

AP3082 Project 2: Ising Model

Kah Jen Wo¹ and Juan Torres¹

¹*Faculty of Applied Sciences, Delft University of Technology, Delft, The Netherlands*
(Dated: May 16, 2021)

Two different algorithms have been employed in studying the 2D square lattice Ising model numerically. The two algorithms used in this study are the Metropolis algorithm and the Wolff algorithm. We verify the validity of our simulation results by comparing the numerical observables obtained with the analytical behaviour. We assess the difference in the observables produced by each of the algorithm by means of investigating the errors produced by each of the algorithms.

I. INTRODUCTION

The Ising model is a nearest-neighbour spin interaction model used to describe ferromagnetism in metals by using discrete binary variables and it has been used to study phase transitions in finite substances such as iron and nickel where ferromagnetism spontaneously arise when cooled below a certain critical temperature, i.e. the Curie temperature.

The model was first introduced by Ernst Ising in an attempt to describe phase transition in ferromagnetic substances [1]. His attempt has proved that ferromagnets in 1D disappointingly do not exhibit phase transitions. Since then, many have made an effort into obtaining a more accurate understanding of the model. In 1943, it was then shown analytically by Lars Onsager that phase transition in 2D Ising model is indeed possible [2]. The Ising model is now being utilised in various fields such as machine learning to extract insights from complex physical systems [3].

In statistical mechanics, the properties of a time-dependent evolving many-body system are calculated as an average over an ensemble of the system under certain constraints. It requires several approximations such as infinitely large domain to obtain insights into the physics of the system. Nevertheless, these insights have proved invaluable in the scientific community's attempt to understand the Ising model in detail. For example, Ising model has provided an accurate description of FeCO₃ and FeCl₂, which are anisotropic crystals [4].

Unfortunately, these approximations do not apply in practice because systems with infinite domain do not exist. This is the reason why computational approach is needed: to recover the properties of the system from the statistical behaviour of an arbitrarily sized simulation domain. Numerical approaches, however, still present their own set of challenges such as computational efficiency, accuracy and limitations due to lack of computational resources.

In a classical Ising model, the computational approach to simulate the system is by means of *Monte-Carlo* procedures. Monte Carlo simulations often see a Metropolis-type update mechanism, which is also known as Metropolis algorithm [5]. This algorithm involves flipping single spins at each iteration and is considered simple. More recently, cluster algorithms were invented, such as the one

by Ulli Wolff, which is now known as the Wolff algorithm [6]. There are more modern and advanced algorithms which exist today for the purpose of simulating Ising model systems. However, in this paper, we will study the Ising model using only the Metropolis and Wolff algorithm.

We begin with the review of relevant background theory in Ising model in section II which is used in our study of the classical 2D Ising model. In section III, we discuss the Metropolis and Wolff algorithm and provide the pseudocode for each of these algorithms. In section IV, we discuss the important criteria which our simulation should meet. In section V, we present our results from the simulations from both algorithms as well as their corresponding discussions. Finally, in section VI, we provide the conclusion of our study.

II. BACKGROUND THEORY FOR ISING MODEL

In this section we review the relevant theories in Ising model for our simulations.

A. Nearest neighbour Hamiltonian

Consider a ferromagnetic 2D square lattice Ising model, where the i^{th} spin in the lattice is denoted by S_i and each of them can either spin down or spin up along the z -axis, i.e. $S_i \in \{-1, 1\}$. Its Hamiltonian is modelled as such

$$H = -J \sum_{\langle i, j \rangle} S_i S_j - h \sum_{i=1}^N S_i \quad (1)$$

where J is a positive ferromagnetic coupling constant, $\langle i, j \rangle$ denotes sum over all nearest neighbour pairs without double counting, h is the strength of the external magnetic field along the z -axis, and $N = L \times L$ is the total number of sites on the square lattice.

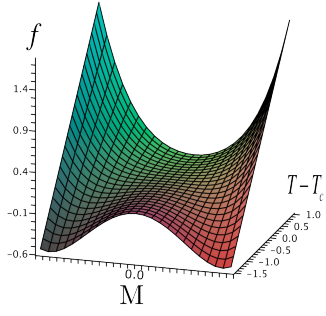


FIG. 1. Free energy (denoted by f in the vertical axis) of a Ising model system for varying magnetisation M and temperature T . The external magnetic field is zero. Source: Adapted from [8].

