Ising model simulations with the Metropolis and Wolff algorithm

Vincent P. Post*
HuygensKamerlingh Onnes Laboratory, Leiden University,
P.O. Box 9504, 2300 RA Leiden, The Netherlands
(Dated: May 3, 2016)

Monte Carlo simulations can be used in statistical mechanics for obtaining the thermodynamic properties and to quantify the phase transitions of a system. One system it is used for is the Ising model, which in three dimensions has no theoretical model explaining it. I present the theory behind Monte Carlo simulations and explain two algorithms: Metropolis & Wolff. From both algorithms the magnetization and magnetic susceptibility are obtained for the two dimensional Ising model, which show signs of a phase transition in accordance with the theory. Also shown that the critical exponents obtained from finite size scaling are in two dimension in good agreement with the theory and in three dimensions are close to the values of other Monte Carlo simulations.

I. INTRODUCTION

The Ising model consists of spins on a lattice which are in one of two states: up or down. It allows one to model ferromagnetism and study it's phase transition properties. The Ising model on a square lattice with external magnetic field $B = 0$ is solved in two dimensions by Onsager [1] but a solution for three dimensions still eludes us. One method to describe the thermodynamic properties of a system in three dimensions is achieved by Monte Carlo simulations. Simulating the system in contact with a heat reservoir, from which the thermal fluctuations flips the spins. Moving the system from one state to another.

Monte Carlo methods are especially used for critical phenomena. The phenomena where, for example, second order phase transition take place. For in the Ising model the system can no longer maintain its spontaneous magnetization above the corresponding phase transition temperature, so called critical temperature [1] $T_c = 2.269$, losing its ability to be magnetized or attracted to a magnet disappears. Due to the universality of the critical exponents, exponents describing the critical phenomena, we can use a simple model to determine them and find the correct values of a much more complicated real-world system [2]. As example measurements from different sized two dimensional finite lattice's give the corresponding critical exponents of the critical phenomena in the Ising model. No need for simulations on a as large as possible lattice.

In the so called critical region, where in the Ising model the phase transition take place, Monte Carlo algorithms tend to be slow. Due to that large cluster of aligned spins are formed so that when they flip they produce a huge fluctuation in the measured thermodynamic properties. Also called critical fluctuations. This has a consequence that it takes a while for the system to be independent of the foregoing states so that one can obtain independent measurements of the thermodynamic properties. This phenomena is called critical slowing down and can be less severe for different Monte Carlo algorithms.

* post@physics.leidenuniv.nl

The paper is constructed as follow: First the theory behind the Ising model and Monte Carlo simulations is explained. Next the implementation of two Monte Carlo algorithms is shown: the Metropolis algorithm [3] and the Wolff algorithm [4]. Implementing what is discussed in the book by G.T. Barkema and M.E.J. Newman [2]. We compare for both algorithms the obtained measurement of the magnetization per spin (m) and of the magnetic susceptibility (χ) with the exact solution of the two dimensional Ising model. In the end the Wolff algorithm is used to obtain the critical exponents by the finite size scaling method and are compared with the theoretical value in two dimensions and with other Monte Carlo simulations in three dimensions

II. THEORY

The Ising hamiltonian (eq 1) is influenced by an external thermal reservoir changing the system from one state into another state, by flipping spins on the lattice. The time-evolution of the system can therefore be described by a master equation (eq 2)

$$H = -J \sum_{<ij>} s_i s_j - B \sum_i s_i \tag{1}$$

The summation $< ij >$ is over the nearest neighbours of the spin s_i. Where an up spin has a value of $+1$ and a down spin a value of -1.

$$\frac{dw_\mu}{dt} = \sum_\nu [w_\nu(t) R(\nu \to \mu) - w_\mu(t) R(\mu \to \nu)] \tag{2}$$

Where $R(\mu \to \nu)$ is the transition rate for transition μ to ν from which we define that $R(\mu \to \nu)dt = P(\mu \to \nu)$ is the probability that the system is in state ν after a time dt later from state μ. P is called the transition probability. The w are a set of weights which represent the probability that the system will be in state μ at time t.

The weights are related to the macroscopic properties

of a system by

$$\langle Q \rangle = \sum_{\mu} Q_{\mu} w_{\mu}(t) \tag{3}$$

The expectation value of Q is seen as a time average of the quantity Q as long as we are in an equilibrium state of the system. Which means that the two terms on the right hand side in the master equation (eq 2) cancel each other for all μ. Letting the weights w_{μ} all have a constant value for the rest of the time. From the thermal nature of the interaction between the system and the thermal reservoir (at a temperature T) the transition rates take on particular values.

