The critical temperature of the two-dimensional Ising model from Markov chain Monte Carlo

Viktor Braaten Gulbrandsen, Katchi Senthil Ganesh
(Dated: November 18, 2026)

The two-dimensional Ising model is one of few models of a phase transition with an exact solution, which makes it a natural test of numerical methods. We simulate it on square lattices with periodic boundary conditions using the Metropolis algorithm. For a 2×2 lattice the estimated mean energy and magnetisation agree with the exact values to a relative error of 10^{-4}, and the heat capacity and susceptibility to within 2 %. Running the chains for different temperatures in parallel gives a speed-up of 6.4 on twelve threads. From the maxima of the heat capacity and the susceptibility for lattice sizes $L = 40$ to 100, and the finite-size scaling relation $T_c(L) - T_c(\infty) \propto 1/L$, we estimate the critical temperature of the infinite lattice as $T_c = 2.2667 \pm 0.0005\, J/k_B$ and $2.2715 \pm 0.0016\, J/k_B$. Both lie within 0.11 % of the exact value $2.2692\, J/k_B$.

Code is available at https://github.uio.no/viktorgu/fys3150, in the folder `project4/`.

I. INTRODUCTION

Late in the nineteenth century, the soon to be Nobel laureate Pierre Curie found a behavior of magnetic metals such as iron which needed explaining. While the metal maintained its magnetization, increasing the temperature caused this magnetization to disappear entirely at a certain point [1]. This specific point of transition was later coined the Curie temperature. In the following decades, starting with Pierre Weiss in 1907 conceptualizing the magnet as a sum of many smaller magnets feeling the average pull of all others [1], the question as to how to exactly describe this transition from magnetized to unmagnetized remained open. It was not until the 1920s when Wilhelm Lenz and his student Ernst Ising at the University of Hamburg properly formalized the framework of a grid of tiny magnets each only pointing up or down and feeling only their nearest neighbours (the Ising model), that this was given a clear enough form for the later Lars Onsager to solve exactly in two dimensions in 1944 [2, 3].

It was, however, not until the 1960s that the applicability and significance of such a simple model was fully appreciated [4]. This was partly because the model could not be solved exactly in three dimensions. With the advent of modern computers, it was instead approached numerically, with the Metropolis algorithm among the first such methods [5]. In the 1960s, measurements on argon and oxygen near their critical points showed behaviour consistent with the Ising model [4].

The two-dimensional Ising model is one of very few models of a phase transition with an exact solution. This makes it a natural test case: a numerical method that reproduces Onsager's critical temperature from finite lattices can be trusted on related models that have no exact solution, such as the Ising model in three dimensions. In this report we implement the Metropolis algorithm for the two-dimensional model, validate it against the exact results for a 2×2 lattice, and use simulations of lattices with up to 10^4 spins to estimate the critical temperature of the infinite lattice.

Section II presents the model and the Monte Carlo method. Section III presents the validation, the equilibration of the chains, the energy distribution, the performance of the parallel code, and the estimate of T_c. Section IV summarises the results. The exact results for $L = 2$ are derived in Appendix A.

II. METHODS

A. The Ising model

We consider a square lattice of $N = L^2$ spins $s_{i,j} \in \{-1, +1\}$, $i, j = 1, \ldots, L$, for example

$$s = \begin{pmatrix} +1 & -1 & +1 \\ +1 & +1 & -1 \\ -1 & +1 & +1 \end{pmatrix}$$

for $L = 3$. Each spin interacts with its four nearest neighbours,

$$N(i,j) = (s_{i+1,j},\ s_{i-1,j},\ s_{i,j+1},\ s_{i,j-1}), \tag{1}$$

where indices are taken modulo L (periodic boundary conditions, Fig. 1). Every spin then has exactly four neighbours, and the lattice has no edges. Without an external magnetic field the energy of a microstate s is

$$E(s) = -\frac{J}{2} \sum_{i,j=1}^{L} \sum_{s_{k,l} \in N(i,j)} s_{i,j}\, s_{k,l}, \tag{2}$$

where $J > 0$ is the coupling constant. The factor $\frac{1}{2}$ compensates for every pair of neighbours appearing twice in the double sum. For $L = 2$ the left and right neighbours of a spin are the same spin, and likewise up and down, so each pair is connected through two bonds. We keep this convention, so that Eq. (2) applies unchanged for all L. For the lattice $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$ it gives $E = -2J(ab + ac + bd + cd)$ (Appendix A).

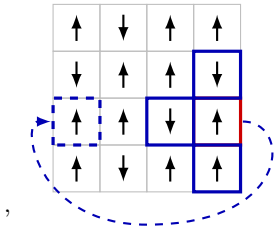

FIG. 1. Spin lattice with periodic boundary conditions. The neighbours of a spin on the edge are found on the opposite edge.

The magnetisation of a microstate is

$$M(s) = \sum_{i,j} s_{i,j} = N_+ - N_-, \qquad m(s) = \frac{M(s)}{N}, \quad (3)$$

where N_+ and N_- are the numbers of spins equal to $+1$ and -1. We also use the energy per spin, $\epsilon(s) = E(s)/N$.

We set $J = k_B = 1$. Energies are then given in units of J, temperatures in J/k_B, heat capacities in k_B and susceptibilities in J^{-1}.

B. Statistical mechanics

In thermal equilibrium at temperature T, the probability of a microstate $s \in \mathcal{S} = \{-1, +1\}^N$ is given by the Boltzmann distribution,

$$P_T(s) = \frac{1}{Z(T)}\, e^{-\beta E(s)}, \qquad Z(T) = \sum_{s' \in \mathcal{S}} e^{-\beta E(s')}, \quad (4)$$

where $\beta = 1/(k_B T)$, and the expectation value of a quantity A is $\langle A \rangle = \sum_{s \in \mathcal{S}} A(s)\, P_T(s)$.