B. Phase transition

In Ising model there are two distinct phases: *paramagnetic* phase and *ferromagnetic* phase. In paramagnetic phase, the spins take on random direction and are disordered due to high thermal fluctuations, i.e. average magnetisation of zero. In ferromagnetic phase, the spins are more likely to align with a certain direction. In this section, we review the critical temperature T_C , below which the Ising model system takes on the ferromagnetic phase and above which takes on the paramagnetic phase.

The phase transition of the system can be studied using the physical observables shown in section II C. The most common observable used to determine the critical temperature of the system is the magnetisation, where its expression is given in Eqn. 3. The average magnetisation is used to represent the *order parameter*. In mean field theory, the critical temperature T_C for the 2D ising model was estimated to be $4J/k_B$. However, in mean field theory we assume that fluctuations are small and thus it tends to overestimates the value of T_C .

Fortunately, the exact analytical expression for the magnetisation of the 2D square lattice Ising model for $T < T_C$ has been obtained by Kramers and Wannier [7]:

$$M = (1 - \sinh^{-4}(2\beta J))^{1/8} \quad (2)$$

Solving for $M = 0$ in Eqn. 2 yields the exact critical temperature, which we will use in our simulation. That is $T_C = \frac{2J}{k_B \ln(1+\sqrt{2})} \approx 2.27J/k_B$.

In Fig. 1, we see the free energy diagram of the Ising model system for varying temperature and magnetisation when there is no external magnetic field. From the temperature axis, we see that when $T = T_C$, the free energy transitions from having two minimas to having one minima. The figure also shows that in the ferromagnetic phase ($T < T_C$), the spins prefer either a spin down or spin up configuration while in the paramagnetic phase ($T > T_C$), the spins do not have a preference for any configuration, leading to zero average magnetisation.

C. Physical observables

The observable quantities are imperative in aiding the study of thermodynamic properties in the Ising model. One of these quantities is the average *magnetisation*

$$\langle M \rangle = \frac{1}{N} \sum_{i=1}^N \langle S_i \rangle \quad (3)$$

The specific heat per spin

$$c = \frac{k_B \beta^2}{N} (\langle E^2 \rangle - \langle E \rangle^2) \quad (4)$$

The susceptibility per spin

$$\chi = \beta N (\langle M^2 \rangle - \langle M \rangle^2) \quad (5)$$

where $\beta = \frac{1}{k_B T}$, k_B is the Boltzmann constant and T is the temperature.

III. MONTE CARLO ALGORITHMS

In this section we review two different algorithms that we used in our study of the 2D square lattice Ising model.

A. Metropolis algorithm

It is worth noting that even though the Metropolis algorithm seems relatively simple, it has been met with considerable amount of success when it comes to generating data of the physical observables in systems that exhibit Ising-like model, e.g. model mono-layer of nanographene with the presence of an external magnetic field [9]. Below there is the pseudo-code for a single iteration of Metropolis algorithm.

Algorithm 1 Metropolis

- 1: Calculate E_{current} .
 - 2: Choose a random spin to flip with uniform distribution.
 - 3: Calculate E_{trial} .
 - 4: Calculate $\Delta E = E_{\text{current}} - E_{\text{trial}}$.
 - 5: **if** $\Delta E \leq 0$ **then**
 - 6: Set $E_{\text{current}} = E_{\text{trial}}$.
 - 7: **else**
 - 8: Choose a random number $r \in (0, 1)$ with uniform distribution.
 - 9: Compute $W = e^{-\frac{\Delta E}{k_B T}}$.
 - 10: **if** $r < W$ **then**
 - 11: Set $E_{\text{current}} = E_{\text{trial}}$.
 - 12: **else**
 - 13: Do nothing. (i.e. the lattice stays the same)
 - 14: **end if**
 - 15: **end if**
-

B. Wolff algorithm

Instead of flipping a single spin at each iteration as shown in the Metropolis algorithm, the Wolff algorithm flips an entire *cluster*. It should also be noted that the Wolff algorithm does not take external magnetic field into account, so we can only simulate Ising model with $h = 0$. Below, we find the pseudo-code for **one iteration** in the Wolff algorithm.