As it is shown by Gibbs [5] that this thermal interaction in equilibrium gives a specific equilibrium occupation probability, also known as the Boltzmann distribution.

$$p_{\mu} = \frac{1}{Z} e^{-\beta E_{\mu}} \quad \text{with} \quad p_{\mu} = \lim_{t \to \infty} w_{\mu}(t) \tag{4}$$

In which E_{μ} is the energy of state μ, k the Boltzmann constant and $\beta = 1/kT$. The Z is the partition function, which can be used to determine most of the thermodynamic variables of a system. Which will be used to check if the Metropolis algorithm is working accordingly, but not employed further because of the enormous amount of calculations needed see eq 6.

From this the expectation value of a quantity Q (eq 3) for a system in equilibrium can be rewritten, with the definition of the partition function filled in

$$\langle Q \rangle = \frac{1}{Z} \sum_{\mu} Q_{\mu} e^{-\beta E_{\mu}} = \frac{\sum_{\mu} Q_{\mu} e^{-\beta E_{\mu}}}{\sum_{\mu} e^{-\beta E_{\mu}}} \tag{5}$$

The corresponding partition function for the Ising model is

$$Z = \sum_{s_1 = \pm 1} \sum_{s_2 = \pm 1} \cdots \sum_{s_N = \pm 1} \exp[\beta J \sum_{<ij>} s_i s_j + \beta B \sum_{i} s_i] \tag{6}$$

From which one sees that the direct calculation of the partition function, even with $B = 0$, is incredible computational expansive. Due to the summation over all the possible states of the system. For a lattice with N spins which are either up or down there are a total of 2^N states, giving for a simple 5×5 (N=25 spins) lattice: 33 554 432 possible states.

To represent a real life system, an infinite lattice, a simulation on a as large as possible lattice is needed. Therefore evaluation of the partition function is not viable, a 6×6 lattice may take about two years [2]. But with Monte Carlo simulations, the states of the system are sampled by the Boltzmann probability distribution (eq 4) making states less likely at different energy/temperatures. The Monte Carlo simulation will therefore not go over all the possible states of the system. But it is still possible to reach every state from any other state (condition of ergodicity).

For the Monte Carlo simulations to be representative of a real system it needs to obey the following rules:

1) Markov Process: mechanism to generate a new state of a system, from $\mu \to \nu$, in a random way. The corresponding probability is the transition probability which must satisfy the constraint

$$\sum_{\nu} P(\mu \to \nu) = 1 \tag{7}$$

A true Markov process must also not vary over time and should depend only on the properties of the current states μ and ν. In order to achieve a succession of states which obtain their probability's from the Boltzmann distribution two more conditions need to be introduced. When these probability's are achieved the system has come to equilibrium.

2) Condition of ergodicity: requirement that from a Markov process every state of the system can be reached from any other state, as long as the program is run long enough.

3) Condition of detailed balance: lays a second constraint onto the transition probabilities to ensure the Boltzmann probability distribution

$$\frac{P(\mu \to \nu)}{P(\nu \to \mu)} = \frac{p_{\nu}}{p_{\mu}} = e^{-\beta(E_{\nu} - E_{\mu})} \tag{8}$$

After the system has come to equilibrium, we can as shown before calculate the time average of a quantity Q (eq 5). Which is actually a calculation of the estimator of Q, Q_M. Only in the limit of averaging Q over an infinite states the estimator is equal to the expectation value, $M \to \infty : Q_M =< Q >$.

From the condition of detailed balance there is a good deal of freedom in the choose of the transition probabilities. Therefore we split the transition probability up in a selection probability part (g), a probability of the algorithm to generate a new target state ν, and an acceptance ratio (A) part, what fraction of the time we will accept the change to a state ν.

$$\frac{P(\mu \to \nu)}{P(\nu \to \mu)} = \frac{g(\mu \to \nu) A(\mu \to \nu)}{g(\nu \to \mu) A(\nu \to \mu)} = e^{-\beta(E_{\nu} - E_{\mu})} \tag{9}$$

From this equation the ratio of A can take any value between zero and infinity thereby both $g's$ can take any values we like. But we want our algorithm to accept as many as possible steps to a new state, so that there is no waste of time by staying in the same state. This problem is solved by making the acceptance ratio be as close to unity as possible. In practice one of the two acceptance probability's is set to one while the other still satisfies the ratio. The aim is to put as much info from the transition probabilities on the characteristics of the states μ and ν into the selection probabilities and as little as possible into the acceptance ratio.

Before measurements can be made a lattice must be initialised, which can be all spins up/down, corresponding to a low temperature state, or a random lattice corresponding to a high temperature system. First it needs

to be investigated how long for both initialisations of the lattice the lattice thermalizes, for the rest of the simulation a thermodynamic property keeps fluctuating around one value. So that after the simulation has come to equilibrium and the system is thermalized measurements can be made of the thermodynamic variables which one wants to know.

For both algorithms the measurements come from the simulation of a square lattice with periodic boundary conditions. Where the external magnetic field $B = 0$ and the interaction energy and Boltzmann constant is set to one ($J = 1, k = 1$). Setting the temperature dimension into energy units. The lattice is initialised with all spins up or down (same energy contribution) and then will be evolved to the lowest temperature one wants to measure. After the measurements are made at this temperature the lattice is stored and reused at the next higher temperature in the temperature sorted array to evolve at this temperature. So on to the highest temperature one wants to measure. In this manner the thermalisation times are small and the system has a lower chance of falling into a local minima, if the temperature steps are not too large.