Flipping every spin leaves E unchanged and changes the sign of M, so $\langle m \rangle = 0$ at every temperature on a finite lattice. As order parameter we therefore use the mean absolute magnetisation per spin, $\langle |m| \rangle$.

We further compute the specific heat capacity and the magnetic susceptibility per spin,

$$C_V = \frac{1}{N k_B T^2} \left(\langle E^2 \rangle - \langle E \rangle^2 \right), \quad (5)$$

$$\chi = \frac{1}{N k_B T} \left(\langle M^2 \rangle - \langle |M| \rangle^2 \right). \quad (6)$$

Equation (5) follows from $C_V = \frac{1}{N}\, \partial \langle E \rangle / \partial T$ (Appendix A), and $|M|$ replaces M in Eq. (6) for the same reason as above. In the thermodynamic limit both quantities diverge at the critical temperature T_c. On a finite lattice they have a finite maximum, which we use to locate the transition.

The sums over $\mathcal{S}$ contain 2^N terms. For $L = 2$ they can be evaluated by hand (Appendix A). For $L = 40$ there are about 10^{481} terms, and the expectation values must instead be estimated by sampling.

Algorithm 1 One Monte Carlo cycle at temperature T.

```
for n = 1, ..., N do
    Draw a site (i, j) uniformly at random
    Compute ΔE from Eq. (9)
    Draw r ~ U(0, 1)
    if ΔE ≤ 0 or r < e^{−βΔE} then
        s_{i,j} ← −s_{i,j}
        E ← E + ΔE,    M ← M + 2 s_{i,j}
Add E, E^2, |M|, M^2 to the running sums
```

C. The Metropolis algorithm

We sample from P_T with a Markov chain, a sequence of states in which the transition probability $W(s \to s')$ depends only on the current state s. The chain has P_T as a stationary distribution if it satisfies detailed balance,

$$P_T(s)\, W(s \to s') = P_T(s')\, W(s' \to s), \quad (7)$$

and converges to it from any initial state if it is also ergodic [5].

Each transition consists of a proposal and an acceptance step, $W(s \to s') = g(s \to s')\, A(s \to s')$. We propose to flip one spin, chosen uniformly among the N spins, so that $g(s \to s') = g(s' \to s) = 1/N$. For a symmetric proposal, detailed balance holds with the Metropolis acceptance probability

$$A(s \to s') = \min\left(1, \frac{P_T(s')}{P_T(s)}\right) = \min\left(1, e^{-\beta \Delta E}\right), \quad (8)$$

where $\Delta E = E(s') - E(s)$. The partition function does not enter. Since every state can be reached through single spin flips, the chain is ergodic.

A flip of $s_{i,j}$ changes only the four bonds to its neighbours, which gives

$$\Delta E = 2J\, s_{i,j} \sum_{s_{k,l} \in N(i,j)} s_{k,l} \;\in\; \{-8J,\, -4J,\, 0,\, 4J,\, 8J\}, \quad (9)$$

with $s_{i,j}$ taken before the flip (Appendix A). Flips with $\Delta E \le 0$ are always accepted, so only $e^{-4\beta J}$ and $e^{-8\beta J}$ are needed. They are computed once per temperature and stored in a table. After an accepted flip, E and M are updated as $E \to E + \Delta E$ and $M \to M + 2 s_{i,j}^{\mathrm{new}}$, at a cost of $\mathcal{O}(1)$ instead of $\mathcal{O}(N)$.

One Monte Carlo cycle consists of N attempted flips (Algorithm 1), so that each spin is proposed once per cycle on average.

D. Sampling and burn-in

The first n_{burn} cycles of each chain are discarded. Afterwards E and M are recorded once per cycle for n

cycles, and expectation values are estimated by sample means,

$$\langle E\rangle \approx \frac{1}{n}\sum_{c=1}^{n} E^{(c)}, \tag{10}$$

and correspondingly for E^2, $|M|$ and M^2, which are inserted into Eqs. (5) and (6). Successive samples are correlated, so the effective number of independent samples is smaller than n.

Chains are started either from an ordered lattice ($s_{i,j} = +1$ for all i, j) or from an unordered lattice, where each spin is ± 1 with equal probability. We choose n_{burn} from the evolution of ϵ and $|m|$ for $L = 20$ at $T = 1.0\,J/k_B$ and $T = 2.4\,J/k_B$, for both initial states.

E. Implementation and tests

The spins are stored row by row in a one-dimensional array of length N. The periodic neighbours are found from two index tables, computed once for $i = 0, \ldots, L-1$,

$$i_+ = (i+1) \bmod L, \qquad i_- = (i-1+L) \bmod L, \tag{11}$$

so that the neighbours of (i, j) are (i_+, j), (i_-, j), (i, j_+) and (i, j_-), and no conditional tests are needed in the inner loop. Adding L keeps the argument of the modulo operation non-negative.

Random numbers are generated with the Mersenne Twister algorithm, which has period $2^{19937} - 1$. Each chain has its own generator and seed.

We test the code in two ways. For $L = 2$ and $T = 1.0\,J/k_B$ we compare the estimates of $\langle\epsilon\rangle$, $\langle|m|\rangle$, C_V and χ with the exact values from Appendix A, as functions of the number of cycles. At the end of every run we also check that the incrementally updated E and M agree with values computed directly from the lattice.

