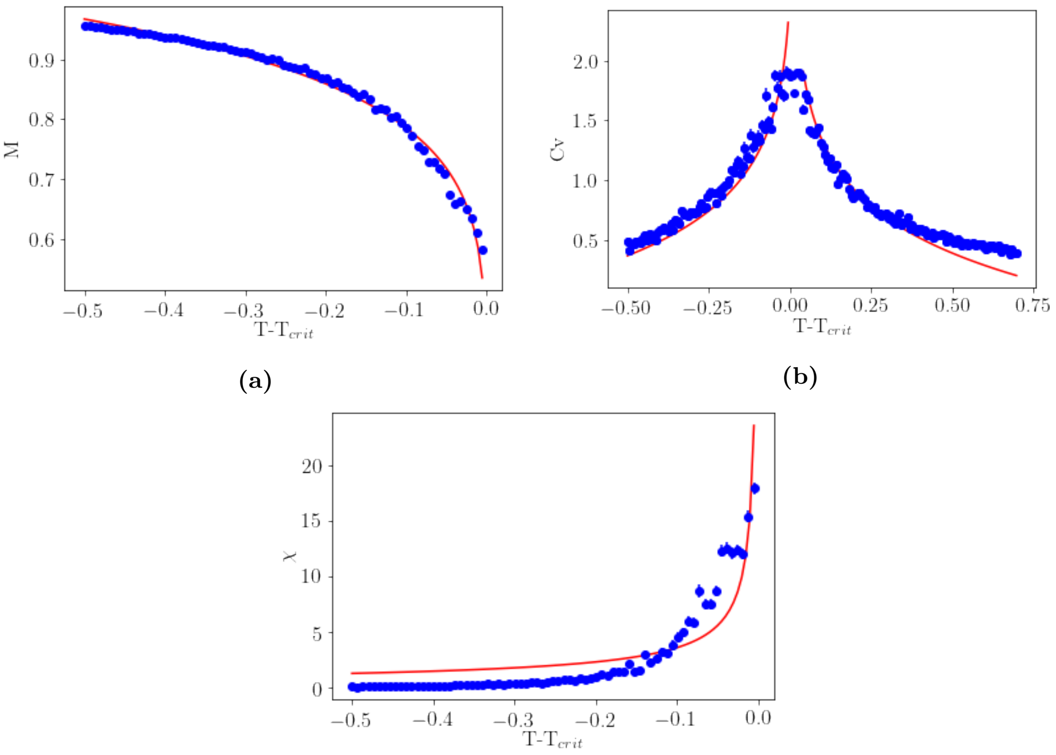
From figure 8a one can see that the magnetisation is good described by the power law. Furthermore one can see from table 1 that the value of the found critical exponent is in good agreement with the literature value.

Furthermore one can see from figure 8b that the behaviour specific heat is approximated well by a logarithmic function.

The susceptibility, however, deviates a lot more form the expected power law as can be seen in figure 8c. Also from table 1 it can be seen that the found critical exponent is not even near the value found in literature.

Table 1: Literature and calculated values for the critical exponents of the 2d Ising model and fitting parameter

Critical exponent	Literature value	Calculated value	Fitting parameter
α	0	0	$-0.55 \pm 6 \cdot 10^{-5}$
β	$\frac{1}{8}$	$0.128 \pm 6 \cdot 10^{-6}$	$1.06 \pm 6 \cdot 10^{-6}$
γ	$\frac{7}{4}$	$0.63 \pm 5 \cdot 10^{-3}$	$0.86 \pm 5 \cdot 10^{-3}$



(a)

(b)

(c)

Figure 8: Fit of critical exponents for the magnetisation (a), specific heat (b) and the susceptibility (c), in which the red lines represent the fit and the blue dots represent the simulated data.

An explanation for the susceptibility fit being significantly worse than those of the specific heat and magnetisation is the fact that for calculating χ, the absolute value of the magnetisation was used. This distorts the system and the effects on the results is not easily predicted and could lead to the deviation in the critical exponent as discussed in section 4.2.1.

4.5 Influence of different grid sizes

An interesting property of the 2 dimensional Ising model is that it exhibits a second order phase transition. This phase transition can be seen, for instance, with the magnetisation of the lattice. Onsager showed that the exact critical temperature of this transition is $\frac{J}{k_b T_{crit}} \approx \frac{1}{0.44}$ [2]. The exact solution is however only valid in the thermodynamic limit. In the simulations one will always have a finite grid size and usually use periodic boundary conditions. The result is that the spins on the lattice becomes ordered when the correlation length becomes the same size as the grid size. This ensures that the critical temperature of the lattice will have a different value than the exact solution.

To show this effect, the magnetisation of the lattice is determined with the Wolff algorithm and plotted for different grid sites in figure 9. From this figure it can clearly be seen that the critical temperature of the lattice approaches to the exact value with increasing grid size.