The thermodynamic variables measured are: the magnetization per spin(m), magnetic susceptibility (χ) and the specific heat (c, only for the Mertropolis algorithm). Expressed with $J = 1, k = 1$ so that $\beta = 1/T$.

$$\langle m \rangle = \frac{1}{N} \left\langle \sum_i s_i \right\rangle \tag{10}$$

$$\chi = \beta N (< m^2 > - < m >^2) \tag{11}$$

$$c = \frac{\beta^2}{N} (< E^2 > - < E >^2) \tag{12}$$

For the measurements ideally the new state needs to be independent of the foregoing state, otherwise the mean of the thermodynamic property will not be representative of the systems value of it. A measurement is independent when the samples are drawn two correlation times apart, 2τ. Where the correlation time is the time where the time displaced autocorrelation ($\chi(t)$) is equal to $1/e$:

$$\chi(t) \sim e^{-t/\tau} \tag{13}$$

In which the time displaced autocorrelation for the magnetization can be calculated, if the measurements are evenly spaced, as follow

$$\chi_m(t) = \frac{1}{t_{max} - t} \sum_{t'=0}^{t_{max}-t} m(t')m(t'+t) - \frac{1}{t_{max} - t} \sum_{t'=0}^{t_{max}-t} m(t') \times \frac{1}{t_{max} - t} \sum_{t'=0}^{t_{max}-t} m(t'+t) \tag{14}$$

With t_{max} a chosen maximum time based on how many independent measurements(n) one wants, $t_{max} = 2\tau n$. When t gets close to t_{max} one ends up integrating over a small time interval increasing the statistical errors due to the random nature of the thermodynamic variable ($m(t)$). It is chosen that t goes at maximum to $t_{max}/2$. In practice dependent measurement will be made because usually the correlation time is only known after a simulation is finished. As long as the dependent measurements between independent measurements are measured the same amount of times, it will not mess up the mean.

For a measure of the accuracy of the results error analysis is applied. There are statistical errors, due to randomness from thermal fluctuations, and systematic error, due to the procedure used e.g. only wait a finite amount of time for the system to thermalize. Because there is no good general method for estimating systematic errors [2] it will not be investigated further.

Estimation of the statistical error is given by

$$\sigma = \sqrt{\frac{1}{n-1}(\bar{m^2} - \bar{m}^2)} \tag{15}$$

But this equation assumes that the samples m_i are statistically independent, which they will not be. Therefore the equation is rewritten and assumed that many measurements will be made [2]

$$\sigma = \sqrt{\frac{2\tau}{t_{max}}(\bar{m^2} - \bar{m}^2)} \tag{16}$$

Both the specific heat and magnetic susceptibility are calculated from two averages, of which the averages are correlated with each other. Therefore eq 15/16 can not be applied. By means of bootstrapping one can obtain errors for average dependent macroscopic quantity's. From the n measurements of E or m one selects n at random with replacement with which we recalculate c or χ. It is shown with these recalculated values we can use the standard deviation to obtain a measure of the error in c and χ [6].

$$\sigma = \sqrt{\bar{c^2} - \bar{c}^2} \tag{17}$$

For obtaining the values of the critical exponents the finite size scaling method is used. The critical exponents are obtained by observing how the measured quantities vary as the size L of the system changes. Where L is the factor multiplied at each dimension, for two dimensions $N = L \times L$. Only the critical exponents from the magnetic susceptibility are determined, γ and ν. And the critical temperature T_c is found. It is shown that the magnetic susceptibility can be written as [2]

$$\chi = L^{\gamma/\nu} \tilde{\chi}(L^{1/\nu} t) \tag{18}$$

It shows how the susceptibility varies with the system size L for systems close to the critical temperature. Wherein $\tilde{\chi}$ is called the scaling function, which will be the same for all system sizes. By plotting the scaling function, which are the measurements obtained from the algorithm close to the guessed critical temperature, against $L^{1/\nu}t$ can

give one the critical exponents. With $t = (T - T_c)/T_c$. The critical exponents are correct if the data points from multiple system sizes overlap, the points collapse on the same curve. The errors are obtained by slightly varying the found values of the critical exponents and investigate the graph when the collapse is not 'good' anymore. This is a very rough error estimate, for a better estimate see [2].

III. METROPOLIS ALGORITHM

The Metropolis algorithm has a so called single spin flip dynamic one step of the algorithm only allows for a single spin to flip. The algorithm picks a spin at random from a lattice looks at it's nearest neighbours determines the interaction energy between them and if some condition is met the spin will be flipped. It picks a random spin as many times as there are spins on the lattice, also called one sweep of the lattice. When one sweep is done a measurement is made, only after the system is thermalized, and a next sweep can be made. The single spin flip dynamic guarantees that the new state ν will have an energy of E_ν differing at most from the current energy(E_μ) $2J$ for each bond between the spin we flip and it's nearest neighbours. For a square lattice with $J = 1$ this new state will only differ a discrete amount, as seen in eq 20.