Algorithm 2 Wolff

- 1: Choose a random spin as the seed with uniform distribution.
 - 2: Check the neighbours of the seed spin.
 - 3: If they have the same direction as that spin, add them to the cluster with probability $P_{\text{add}} = 1 - e^{-\frac{2J}{k_B T}}$.
 - 4: For each added spin from the previous step, repeat step 3 until there are no spins left in the cluster whose neighbours have not been considered for inclusion in the cluster.
 - 5: Flip the cluster.
-

IV. ERGODICITY AND DETAILED BALANCE

It is imperative that the Monte Carlo algorithms satisfy the following criteria [10]:

1. It should be possible for the state of the system from a configuration a lead to a configuration b such that all configurations in phase space are reached if we run it for long enough. This is a must to generate states for correct Boltzmann probabilities. (**Ergodicity**)
2. For a long enough simulation, the probability distribution that the equilibrated system generates must be the Boltzmann probability distribution. (**Detailed balance**)

It is interesting to note that for local algorithms such as Metropolis algorithm (given in section III A), they fulfil the criteria for the most part in Ising model but fail when the system is near the criticality regime. This is due to the algorithm implementing too small of a change in the magnetisation in one iteration while there is a wide distribution of magnetisation, i.e. there are regions of the same spin which are difficult for the Metropolis algorithm to flip. Hence, cluster algorithms such as the Wolff algorithm was developed as a solution to this problem, which is given in section III B.

V. RESULTS AND DISCUSSION

To study physical quantities from a simulation with a statistical nature, a systematic study of different system parameters is given. First, a study of the convergence of the system towards the expected behaviour as a function of system size and number of relaxation steps is presented

for both algorithms. Once a criteria for significant results is established, a derivation of three physical quantities is presented, i.e. magnetisation, specific heat and magnetic susceptibility. The three critical exponents associated to these quantities are presented. Similarly, the full range of parameters is studied in a phase diagram. Finally, a discussion about the performance of the code is presented.

A. Error calculation

Two kinds of errors will be discussed. On one hand, errors obtained by comparison with exact analytical expression derived for the 2D Ising model [2]. On the other hand, statistical errors induced by correlations in the data sets due to the nature of the algorithms used. To describe the statistical errors, the autocorrelation function of a given data set is calculated, and the correlation time τ is extracted from it. The autocorrelation function of observable A is given as,

$$\chi_A(t) = \frac{(N-t) \sum_n A_n A_{n+t} - \sum_n A_n \sum_n A_{n+t}}{\sigma_n \sigma_{n+t}}, \quad (6)$$

where $\sigma_n = \sqrt{(N-t) \sum_n A_n^2 - (\sum_n A_n)^2}$, and N is the length of the data set. Errors are calculated using the block data method [10]. First, the data set is divided into uncorrelated groups separated by a distance τ . Each block is then converted into a single value that corresponds to the mean of the block, and its error corresponds to the standard deviation of the data inside the block. In Fig. 2 one can observe that the correlation length increases with the system size for the Metropolis algorithm. That is, correlation increases with system size. On the other hand, for the Wolff algorithm it always remains much smaller. From these results, one can expect a priori to have smaller errors for the Wolff algorithm.

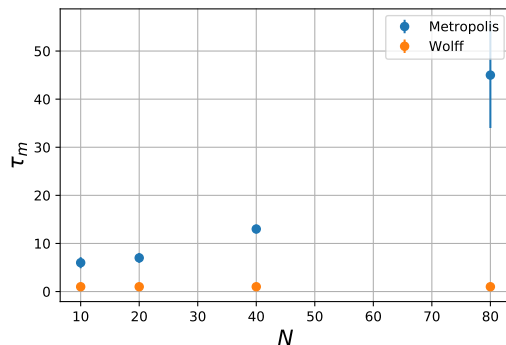


FIG. 2. Correlation time versus system size for the Metropolis and Wolff algorithms. These results correspond to the ones shown in Fig. 3.

B. System size dependence

An exact solution for the Ising model was derived by Onsager [2] for the case of an infinite system. In a finite system, we converge towards the exact result as the system size increases. In Fig. 3 we can observe the behaviour of the magnetisation as a function of the temperature for different system sizes.

