University of Trento
Department of Physics
Bachelor's Degree in Physics

~ · ~

Academic Year 2023–2024

Numerical Implementation of the Spherical Model

Supervisor
Prof. Stefano Giorgini

Graduate Student
Marco Fava
227408

Final examination date: November 18, 2024

Acknowledgments

I would like to express my gratitude to my supervisor, Professor Giorgini, for his guidance and support throughout the course of this thesis. Their insightful feedback and encouragement were essential to the completion of this work. I would also like to extend my sincere thanks to my colleague, Edoardo Stazzu, for his collaborative spirit and helpful discussions, which greatly inspired and enriched my research since the beginning.

Abstract

This thesis investigates the numerical implementation of the spherical model using Monte Carlo simulations, focusing on the efficiency of different Metropolis algorithms in achieving thermalisation and exploring equilibrium states. The work compares standard Metropolis methods with adaptive approaches, particularly a mixed θ-strategy, to evaluate their impact on convergence speed and accuracy. Key thermodynamic quantities, such as internal energy, magnetisation, specific heat, and susceptibility, are analysed across different system sizes and dimensions. The results demonstrate that adaptive methods, especially the mixed θ-method, significantly enhance convergence efficiency compared to conventional approaches, enabling faster sampling of equilibrium states. The thermodynamic properties obtained from the simulations align closely with theoretical expectations, though minor deviations due to finite-size effects are observed. This work provides a strong foundation for further studies on the spherical model, suggesting potential directions for future research into critical exponents, autocorrelation functions, and the model's behaviour under varying external magnetic fields.

Contents

Introduction

The spherical model has long stood as a cornerstone in ferromagnetic studies, offering valuable insights into phase transitions and critical phenomena. Its solvability and analytical simplicity, particularly in high dimensions, make it an essential tool for exploring complex systems. However, translating theoretical predictions into practical, numerically accurate results poses significant challenges. Among these, the efficient exploration of the configuration space and the accurate determination of thermodynamic quantities near critical points are particularly prominent.

This thesis focuses on the implementation of the spherical model through Monte Carlo simulations, with an emphasis on understanding the convergence properties of various Metropolis algorithms. By leveraging adaptive strategies, we aim to overcome the limitations of standard approaches, enabling faster thermalisation and more reliable sampling of equilibrium states. The study further investigates the model's behaviour across different system sizes and dimensionalities, examining key thermodynamic quantities such as internal energy, magnetisation, specific heat, and susceptibility.

Through this work, we not only assess the numerical performance of adaptive methods but also delve into the physical implications of finite-size effects and their impact on critical temperature estimates. The findings presented here aim to close the gap between theoretical predictions and computational implementations, providing a robust framework for future research into the spherical model and its broader applications in statistical mechanics.

The implementation of the spherical model and the numerical simulations discussed in this thesis are available in an open-source repository on GitHub: https://github.com/MarcoFava/Spherical-Model.

Chapter 1

Theoretical Background of the Spherical Model

1.1 The Gaussian model

Kac, in 1947, proposed a model where the spins σ_i take continuous values from $-\infty$ to $+\infty$, instead of discrete values (-1 or $+1$ for the Ising model), and are subject to a Gaussian probability distribution:

$$p(\sigma_i)d\sigma_i = \sqrt{\frac{A}{\pi}}e^{-A\sigma_i^2}d\sigma_i, \qquad (i = 1, \ldots, N),$$

such that on average, $\langle\sigma_i\rangle = 0$ and $\langle\sigma_i^2\rangle = \frac{1}{2A}$. By setting $\langle\sigma_i^2\rangle = 1$ as a standard practice, we choose $A = \frac{1}{2}$.
The partition function, in the presence of an external field, is given by:

$$Q_N = \int_{-\infty}^{+\infty}\cdots\int_{-\infty}^{+\infty}\left(\frac{A}{\pi}\right)^{N/2}\exp\left[-A\sum_i\sigma_i^2 + K\sum_{(i,j)}\sigma_i\sigma_j + h\sum_i\sigma_i\right]\prod_i d\sigma_i, \qquad (K = J/kT,\ h = H/kT).$$