The algorithm chooses for the selection probabilities $g(\mu \to \nu) = 1/N$ with N the total number of spins on the lattice. Which is equal for all possible states of ν. There are N possible spins to flip and therefore N selection probabilities, the rest are set to zero. With this choose the condition of detailed balance, eq 9, will have the ratio of g equal to one. The acceptance ratio is

$$A(\mu \to \nu) = \begin{cases} e^{-\beta(E_\nu - E_\mu)} & \text{if } E_\nu - E_\mu > 0 \\ 1 & \text{otherwise} \end{cases} \tag{19}$$

Satisfying detailed balance while having the acceptance ratio as close to unity as possible. It says if a new configuration has a lower energy it always flips the spin under investigation. If it has a higher energy there is a probability $e^{-\beta(E_\nu - E_\mu)}$ to flip the spin. The spin will flip if the value is higher than a random float between 0 and 1. The random number is created by the Mersenne Twister pseudorandom number generator [7], which the writer does not know (hopes) if it is random enough for the shown Monte Carlo simulations. If it is not completely random the simulations may become not representative of a real system. The single spin flip dynamic allows the lattice to get from any state to any other state thereby satisfying ergodicity. Together with detailed balance it guarantees that the multiple Markov processes will generate the Boltzmann probabilities, after thermalization. The Metropolis algorithm is representative of a real system.

After initialisation of the lattice the total energy is calculated directly from the hamiltonian, eq 1, and the total magnetization as a sum over all the spins. It is shown [2] that when a spin flips the new energy and magnetization is updated as

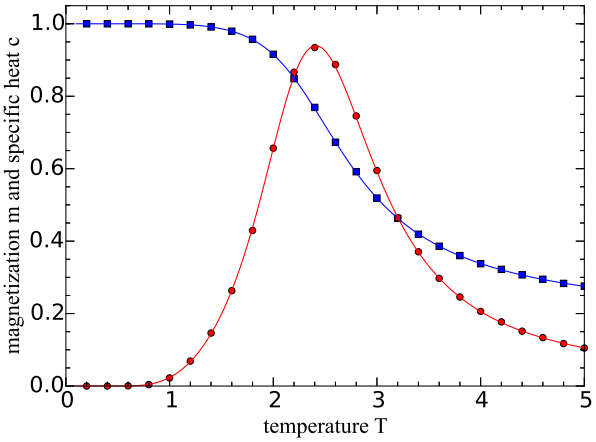

FIG. 1. The magnetization (blue squares) and the specific heat (red circles) per spin for a 5×5 lattice of a Metropolis Monte Carlo calculation. Compared with the exact calculation from the partition function.

$$E_\nu = E_\mu + 2Js_k^\mu \sum_{i \text{ n.n. to } k} s_i^\mu \tag{20}$$

$$M_\nu = M_\mu + 2s_k^\nu \quad \text{with} \quad M_\mu = \sum_i s_i^\mu \tag{21}$$

Which is found to be faster then a complete calculation of the energy and magnetization.

Now the framework for the algorithm is in place a quick test is made to see if everything is working accordingly. This is done by comparing the magnetization (eq 10) and specific heat(eq 12) measurements with the direct, computational expansive, evaluation of the partition function for a 5×5 lattice, see eq 5/6. For both the magnetization and specific heat 1000 measurements are made but no thermalisation times, correlation times and errors are calculated yet. Therefore this will only give a rough idea if the algorithm is working accordingly. Figure 1 shows that the algorithm is working accordingly, it returns the values in agreement with the exact calculation from the partition function.

For a complete correct calculation first the thermalization times need to be found at the temperatures calculated. Figure 2 shows that after a number of sweeps the 100×100 multiple lattice's become thermalized, fluctuating around a specific value. Depending on how the lattice was initiliazed, all spins random (high temperature) or all spins up or down (low temperature). It is shown for four temperatures, two below the phase transition and two above. Where the phase transition is at [1] $T_c = 2.269$. It is seen that for temperatures close to the phase transition and especially after the system transitioned to another phase the system thermalizes after a