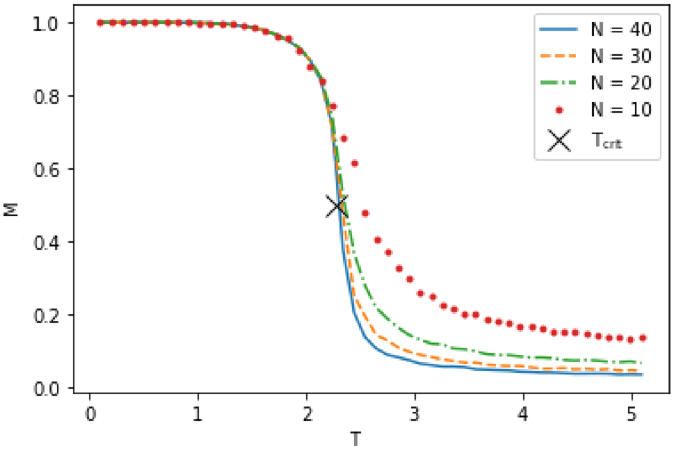


Figure 9: Plot of the magnetisation of the Ising model with the Wolff algorithm for different grid sizes

5 CONCLUSION

In this report, the results of an Ising model are presented. Macroscopic quantities such as the specific heat, magnetisation, and susceptibility are computed as well as their critical exponents. Several algorithms have been discussed in this report and used for the acquisition of these quantities: Metropolis, heat-bath, Wolff, and checker-board. These are compared in terms of speed, correlation time, and time it takes to cool down through a phase transition.

All algorithms show similar results in computation of the magnetisation, specific heat, and susceptibility, and are consistent with literature. In each of these quantities, a phase transition is clearly identified near the theoretical value of $kBT/J \approx 2.7$.

Near the critical temperature, the macroscopic quantities exhibit power-law behaviour as a function of the $|T - T_c|$. For the specific heat, this power-law exponent was found to be $\beta = 0.128$ using the Wolff algorithm and a 30x30 lattice, consistent with the theoretical value of $1/8$. For the susceptibility however, this exponent was found to be $\gamma = 0.63$, compared to the $7/4$ found in literature. This discrepancy may be the result of the small 30x30 lattice that was chosen as well as inaccurate assessment of the critical temperature for these lattice dimensions.

The algorithms showed similar different auto-correlation behaviour. The correlation time of the energy for the checker-board was found to be 1.41 MCS ±0.05, 1.98 MCS ±0.08 for metropolis, and 2.01 MCS ±0.06 MCS for heat-bath. For Wolff, the auto-correlation function can not be computed directly because the time between samples is not constant, but it is assumed to be in the same order of magnitude.

The checker-board algorithm has proven to be the fastest algorithm to compute a set number of MCS. Its parallel nature causes it to scale favourably with increasing number of spins compared to other algorithms.

Despite the periodic boundary conditions of the lattice, the finite grid size does have an influence on the results. The location of the phase transition shifts to higher temperatures for decreasing lattice sizes, with the analytical value only being valid in the thermodynamic limit as[2].

6 REFLECTION GROUP WORK

In this study, we worked together as a group of three students. The bulk of the programming, we performed together, as one of us coded and the other two brought up ideas for solving the problems we ran into along the way. We varied these roles frequently, such that everyone got to do a lot of everything. Some parts were easily decoupled and distributed over the team, such that we could work on these individually (such as implementing different Monte Carlo algorithms, making the animation, writing functions for calculating macroscopic properties, bug-fixing and so on). Also in this case there was clear communication as to what problems were encountered, such that we could all bring in ideas for every separated part.

Throughout the project, the work-load was evenly distributed. This included the last week, in which the report had to be written and the code had to be cleaned up. In writing the report, the following distribution was made:
Martijn: Methods (checker-board algorithm), Theory, Results (magnetisation, critical exponents, influence of grid-size).
Cyrus: Conclusion, Theory, Methods (heat bath, correlation time), Results (correlation time).
Richard: Introduction, Abstract Methods (Metropolis, Wolff, Quantities, critical exponents, correlation time, measures of time), Results (specific heat/susceptibility, comparing algorithms).
All in all, the collaboration was well-coordinated and the workload was evenly distributed.

References

[1] J. Thijssen. *Computational Physics.* Cambridge University Press, 2007.

[2] L. Onsager. Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Phys. Rev.*, 65:117–149, Feb 1944.

[3] E. Carlon. Advanced monte carlo methods. http://itf.fys.kuleuven.be/ẽnrico/Teaching/monte_carlo_2012ṗdf, 2012.

[4] Lilian Witthauer and Manuel Dieterle. The phase transition of the 2d-ising model. http://quantumtheory.physik.unibas.ch/people/bruder/Semesterprojekte2007/p1/Ising.pdf, 2007.