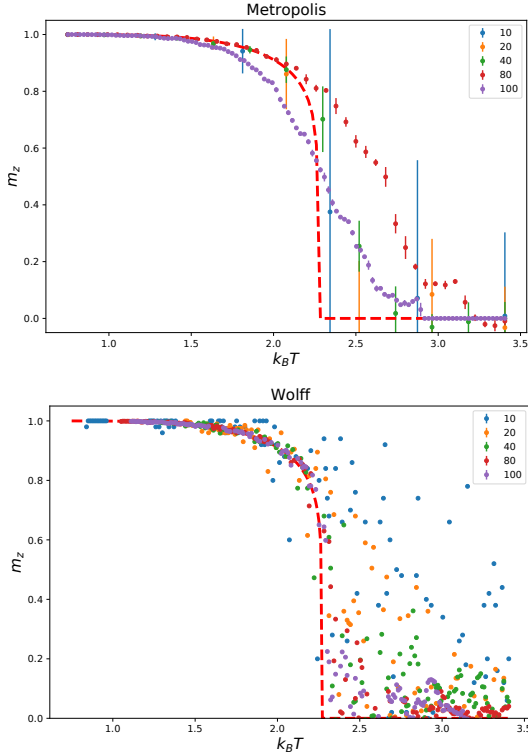


FIG. 3. Magnetisation with respect to temperature. Here $k_B = 1$, $J = 1$ and $h = 0$. Red dashed curve corresponds to exact analytical result for an infinite system. Different curves correspond to different system sizes, i.e. N . Both cases were evaluated for 265 points. Data was further decreased by data blocking error calculation. (Top): Metropolis algorithm with 6000 relaxation steps. (Bottom): Wolff algorithm with 10 relaxation steps. Error is too small to be observed.

In the top panel, the calculation was performed using the Metropolis algorithm. One can observe that there are significant differences between the exact case and the simulation. For small lattices, the errors around the phase transition are very large with respect to the distance between points. As the system size increases, the errors decrease. However, the relaxation towards zero magnetisation is not as expected.

In the lower panel, we can observe that for small lattices, the data is non-uniformly distributed. As the system size increases, the transition follows the exact behaviour. If one looks closely to the case $N = 100$ (purple), one finds that it behaves very similar to the exact case around the critical temperature.

C. Relaxation steps

In Monte Carlo simulations, the state of the system at a given time step depends on the state at the previous step. When there is a change in the parameters, e.g. temperature, the system requires time to reach equilibrium. Otherwise, the new system configuration will be mostly given by the previous configuration rather than the change in parameters. Here, the relaxation time is studied. In Fig. 4 one can observe the magnetisation with respect to temperature for different relaxation times calculated using the Metropolis algorithm. For the lowest case (blue curve), the data is separated the most from the exact result. As the relaxation time increases, the data moves towards a different curve as can be observed from the high overlap of the other data points. Nevertheless, the Metropolis algorithm does not reach the exact prediction. After this simulation, we consider than relaxation times larger than 3000 steps can produce reliable results.

Since the Metropolis algorithm is based on changing a single spin at a time, it will require longer relaxation times. On the contrary, the Wolff algorithm that flips many spins at a single time requires short relaxation times. Also, it would be very costly computationally to have relaxation times comparable to the ones used in Metropolis since the number of operations involved is different.

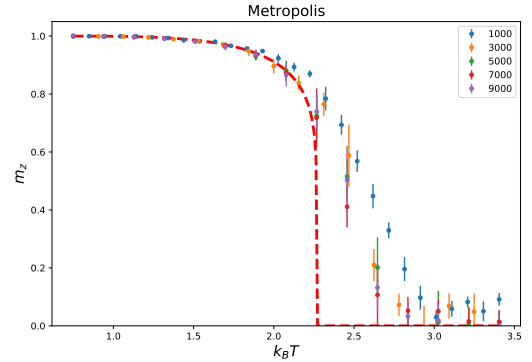


FIG. 4. Magnetisation with respect to temperature. Here $k_B = 1$, $J = 1$ and $h = 0$. Red dashed curve corresponds to exact analytical result for an infinite system. Different curves correspond to different relaxation times. Both cases were evaluated for 265 points. Data was further decreased by data blocking error calculation.