F. Parallelisation

The chains at different temperatures are independent. We therefore parallelise the loop over temperatures with shared-memory threads, assigning the next temperature to whichever thread becomes free (dynamic scheduling), since chains at different temperatures take different times. Each thread holds its own lattice and generator, and temperature number i uses the seed $s_0 + 7919\,i$, so that the results do not depend on the number of threads. The speed-up t_1/t_p, with t_p the wall-clock time on p threads, is measured for 24 temperatures with $L = 20$ and 2×10^4 cycles each, for $p = 1, 2, 4, 8, 12$.

G. Critical temperature

For $L = 40, 60, 80, 100$ we run chains at 31 equally spaced temperatures in $[2.1, 2.4]\,J/k_B$, with $n_{\mathrm{burn}} = 10^5$ and $n = 10^6$. The position of the maximum of C_V or χ defines a critical temperature $T_c(L)$ for each lattice size. These approach the infinite-lattice value as

$$T_c(L) - T_c(L=\infty) = aL^{-1/\nu}, \tag{12}$$

where a is a constant and ν is the critical exponent of the correlation length, $\xi \propto |T - T_c(L=\infty)|^{-\nu}$. For the two-dimensional Ising model $\nu = 1$.

To resolve the maxima below the grid spacing of $0.01\,J/k_B$, we fit a second-degree polynomial to the seven points nearest the largest value and take the vertex as $T_c(L)$. With $\nu = 1$, $T_c(L=\infty)$ is the intercept of a linear regression of $T_c(L)$ against $1/L$, and its uncertainty is the standard error of the intercept. The result is compared with the exact value [3]

$$T_c = \frac{2}{\ln(1+\sqrt{2})}\,\frac{J}{k_B} \approx 2.269\,\frac{J}{k_B}. \tag{13}$$

Tools

The simulations are written in C++, compiled with `g++` with optimisation flag `-O3`, and parallelised with OpenMP [6]. Random numbers come from the standard library implementation of the Mersenne Twister, `std::mt19937` [7]. The figures and the linear regression were made in Python with NumPy [8], SciPy [9] and matplotlib [10]. The AI assistant Claude (Anthropic) was used to help structure the code and the plotting scripts, to help with LaTeX, and to give feedback on the tex and consult on improvements. All code and text have been checked and written by the authors.

III. RESULTS AND DISCUSSION

A. Validation for $L = 2$

Figure 2 shows the relative error of the four estimates for $L = 2$ at $T = 1.0\,J/k_B$, starting from an unordered lattice without burn-in. At this temperature the two ground states carry 99.8 % of the probability. For the first $\sim 10^3$ cycles the chain does not leave the ground states, so the estimates of C_V and χ are zero and their relative error is 1. After that the errors decrease roughly as $1/\sqrt{n}$, with large fluctuations around the trend.

The estimates of $\langle\epsilon\rangle$ and $\langle|m|\rangle$ are within a relative error of 10^{-4} after 10^4 cycles. C_V and χ converge more slowly: after 10^6 cycles we obtain $C_V = 0.03264\,k_B$ and $\chi = 0.004092\,J^{-1}$, which deviate by 1.7 % and 2.0 % from the exact values $0.032082\,k_B$ and $0.004011\,J^{-1}$ (Table I). Good agreement thus requires about 10^4 cycles for the mean values, but of order 10^6 cycles for C_V and χ. Both are variances, determined by the rarely visited excited states, so they need many more samples than the means. The incrementally updated E and M agreed exactly with the recomputed values in all runs.

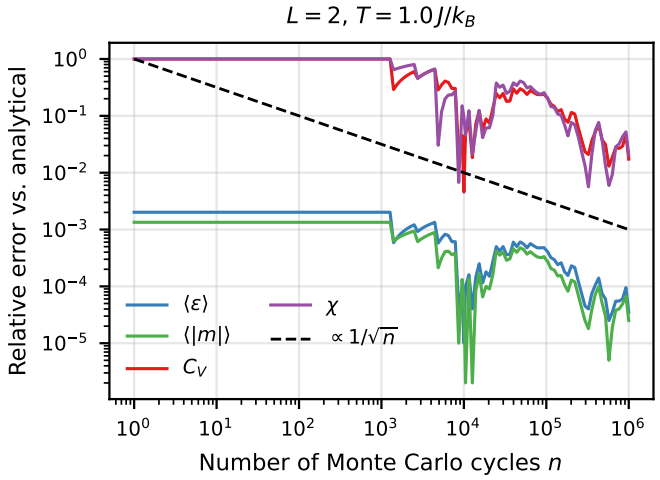

FIG. 2. Relative error of the Monte Carlo estimates for $L = 2$ at $T = 1.0\,J/k_B$ as a function of the number of cycles n. The dashed line is proportional to $1/\sqrt{n}$.

TABLE I. Monte Carlo estimates for $L = 2$, $T = 1.0\,J/k_B$ after 10^6 cycles, compared with the exact values from Appendix A.

Quantity	MC	Exact	Rel. error
$\langle\epsilon\rangle$ $[J]$	-1.99591	-1.99598	3.5×10^{-5}
$\langle\lvert m\rvert\rangle$	0.998636	0.998661	2.5×10^{-5}
C_V $[k_B]$	0.03264	0.03208	1.7×10^{-2}
χ $[J^{-1}]$	0.004092	0.004011	2.0×10^{-2}

B. Burn-in

Figure 3 shows ϵ and $|m|$ for $L = 20$ for the first 10^5 cycles. At $T = 1.0\,J/k_B$ the ordered chain starts in equilibrium. The unordered chain reaches the same energy after about 10^2 cycles, as domains of equal spins grow until one orientation fills the lattice. At $T = 2.4\,J/k_B$ the two chains become indistinguishable after a few hundred cycles, and the fluctuations in both ϵ and $|m|$ are much larger than at $T = 1.0\,J/k_B$.