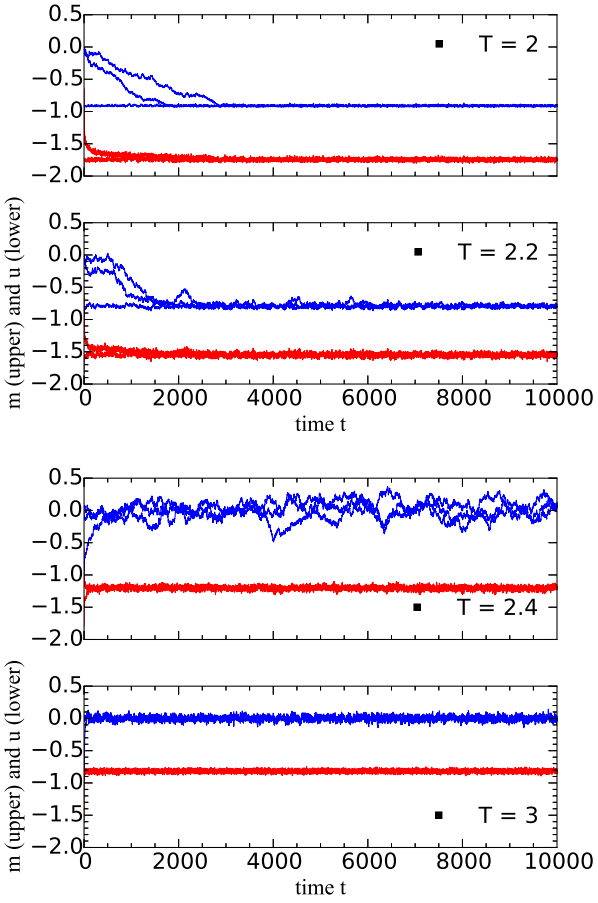

FIG. 2. Magnetization per spin(m, blue line) and internal energy ($u = E/N$, red line) plotted against the total number of sweeps made (time t). For the Metropolis algorithm of a 100×100 lattice initialised twice from higher temperature (random spins) and low temperature (all aligned, up or down) state. At four different temperatures. From which the thermalisation times are obtained.

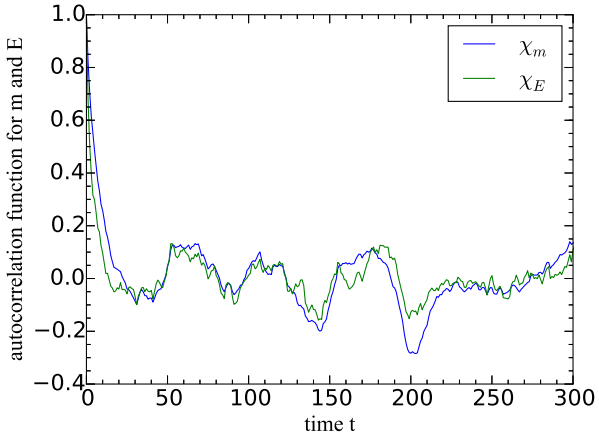

FIG. 3. The magnetization and energy autocorrelation function $\chi(t)$ for a 100×100 lattice shown at a temperature of $T = 2$. Calculated with the Metropolis algorithm after the system is thermalized. Both are normalized with the first element in the array. Time is measured in sweeps.

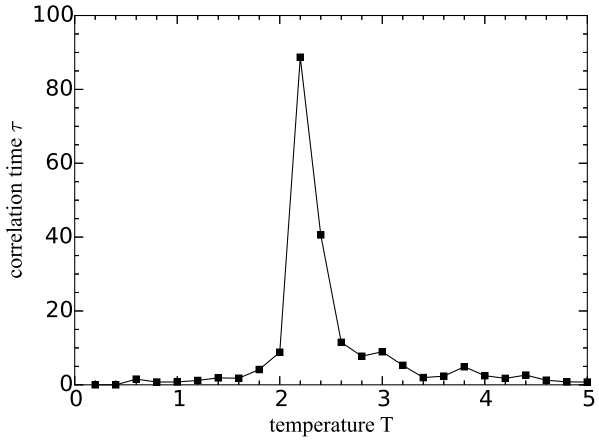

FIG. 4. Correlation times for a 100×100 lattice. For the Metropolis algorithm. Time is measured in sweeps. The straight line connecting the points are there as a guide for the eyes.

long while, or never thermalizes fully but keeps largely fluctuating. Which at $T = 2.4$ can be attributed to critical fluctuation. The thermalisation times found are always doubled just to be sure we are thermalized, due to the randomnes. E.g. for $T = 2$ the thermalisation time seen is roughly 4000 sweeps and therefore doubled to 8000 sweeps and for $T = 2.2$ roughly 8000 sweeps doubled to 16000 sweeps.

For finding the correlation times eq 14 is used after the system is thermalized, for the energy and the absolute value of the magnetization. In figure 3 the autocorrelation is shown for m and E at a temperature of 2. Both are normalized with the first element in their respective array. What is seen in all the autocorrelation figures made is that for the energy autocorrelation the measurements are earlier independent than the magnetization autocorrelation. What is also seen is a not understood wobble after reaching 0 once. Which in comparison with [2], where it is a smooth line after reaching 0 once, shows that something may be wrong in calculation of the autocorrelation values. Multiple things were tested to smooth the wobble but to no avail, letting the system thermalize longer and generating more data. For calculation of the correlation times the figure is replotted on a semi-log scale (y-axis log) and a least-squares fit is applied to the straight line portion of the plot. The slope gives one the correlation time, see eq 13. No errors were calculated on the fit because of the weird wobble slightly messing things up. What is done is that the correlation time is increased by a factor of 1.2 for every correlation time larger than one. Figure 4 the correlation times found for a 100×100 lattice are shown, in which one can clearly see the critical slowing down near the critical temperature.