D. Physical quantities

Physical quantities are not-analytical around the critical temperature. For the quantities discussed here, a power law is followed in this region. The specific heat and the magnetic susceptibility were calculated according to Eqn. 4 and Eqn. 5, respectively. The results for each one as a function of temperature for different system

sizes are presented in Fig. 5 and Fig. 6, respectively.

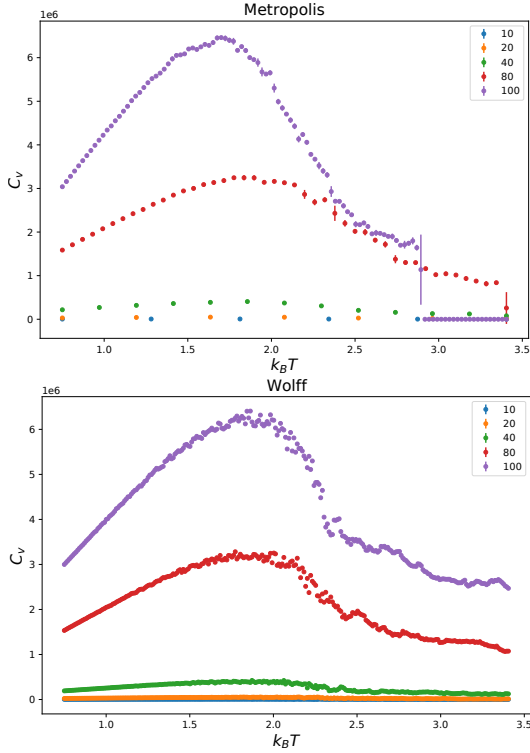


FIG. 5. Specific heat with respect to temperature for different system sizes. Here $k_B = 1$, $J = 1$ and $h = 0$. Different curves correspond to different system sizes, i.e. N . Both cases were evaluated for 265 points. Data was further decreased by data blocking error calculation. (Top): Metropolis algorithm with 6000 relaxation steps. (Bottom): Wolff algorithm with 10 relaxation steps. Error is too small to be observed.

For the specific heat, one can observe that the shape is preserved for both algorithms. The height of the divergence at the critical temperature depends on the system size. For the Wolff algorithm (lower panel), there are many more data points than for the Metropolis algorithm (upper panel). Also, data generated by Metropolis overlaps for different system sizes. This does not occur for the Wolff algorithm.

For the magnetic susceptibility, one can observe significant differences when comparing the data generated by Metropolis and Wolff algorithm. First, for the Wolff algorithm (lower panel) one observes that the position of the divergence is the same regarding the system size. On the contrary, for the Metropolis algorithm (upper panel) the divergence is centred at different temperatures for different system sizes. As expected, for larger sizes, the divergence goes towards the critical temperature.

One can observe that for the Metropolis algorithm, the data sharply goes to zero after reaching the critical temperature. However, this range is extended inversely proportional to the system size. For the Wolff algorithm, this feature is not observed. Instead, one finds a smooth decay from the divergence.

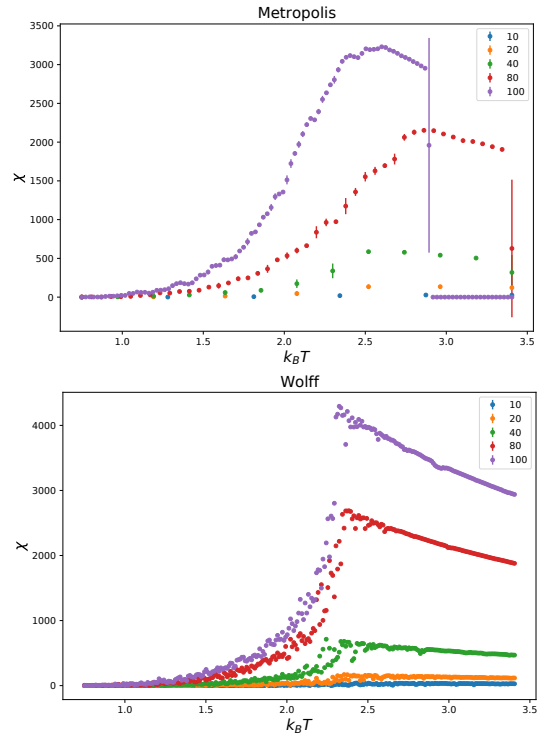


FIG. 6. Magnetic susceptibility with respect to temperature for different system sizes. Same parameter configuration as in previous results is used here.