The running means approach their final values much more slowly than the instantaneous values, since they include the first cycles. At $T = 2.4\,J/k_B$ the means of $|m|$ from the two chains still differ after 10^5 cycles (0.454 and 0.432), because successive values of $|m|$ are strongly correlated close to T_c.

For all four chains, a 100-cycle moving average of ϵ is within half a standard deviation of its equilibrium value after at most 3×10^2 cycles. We use a burn-in of 2×10^4 cycles for $L = 20$ and 10^5 cycles for the larger lattices near T_c, where equilibration is slower.

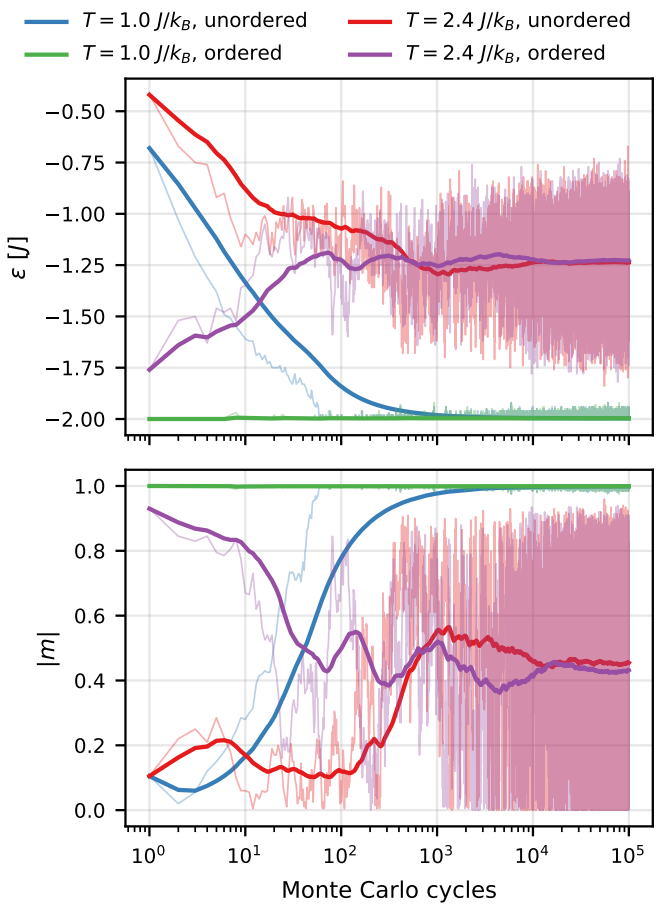

FIG. 3. Energy per spin (top) and absolute magnetisation per spin (bottom) for $L = 20$ as functions of the number of cycles, for ordered and unordered initial states at $T = 1.0\,J/k_B$ and $T = 2.4\,J/k_B$. Thin lines are the values after each cycle, thick lines the running means.

C. Energy distribution

Figure 4 shows the estimated probability distribution of ϵ for $L = 20$, from 10^6 cycles after a burn-in of 2×10^4 cycles. Since E changes in steps of $4J$, each bin contains exactly one allowed value of ϵ.

At $T = 1.0\,J/k_B$ the lattice is in the ground state $\epsilon = -2J$ with probability 0.87. The next value, $\epsilon = -2J + 8J/N = -1.98J$, corresponds to a single flipped spin and has probability 0.12. The variance is $\mathrm{Var}(\epsilon) = 5.8 \times 10^{-5}\,J^2$. At $T = 2.4\,J/k_B$ the distribution is broad and close to Gaussian, with $\langle\epsilon\rangle = -1.236\,J$ and $\mathrm{Var}(\epsilon) = 2.0 \times 10^{-2}\,J^2$, more than two orders of magnitude larger. Since $C_V = N\,\mathrm{Var}(\epsilon)/(k_B T^2)$, the widths correspond to $C_V = 0.023\,k_B$ and $1.41\,k_B$.

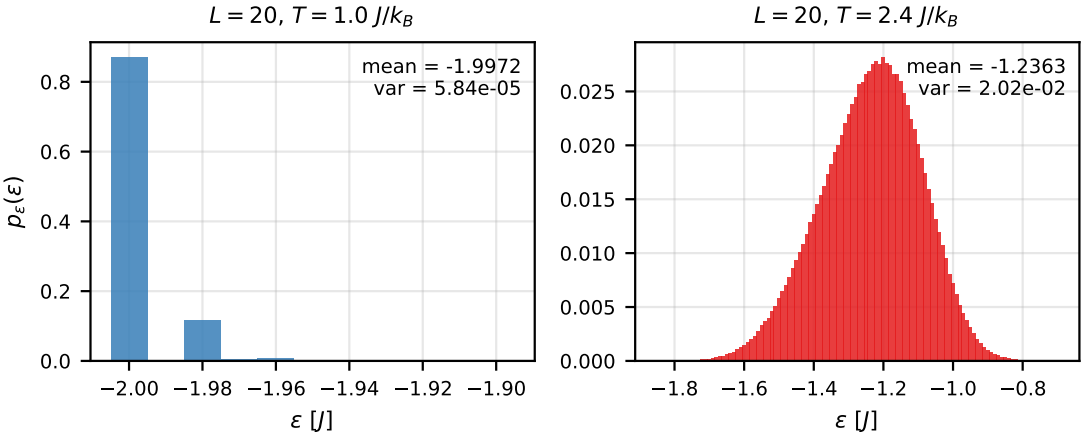

FIG. 4. Normalised histograms of ϵ for $L = 20$ at $T = 1.0\,J/k_B$ (left) and $T = 2.4\,J/k_B$ (right), from 10^6 cycles after a burn-in of 2×10^4 cycles. Each bin contains one allowed value of ϵ.