Chosen for every square lattice size investigated is that

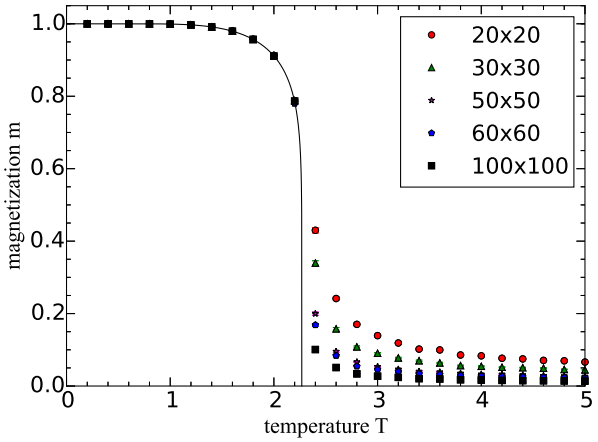

FIG. 5. The magnetization per spin of the two dimensional Ising model. With the solid line the theoretical calculation(infinite lattice) and the points measurements from the Metropolis algorithm for different lattice sizes. The errors are smaller than the points.

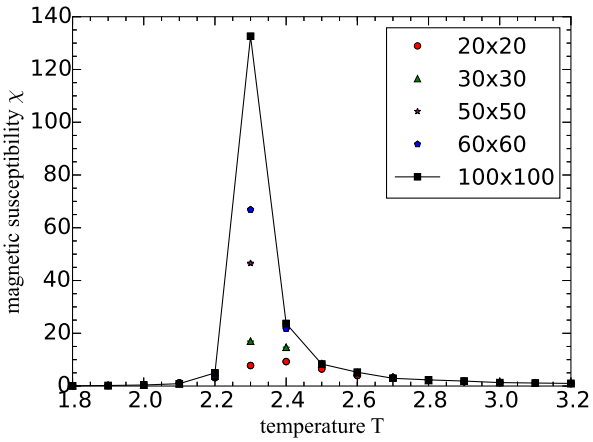

FIG. 6. Measurements of the magnetic susceptibility at different lattice sizes from the Metropolis algorithm of a two dimensional lattice. Where the 100×100 lattice has a line through it as a guide for the eyes. The errors are smaller than the points.

$n = 1000$ independent measurements are made. But more measurements are made, chosen a maximum of 10 dependent measurements are made between two independent measurements. The according simulation time in sweeps can be calculated using $t_{max} = 2\tau n$.

Figure 5/6 show the calculated magnetization per spin m and magnetic susceptibility χ from a system which had ample time to thermalize and measurements made in accordance with the found correlation times shown in figure 4. Where the deviation of the magnetization per spin from the exact solution of the two dimensional Ising model comes from the fact that we run the simulation on a finite lattice. Both show a clear sign of a phase transition near the critical temperature $T_c = 2.269$. Also seen for the magnetization is as the system size increases measurements correspond more to the theoretical calculations. The magnetic susceptiblity shows a higher and higher peak near the critical temperature, which will be infinity for a real system.

Error calculation for the magnetization is done by use of eq 16. Which is shown to be smaller than the points in figure 5. Error calculation for the magnetic susceptibility is obtained by bootstrapping 200 times, see eq 17. Which is also shown to be smaller than the points in figure 6, but I believe this is incorrect. Expected is that near the critical temperature the errors will be larger, because the values near the critical temperature are also large.

IV. WOLFF ALGORITHM

The Wolff algorithm is a so called cluster-flipping algorithm, it creates cluster which have their spins oriented in the same direction and then flip all the spins in the cluster in one go. The big advantage is that it has barely any critical slowing down. Therefore it will be used to obtain the critical exponents of the $2D$ and $3D$ Ising model, by usage of finite size scaling. The algorithm chooses a so called seed spin, random spin from the lattice, it looks at its nearest neighbours and if they are oriented in the same direction as the seed spin there is a probability P_{add} that it gets added to the cluster. Then it looks in turn at each of the nearest neighbours of that spin, and if they are pointing in the same direction as the seed spin again add it with a probability P_{add}. For each spin that was added do the same, this is repeated as many times as necessary until there are no spins left in the cluster whose neighbours have not been considered for inclusion in the cluster. Then the cluster is flipped.