E. Critical exponents

To model the non-analytic behaviour of the physical quantities around the critical temperature, we fitted our simulated results with a power law decay in order to extract the critical exponents of the system. These results can be compared with the exact analytical results [2].

Exponent	Exact	Metropolis	Wolff	Error
α	0	2.813 ± 0.02	2.324 ± 0.01	
β	$1/8 \approx 0.125$	0.404 ± 0.003	0.394 ± 0.001	30.94%
γ	$7/4 \approx 1.75$	1.522 ± 0.003	1.251 ± 0.006	14.98%

TABLE I. Critical exponents for the specific heat(α), magnetisation(β) and magnetic susceptibility(γ) taken from results shown in Fig. 5, 3, and 6, respectively. The case with the largest lattice ($N = 100$) was fitted to a power law decaying function given by $f(t, l) = t^{-l}$ where t is the reduced temperature, and l is the critical exponent to be fitted. For the magnetisation, $l, t \rightarrow -l, -t$ in $f(t, l)$. Errors in the right column are calculated with respect to Metropolis results.

In Table I the results of the fitted critical exponents are presented. One can observe that two of them have an error that is below $\sim 31\%$. The exponent for the specific heat is quite far from the theory prediction.

F. Phase diagram

Using Metropolis algorithm, we began from a lattice with randomly aligned spins and slowly raised (i.e. relaxation steps of 2000) the temperature T/T_C from 0 to a target value while also sweeping through the possible values for the external magnetic field $h \in (-1, 1)$. The results of this simulation sweep can be seen in Fig. 7. Since the result was generated using the Metropolis algorithm, we expect the magnetisation to not drop to zero at exactly $T/T_C = 1$ but slightly higher, which is indeed what we observe.

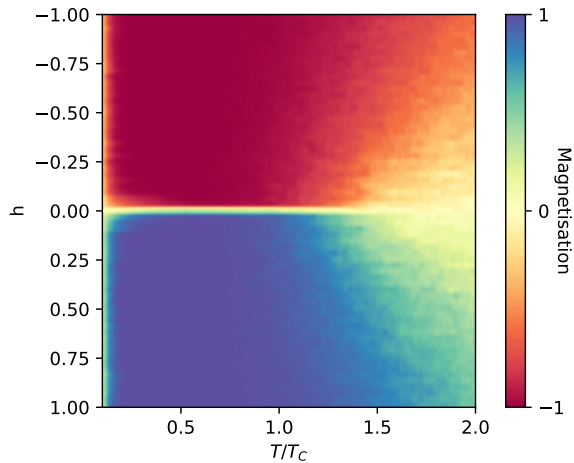


FIG. 7. Phase diagram of the 2D square lattice Ising model using the Metropolis algorithm. The parameters are: $L = 50$, $J = 1$ and relaxation steps = 2000.

Unfortunately, the Wolff algorithm which we implemented does not support varying external magnetic fields, thus there is no corresponding result generated by the Wolff algorithm.

G. Performance

In this section we study and compare the performance of the Metropolis algorithm and Wolff algorithm. Since the way that each of the algorithm work is fundamentally different (i.e. the former flips a single spin while the latter flips an entire cluster in a single iteration), it is crucial that a proper benchmarking procedure is designed so that we can compare them and retrieve relevant insights.

To study the performance of the two algorithms, we measured the time needed to run them starting from a completely positive spin lattice until the absolute average magnetisation of the corresponding systems reaches a threshold value, which we define as $\varepsilon = 0.1$. We repeat this 10 times for each value of temperature T/T_C , which we chose to range from 1.2 to 2.0 with 20 data points. Then, we calculate the time average of each of the 10 runs and it is denoted by \bar{t} . The result can be obtained

by running `benchmark.py` and its plot can be seen in Fig. 8.