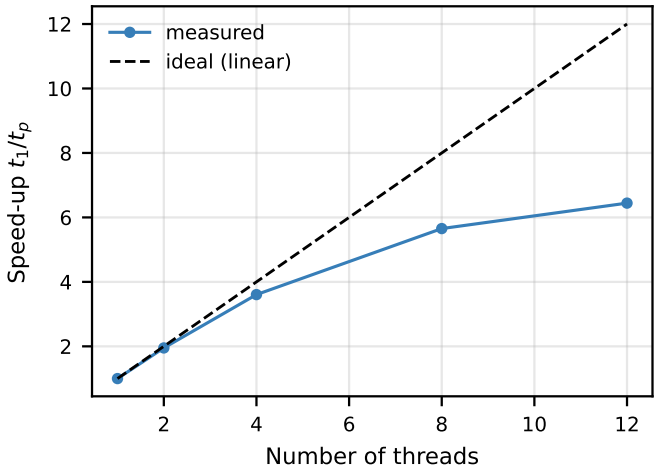

FIG. 5. Speed-up t_1/t_p for a scan of 24 temperatures ($L = 20$, 2×10^4 cycles each) as a function of the number of threads p. The dashed line is the ideal linear speed-up.

D. Parallelisation

Figure 5 shows the measured speed-up. It is close to linear for two (1.95) and four threads (3.61), and levels off at 5.65 for eight and 6.44 for twelve threads. The processor has twelve hardware threads but fewer physical cores, and threads sharing a core compete for the same execution units. Although 24 temperatures divide evenly between eight threads, chains at different temperatures take different times, so some threads finish early and stay idle. On one thread the code performs about 7×10^7 attempted flips per second.

E. Phase transition

Figure 6 shows $\langle\epsilon\rangle$, $\langle|m|\rangle$, C_V and χ for $L = 40, 60, 80, 100$. The full scan took 29 minutes on twelve threads, of which 14 minutes were spent on $L = 100$, consistent with a cost proportional to L^2.

Below $T \approx 2.25\,J/k_B$ the curves for all L coincide. Around T_c the magnetisation drops, and the drop becomes steeper with increasing L. Above T_c, $\langle|m|\rangle$ decreases faster for larger lattices, towards the value $m = 0$ of the infinite system. It remains positive for every finite L because $|m| \geq 0$.

C_V and χ both have a maximum that grows and moves to lower T as L increases. The maximum of C_V increases slowly, from $2.0\,k_B$ at $L = 40$ to $2.5\,k_B$ at $L = 100$, in line with the logarithmic divergence of the exact solution. The maximum of χ grows from $30.6\,J^{-1}$ to $151.9\,J^{-1}$, a factor of 4.97, which agrees with $(100/40)^{7/4} = 4.97$, the expected finite-size scaling $\chi_{\max} \propto L^{7/4}$. The maxima of χ lie at higher temperatures than those of C_V. Part of this shift is caused by using $\langle|M|\rangle$ instead of $\langle M\rangle$ in Eq. (6).

F. Critical temperature

Table II lists $T_c(L)$ from the maxima of C_V and χ, and Fig. 7 shows them against $1/L$. Both sets lie close to straight lines, as expected from Eq. (12) with $\nu = 1$. The linear regressions give

$$T_c(L = \infty) = 2.2667 \pm 0.0005\,J/k_B \quad (C_V), \tag{14}$$
$$T_c(L = \infty) = 2.2715 \pm 0.0016\,J/k_B \quad (\chi), \tag{15}$$

with slopes $a = 0.91$ and $a = 2.08$. The relative deviations from the exact value $2.2692\,J/k_B$ are 1.1×10^{-3} and 1.0×10^{-3}, with the two estimates on opposite sides of it.

The estimate from C_V lies $2.5 \times 10^{-3}\,J/k_B$ below the exact value, about five times its standard error. The standard error only describes how well four points fit a line. It does not include corrections of higher order in $1/L$, which are largest for $L = 40$, or the error from