It satisfies the condition of detailed balance (eq 9) by making the following chooses, see [2]

$$\frac{g(\mu \to \nu)}{g(\nu \to \mu)} = (1 - P_{add})^{m-n} \quad \text{and} \quad E_\nu - E_\mu = 2J(m-n) \tag{22}$$

With m the bonds that are broken from going from μ to ν gives an energy change of $2J$. With n the bonds which made, giving an energy change of $2J$. But I set $J = 1$. Substitute eq 22 into the condition of detailed balance and rearranging gives for the acceptance ratios

$$\frac{A(\mu \to \nu)}{A(\nu \to \mu)} = [e^{2\beta J}(1 - P_{add})]^{n-m} \quad \text{with} \quad P_{add} = 1 - e^{-2\beta J} \tag{23}$$

By choosing this value of P_{add} it makes the acceptance ratios equal to unity, every move is accepted while still satisfying detailed balance. Notice the temperature dependence of the algorithm is in this probability, for high temperature a small chance of creating large clusters for low temperatures a high chance. The algorithm also satisfies ergodicity, there is a finite chance at any move that

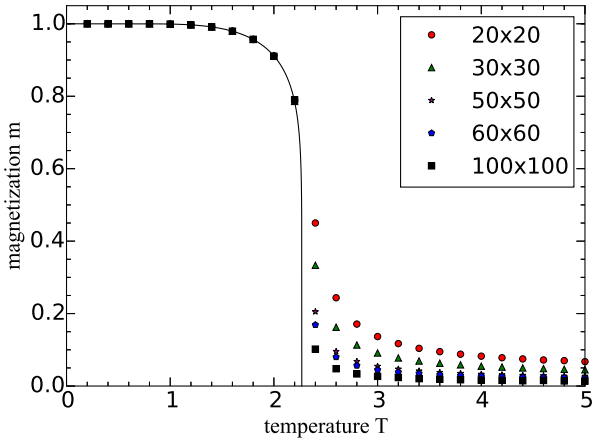

FIG. 7. The magnetization per spin of the two dimensional Ising model. With the solid line the theoretical calculation (infinite lattice) and the points measurements from the Wolff algorithm for different lattice sizes.

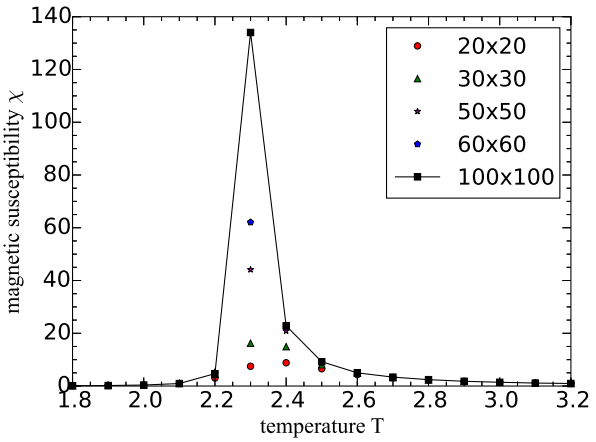

FIG. 8. Measurements of the magnetic susceptibility at different lattice sizes from the Wolff algorithm of a two dimensional lattice. Where the 100×100 lattice has a line through it as a guide for the eyes.

any spin will be chosen as the sole member of a cluster of one, which then gets flipped. The succesion of such moves will generate from any state any other state in a finite time. Together with detailed balance it guarantees that the multiple Markov Processes will generate the Boltzmann probabilities, after thermalization. The Wolff algorithm is representative of a real system.

In comparison with the Metropolis algorithm at temperatures lower and around the critical temperature the thermalization times are shorter and the correlation times are close to one. Both are due to that a cluster flips mostly flips a large amount of spins at once. Therefore this algorithm is used for obtaining the critical exponents. At higher temperature though it becomes a different story the algorithm almost becomes a single flip algorithm. Due to that the correlation time are expected to be close to one, and assumed it is not high for high temperatures thefore no calculation is made of the correlation time. To note is that after thermalization a large cluster can be flipped pushing the system shortly out of its thermalized value range, which can take a few steps to return to the thermalization equilibrium. Therefore a lot of measurements are made after thermalization to average out this noise factor. Depending on the lattice size a minimum of 100 000 measurements are made after thermalization.

Because of the error in the error calculation of the magnetic susceptiblity from the Metropolis algorithm no error calculation is applied here.

Shown in figure 7/8 is the magnetization per spin and the magnetic susceptiblity. Where the magnetization per spin is calculated every step by a direct sum over the spins on the lattice divided by the total spins on the lattice. From which the magnetic susceptibility can be calculated using eq 11. Both show a clear sign of a phase transition near the critical temperature $T_c = 2.269$. Again