We chose this specific T/T_C range in order to accentuate the difference in performance between the Metropolis algorithm and the Wolff algorithm. From Fig. 8, we indeed see that the Metropolis algorithm requires significantly more time to reach an absolute average magnetisation of 0.1 starting from a completely positive spin lattice compared to the Wolff algorithm. On the other hand, the average run-time \bar{t} for the Wolff algorithm seems to remain constant at ~ 0.2 s with slight fluctuations. This implies that the fact that we are near the criticality regime does not impact its efficiency. The discrepancy in performance between the two algorithms can be explained by section IV.

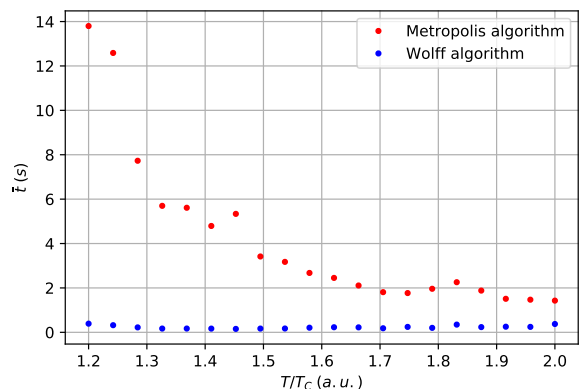


FIG. 8. Average run-time \bar{t} for Metropolis algorithm (red) and Wolff algorithm (blue) for the system to reach an absolute average magnetisation of 0.1 beginning from a completely positive spin lattice for varying T/T_C . The system size used is $L = 64$. The results were obtained on a machine with an Intel i5-7200U CPU.

VI. CONCLUSION

To conclude, we have studied the system size dependence in section VB, where we see that there is an inherent need for the simulation domain to be large such that the physical observables obtained can approach those of analytical results. Then in section VC, we showed that ability for the Metropolis algorithm to match the analytical result seems to plateau at high relaxation times, while in section VB the Wolff algorithm achieved close proximity to the analytical data. We studied the phase diagram and found expected behaviour in section VF.

We also accessed the physical quantities and the critical exponents in sections VD and VE, respectively. We found the expected general behaviour for the physical quantities for varying simulation parameters. However, the general shape of the plots of the specific heat capacity and the magnetic susceptibility seems to deviate from the literature. Furthermore, we found that the critical exponents that we retrieved from the simulations also deviates

from the analytical results by a considerable amount.

From the performance viewpoint, We benchmark-ed our implementation of both the algorithms in section **V G** and found that Wolff algorithm is highly preferred over the Metropolis algorithm especially in the vicinity of the criticality regime.

Finally, despite the considerable deviation of some of the simulation results we obtained, it is no doubt that development in computational physics in simulating ferromagnetism had come a long way and it has become an invaluable tool in studying this phenomena.

VII. ACKNOWLEDGEMENTS

This work is part of the course AP3082 Computational Physics supported by the Delft University of Technology.

Appendix A: Source code

The source code can be found in this Gitlab repository: [Gitlab link](#).

-
- [1] E. Ising, *Zeitschrift für Physik* **31**, 253 (1925).
 - [2] L. Onsager, *Phys. Rev.* **65**, 117 (1944).
 - [3] N. Walker, K.-M. Tam, and M. Jarrell, *Scientific reports* **10**, 1 (2020).
 - [4] B. Diu, *Eléments de physique statistique* (Hermann, Paris, 1989).
 - [5] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *The Journal of Chemical Physics* **21**, 1087 (1953).
 - [6] U. Wolff, *Phys. Rev. Lett.* **62**, 361 (1989).
 - [7] H. A. Kramers and G. H. Wannier, *Phys. Rev.* **60**, 252 (1941).
 - [8] Mean field theory of phase transformations, in *Phase-Field Methods in Materials Science and Engineering* (John Wiley & Sons, Ltd, 2010) Chap. 2, pp. 9–25.
 - [9] V. Ivashko, O. Angelsky, and P. Maksimyak, *Journal of Magnetism and Magnetic Materials* **492**, 165617 (2019).
 - [10] M. E. J. Newman and G. T. Barkema, *Monte Carlo methods in statistical physics* (Clarendon Press Oxford University Press, Oxford New York, 1999).