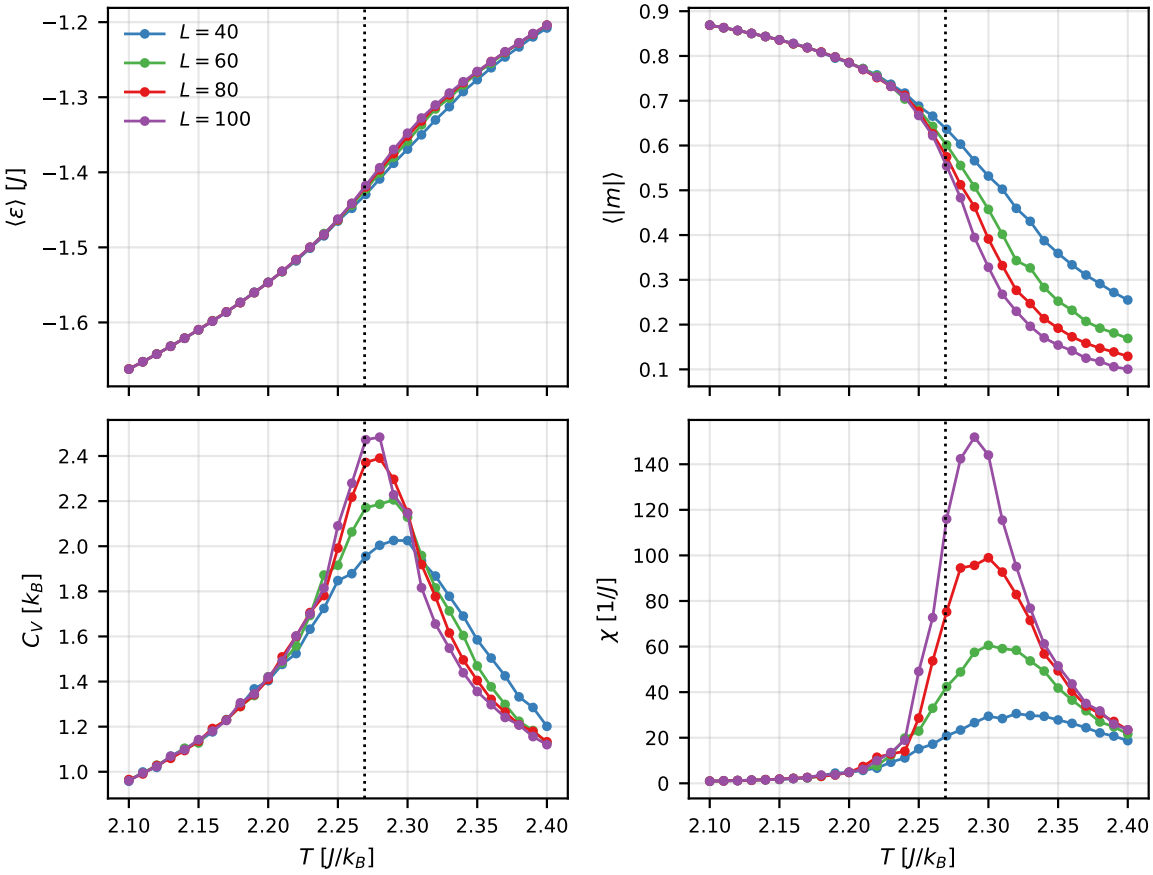

FIG. 6. $\langle\epsilon\rangle$, $\langle|m|\rangle$, C_V and χ as functions of temperature for $L = 40, 60, 80, 100$, each point from 10^6 cycles after a burn-in of 10^5 cycles. The dotted line marks the exact T_c, Eq. (13).

TABLE II. $T_c(L)$ from the maxima of C_V and χ, the extrapolated values and the exact result. All values in units of J/k_B.

L	$T_c(L)$ from C_V	$T_c(L)$ from χ
40	2.2895	2.3230
60	2.2823	2.3074
80	2.2784	2.2975
100	2.2756	2.2916
∞ (fit)	2.2667 ± 0.0005	2.2715 ± 0.0016
∞ (exact)	2.2692	

approximating each peak by a parabola. A realistic uncertainty is therefore around $3 \times 10^{-3}\, J/k_B$, or 0.1 % of T_c.

G. Limitations

The main limitation is the resolution of the maxima. The temperature step is $0.01\, J/k_B$, and C_V and χ are noisy near their peaks, most visibly for C_V at $L = 40$. A finer grid around the peaks and several independent chains per temperature, to give error bars on $T_c(L)$, would make the extrapolation more reliable. Larger lattices would reduce the finite-size corrections, but near T_c the autocorrelation time of single-flip Metropolis grows roughly as L^2 (critical slowing down). Cluster algorithms such as the Wolff algorithm avoid most of this. Finally, we fixed $\nu = 1$. Fitting ν as well would test Eq. (12) instead of assuming it, but requires more lattice sizes and more precise peak positions.

IV. CONCLUSION

We have used the Metropolis algorithm to estimate the critical temperature of the two-dimensional Ising model. The method reproduces the exact results for a 2×2 lattice, to a relative error of 10^{-4} for the mean energy and magnetisation and within 2 % for the heat capacity and susceptibility, which converge more slowly. For $L = 20$ the chains reach equilibrium within a few hundred cycles, and the energy distribution broadens from almost only the ground state at $T = 1.0\, J/k_B$ to a nearly Gaussian shape at $T = 2.4\, J/k_B$.

The finite-size scaling of the peak positions works well already for $L = 40$ to 100: the heat capacity and the

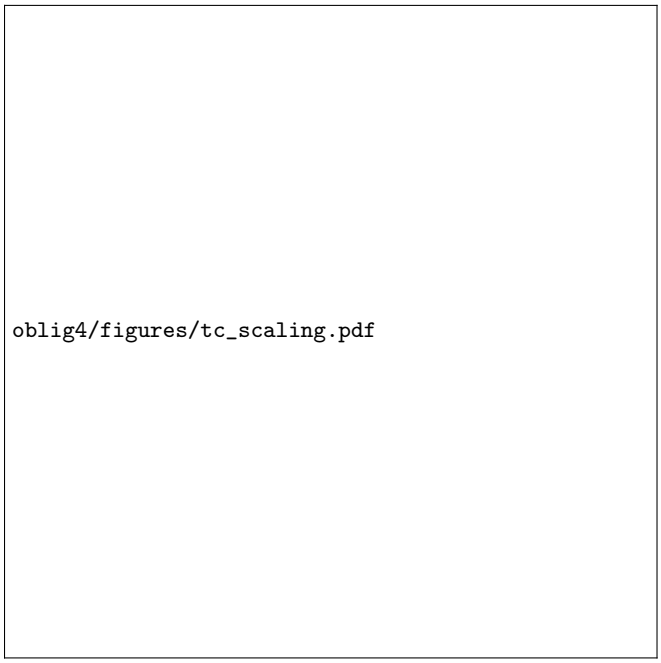

FIG. 7. $T_c(L)$ from the maxima of C_V and χ against $1/L$, with linear fits to Eq. (12). The intercepts estimate $T_c(L=\infty)$. The star marks the exact value.

susceptibility give estimates of the critical temperature that bracket Onsager's value, $2.5\times10^{-3}\,J/k_B$ below and $2.3\times10^{-3}\,J/k_B$ above it. This is larger than the standard errors of the fits, so the remaining error is systematic, from corrections of higher order in $1/L$ and the temperature resolution of $0.01\,J/k_B$. A finer scan near the maxima, with error estimates from independent chains, would reduce both. Since the method reproduces Onsager's exact result to 0.1 % in two dimensions, the same approach can be applied with confidence to the three-dimensional model, where no exact solution is known.