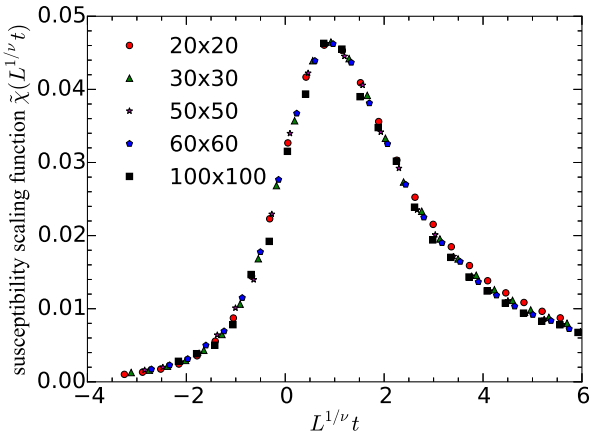

FIG. 9. Data collapse on the magnetic susceptibility from Monte Carlo measurements from the Wolff algorithm for the two dimensional Ising model. For five different system sizes. Values found for the critical exponents are: $\gamma = 1.76$, $\nu = 1.00$ and $T_c = 2.269$.

seen for the magnetization is as the system size increases measurements correspond more to the theoretical calculations. Again the magnetization susceptibility shows an increasing peak around the critical temperature. Both figures show a clear correspondence with the magnetization and magnetic susceptibility from the Metropolis algorithm (see fig 5/6)

By usage of the finite size scaling method the critical exponents for the two and three dimensional Ising model are retrieved, see eq 18. The critical exponents found for the two dimensional case, see fig 9, are: $\gamma = 1.76 \pm 0.02$, $\nu = 1.00 \pm 0.02$ and $T_c = 2.269 \pm 0.005$. Which are in good agreement with the exact known values, $\gamma = 7/4$, $\nu = 1$ and $T_c = 2.269$.

The critical exponents found for the three dimensional

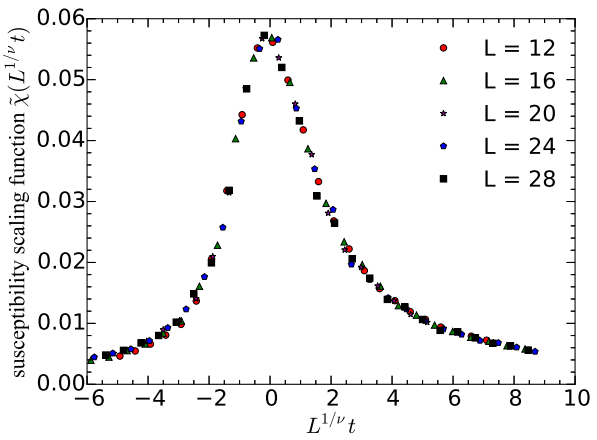

FIG. 10. Data collapse on the magnetic susceptibility from Monte Carlo measurements from the Wolff algorithm for the three dimensional Ising model. For five different system sizes, with L being the length of one axis of the lattice, other two axis are the same value. Values found for the critical exponents are: $\gamma = 1.26$, $\nu = 0.63$ and $T_c = 4.51$.

case, see fig 10, are: $\gamma = 1.26 \pm 0.01$, $\nu = 0.63 \pm 0.01$ and $T_c = 4.51 \pm 0.005$. Which are close to the values from other Monte Carlo simulation (see [8] and [9]), $\gamma = 1.237$, $\nu = 0.6301$ and $T_c = 4.5115/4.5103$. Where the small deviation from my collapse comes from that the simulation is run on small lattice's. Simulation of lattice larger than $L = 28$ would take more than 8 hours. While already at $L = 24$ and $L = 28$ noise is visible, some clear points deviate from the line which is not due to the wrong critical exponents. Also seen for measurements of the two dimensional 100×100 lattice. Implying that even more measurements need to be made pushing the CPU time even higher.

V. CONCLUSION

I have presented the theory behind Monte Carlo simulations and shown two sort of algorithms. A single flip algorithm, the Metropolis algorithm, and a cluster flip algorithm, the Wolff algorithm. Where both the magnetization and magnetic susceptibility show a clear sign of a phase transition. In correspondence with the theory, where the deviation comes from the fact that the measurements come from a finite lattice. Also shown with the Wolff algorithm is that the critical exponents in the two dimensional case are in good agreement with the theory and in three dimensions close to other Monte Carlo simulations.

[1] L. Onsager, Physical Review **65**, 117 (1944).
[2] M. E. Newman, G. T. Barkema, and M. Newman, *Monte Carlo methods in statistical physics*, Vol. 13 (Clarendon Press Oxford, 1999).
[3] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, The journal of chemical physics **21**, 1087 (1953).
[4] U. Wolff, Physical Review Letters **62**, 361 (1989).
[5] J. W. Gibbs, *Elementary principles in statistical mechanics* (Courier Corporation, 2014).
[6] B. Efron, SIAM review **21**, 460 (1979).
[7] M. Matsumoto and T. Nishimura, ACM Transactions on Modeling and Computer Simulation (TOMACS) **8**, 3 (1998).
[8] H. Blote, E. Luijten, and J. R. Heringa, Journal of Physics A: Mathematical and General **28**, 6289 (1995).
[9] T. Preis, P. Virnau, W. Paul, and J. J. Schneider, Journal of Computational Physics **228**, 4468 (2009).