Appendix A: Analytical results

In this appendix we derive the analytical results used in Sec. II and Sec. III. Throughout, we consider the Ising model on a 2×2 lattice with periodic boundary conditions, no external field, and units where $J=1$ and $k_B=1$ (we keep J and β explicit in the intermediate steps for clarity).

1. Energy and magnetisation of the microstates

a. *Energy.*

Write the spin configuration as

$$\mathbf{s}=\begin{pmatrix} a & b \\ c & d \end{pmatrix}, \qquad a,b,c,d\in\{-1,+1\}. \tag{A1}$$

The distinct neighbouring pairs are (a,b) and (c,d) horizontally, and (a,c) and (b,d) vertically. With periodic boundary conditions on a 2×2 lattice, each spin's left and right neighbour is the same spin, and likewise for up and down, so every pair is counted twice in $E=-J\sum_{\langle ij\rangle}s_is_j$. This gives

$$E=-2J\,(ab+ac+bd+cd). \tag{A2}$$

We choose to keep this double counting, so that the boundary conditions work in the same way as for general L in the code.

b. *Magnetisation.*

Let N_+ and $N_-=4-N_+$ be the number of spins equal to $+1$ and -1. Then

$$M=\sum_i s_i=N_+-N_-=2N_+-4. \tag{A3}$$

c. *Enumeration of states.*

There are $2^4=16$ microstates, which we group by N_+:

- $N_+=4$ or $N_+=0$ (one state each): all spins are parallel, so every product in Eq. (A2) equals $+1$ and $E=-8J$.
- $N_+=3$ or $N_+=1$ (four states each): one spin differs from the other three. Two of the four products are -1 and two are $+1$, so $E=0$.
- $N_+=2$ (six states): in four of them the two up-spins are neighbours (e.g. $a=b=+1$, $c=d=-1$), which gives $ab=cd=+1$ and $ac=bd=-1$, so $E=0$. In the remaining two the up-spins sit diagonally ("checkerboard", $a=d=+1$, $b=c=-1$, or the reverse), so all four products are -1 and $E=+8J$.

The result is summarised in Table III. The degeneracies sum to $1+4+2+4+4+1=16$, as they should.

N_+	E	M	Degeneracy
4	$-8J$	$+4$	1
3	0	$+2$	4
2	$+8J$	0	2
2	0	0	4
1	0	-2	4
0	$-8J$	-4	1

TABLE III. All microstates of the 2×2 lattice.

2. Partition function

The partition function is the sum of Boltzmann weights over all microstates. Using Table III,

$$\begin{aligned} Z &= \sum_i e^{-\beta E_i} \\ &= 2e^{8\beta J} + 2e^{-8\beta J} + 12 \\ &= 4\left[\cosh(8\beta J) + 3\right]. \end{aligned} \tag{A4}$$

Each term is weighted by its degeneracy: the two ordered states give $2e^{8\beta J}$, the two checkerboard states give $2e^{-8\beta J}$, and the twelve states with $E = 0$ give 12.

3. Expectation values

With $P_i = e^{-\beta E_i}/Z$ and $\langle A\rangle = \sum_i A_i P_i$, we read off the sums directly from Table III.

a. *Energy.*

$$\begin{aligned} \langle E\rangle &= \frac{1}{Z}\left[2(-8J)e^{8\beta J} + 2(8J)e^{-8\beta J}\right] \\ &= -\frac{32J\sinh(8\beta J)}{Z} \\ &= -\frac{8J\sinh(8\beta J)}{\cosh(8\beta J)+3}, \end{aligned} \tag{A5}$$

$$\begin{aligned} \langle E^2\rangle &= \frac{1}{Z}\left[2(64J^2)e^{8\beta J} + 2(64J^2)e^{-8\beta J}\right] \\ &= \frac{256J^2\cosh(8\beta J)}{Z}. \end{aligned} \tag{A6}$$

b. *Magnetisation.*

For every state with magnetisation M there is a spin-flipped partner with magnetisation $-M$ and the same energy, hence the same probability. Therefore

$$\langle M\rangle = 0 \tag{A7}$$

at all temperatures. As in the project text, we instead use $|M|$, which gives

$$\begin{aligned} \langle |M|\rangle &= \frac{1}{Z}\left[2\cdot 4\, e^{8\beta J} + 8\cdot 2\right] \\ &= \frac{8e^{8\beta J}+16}{Z}, \end{aligned} \tag{A8}$$

$$\begin{aligned} \langle M^2\rangle &= \frac{1}{Z}\left[2\cdot 16\, e^{8\beta J} + 8\cdot 4\right] \\ &= \frac{32e^{8\beta J}+32}{Z}. \end{aligned} \tag{A9}$$

Here the factor 8 counts the eight states with $|M| = 2$, all of which have $E = 0$. The six states with $M = 0$ do not contribute.

c. *Per spin.*

Dividing by $N = 4$,

$$\langle \epsilon\rangle = \frac{\langle E\rangle}{4} = -\frac{2J\sinh(8\beta J)}{\cosh(8\beta J)+3}, \tag{A10}$$

$$\langle |m|\rangle = \frac{\langle |M|\rangle}{4} = \frac{e^{8\beta J}+2}{2\left[\cosh(8\beta J)+3\right]}. \tag{A11}$$

As a check, for $\beta \to \infty$ we get $\langle\epsilon\rangle \to -2J$ and $\langle |m|\rangle \to 1$ (fully ordered), while for $\beta \to 0$ we get $\langle\epsilon\rangle \to 0$ and $\langle |m|\rangle \to 3/8$, which equals the average of $|M|/4$ over all 16 states.

4. Heat capacity and susceptibility

As in Eqs. (5) and (6), C_V and χ are given per spin.

a. *Heat capacity.*

Since $\langle E\rangle = -\partial_\beta \ln Z$, we have $\partial_\beta\langle E\rangle = -\left(\langle E^2\rangle - \langle E\rangle^2\right)$. Using $\partial\beta/\partial T = -1/(k_B T^2) = -k_B\beta^2$,

$$\begin{aligned} C_V &= \frac{1}{N}\frac{\partial\langle E\rangle}{\partial T} = \frac{1}{N}\frac{\partial\beta}{\partial T}\frac{\partial\langle E\rangle}{\partial\beta} \\ &= \frac{k_B\beta^2}{N}\left(\langle E^2\rangle - \langle E\rangle^2\right), \end{aligned} \tag{A12}$$

which is Eq. (5). With $N = 4$ and $x \equiv 8\beta J$, inserting Eqs. (A4), (A5) and (A6) gives

$$C_V = \frac{k_B\beta^2}{4}\left[\frac{256J^2\cosh x}{Z} - \frac{1024J^2\sinh^2 x}{Z^2}\right] = 16\, k_B\beta^2 J^2\left[\frac{\cosh x}{\cosh x + 3} - \frac{\sinh^2 x}{(\cosh x + 3)^2}\right]. \tag{A13}$$

Combining the terms over a common denominator and using $\cosh^2 x - \sinh^2 x = 1$,

$$C_V = \frac{16\, k_B\beta^2 J^2\ (1 + 3\cosh x)}{(\cosh x + 3)^2}. \tag{A14}$$

b. Susceptibility.

Following the project text, we replace $M \to |M|$ and use Eq. (6),

$$\chi = \frac{\beta}{N}\left(\langle M^2\rangle - \langle |M|\rangle^2\right). \tag{A15}$$

Inserting Eqs. (A4), (A8) and (A9) with $N = 4$,

$$\chi = \frac{\beta}{4}\left[\frac{8\,(e^x+1)}{\cosh x + 3} - \frac{4\,(e^x+2)^2}{(\cosh x+3)^2}\right] = \beta\left[\frac{2\,(e^x+1)}{\cosh x + 3} - \frac{(e^x+2)^2}{(\cosh x+3)^2}\right]. \tag{A16}$$

Equations (A5), (A8), (A14) and (A16) are the reference values used to test the Monte Carlo code.

5. Energy change from a single spin flip

a. General expression.

Consider a spin s_k with the four neighbours s_n (left, right, up, down). Only the bonds involving s_k change when it is flipped. Before the flip they contribute $-Js_k\sum_n s_n$ to the energy, and after the flip $+Js_k\sum_n s_n$. Hence

$$\Delta E = E_{\text{after}} - E_{\text{before}} = 2J\, s_k \sum_{n=1}^{4} s_n. \tag{A17}$$

For $L = 2$ the four neighbours are only two distinct spins, each appearing twice. Equation (A17) is then consistent with Eq. (A2): flipping a gives $\Delta E = 4J\,a(b+c)$ in both cases.

b. Possible values.

The neighbour sum $\sum_n s_n$ is a sum of four numbers ± 1, so it can only take the values $-4, -2, 0, 2, 4$. Since $s_k = \pm 1$, the product $s_k\sum_n s_n$ takes the same five values (the sign of s_k only permutes them), and

$$\Delta E \in \{-8J,\ -4J,\ 0,\ 4J,\ 8J\}. \tag{A18}$$

c. Boltzmann factors.

The acceptance probability in the Metropolis algorithm is $\min(1, e^{-\beta\Delta E})$. For $\Delta E \le 0$ it equals 1, so the only nontrivial factors are

$$e^{-4\beta J}\quad(\Delta E = 4J), \qquad e^{-8\beta J}\quad(\Delta E = 8J), \tag{A19}$$

which can be computed once per temperature and stored in an array. This avoids calling the exponential function inside the main Monte Carlo loop.

[1] Hugo Duminil-Copin. 100 years of the (critical) Ising model on the hypercubic lattice. In *Proceedings of the International Congress of Mathematicians 2022*, volume 1, pages 164–210. EMS Press, 2023.

[2] Thomas Ising, Reinhard Folk, Ralph Kenna, Bertrand Berche, and Yurij Holovatch. The fate of Ernst Ising and the fate of his model. *Journal of Physical Studies*, 21(3):3002, 2017.

[3] Lars Onsager. Crystal statistics. I. A two-dimensional model with an order-disorder transition. *Physical Review*, 65(3–4):117–149, 1944.

[4] Martin Niss. *Phenomena, Models and Understanding: The Lenz–Ising Model and Critical Phenomena 1920–1971*. PhD thesis, Roskilde University, 2005. IMFUFA tekst 444.

[5] N. Metropolis, Arianna W. Rosenbluth, M. N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[6] Leonardo Dagum and Ramesh Menon. OpenMP: an industry standard API for shared-memory programming. *IEEE Computational Science and Engineering*, 5(1):46–55, 1998.

[7] Makoto Matsumoto and Takuji Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation*, 8(1):3–30, 1998.

[8] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, et al. Array programming with NumPy. *Nature*, 585:357–362, 2020.

[9] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, 2020.

[10] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.