The expression in the exponent is a symmetric, quadratic function in the σ_i, so it can be diagonalised, and the integral can be solved in any dimension (see section 2.1.1 for further information on the construction of the interaction matrix needed for the sum $\sum_{(i,j)}\sigma_i\sigma_j$). The calculations were further developed by Berlin and Kac [2] in 1952, and the results for the free energy per particle ψ in one, two, and three dimensions are given as follows:

$$-\frac{\psi^{(1D)}}{k_BT} = -\frac{1}{2}\frac{1}{2\pi}\int_0^{2\pi}d\omega_1\ln[1-2K\cos\omega_1], \tag{1.1}$$

$$-\frac{\psi^{(2D)}}{k_BT} = -\frac{1}{2}\frac{1}{(2\pi)^2}\int_0^{2\pi}\int_0^{2\pi}d\omega_1 d\omega_2\ln[1-2K(\cos\omega_1+\cos\omega_2)], \tag{1.2}$$

$$-\frac{\psi^{(3D)}}{k_BT} = -\frac{1}{2}\frac{1}{(2\pi)^3}\int_0^{2\pi}\int_0^{2\pi}\int_0^{2\pi}d\omega_1 d\omega_2 d\omega_3\ln[1-2K(\cos\omega_1+\cos\omega_2+\cos\omega_3)]. \tag{1.3}$$

The problem with this model arises for temperatures lower than the critical temperature threshold T_C, given by the relation $K_C = A/d = 1/2d$[1] (with d being the dimensionality of the

[1] Note that this is the same result obtained from the mean-field theory.

system), at which point the free energy ψ becomes complex, causing the model to break down.

Moreover, it can be shown that the system undergoes a phase transition at the finite temperature T_C for every dimension greater than 2 ($d > 2$), with the critical exponents being non-classical for $2 < d < 4$.

For comparison, we also provide the expressions for the free energy of the one-dimensional and two-dimensional Ising models:

$$-\frac{\psi_{\text{Ising}}^{(1D)}}{k_B T} = -\frac{1}{2}\frac{1}{2\pi}\int_0^{2\pi} d\omega_1 \ln[\cosh(4K) - \sinh(4K)\cos\omega_1], \tag{1.4}$$

$$-\frac{\psi_{\text{Ising}}^{(2D)}}{k_B T} = -\frac{1}{2}\frac{1}{(2\pi)^2}\int_0^{2\pi}\int_0^{2\pi} d\omega_1 d\omega_2 \ln[\cosh^2(4K) - \sinh(4K)(\cos\omega_1 + \cos\omega_2)]. \tag{1.5}$$

There is a formal similarity between the expressions for the free energy per particle of the Ising and Gaussian models.

1.2 The spherical model

To develop a model valid at all temperatures, Berlin and Kac replaced the Gaussian distribution of the spins with a uniform distribution while introducing an additional constraint on the spins:

$$\sum_{i=1}^{N} \sigma_i^2 = N. \tag{1.6}$$

This constraint brings the model closer to the Ising model, as it not only ensures $\langle \sigma_i \rangle = 0$ and $\langle \sigma_i^2 \rangle = 1$ but also imposes a stronger, more Ising-like condition on the system. While the Ising model configurations can be visualised as the vertices of a hyper-dimensional cube, this model's spin configurations can be thought of as components of a vector lying on a hypersphere in N-dimensional space.

The model may seem unphysical due to non-local interactions and because it retains some issues typical of systems with continuous spins. As a matter of fact, at absolute zero, the entropy remains finite, and the specific heat per particle does not vanish but instead approaches a finite value. However, Stanley 1968 showed it to be a limiting case of the n-vector mode (see [7]), thus justifying its use in modelling critical behaviours, where our primary focus should lie.

1.2.1 Free Energy

The first step is to evaluate the partition function. The constraint can be visualized as a delta function multiplying the integrand (i.e., $\delta[N - \sum_j \sigma_j^2]$). Additionally, an extra factor $\exp\left[aN - a\sum_j \sigma_j^2\right]$ can be introduced into the integrand, as the delta function ensures this factor equals unity. Using the integral representation of the delta function, we write:

$$Z_N = \frac{1}{2\pi}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} d\sigma_1 \ldots d\sigma_N \int_{-\infty}^{\infty} ds \exp\left[K\sum_{(j,l)}\sigma_j\sigma_l + h\sum_j \sigma_j + (a+is)N - (a+is)\sum_j \sigma_j^2\right], \tag{1.7}$$

where the first summation runs over all neighboring spin pairs (j,l) on the hyper-lattice without double counting, and the second summation is over all lattice sites j. The key step is to rewrite the exponent as a quadratic form involving a cyclic symmetric matrix V. By choosing a sufficiently large, we ensure that all eigenvalues of V have positive real parts. Unlike the Gaussian model, this ensures the spherical model's partition function is defined at all temperatures.

Next, the order of integration between $\bar{\sigma}$ and s can be interchanged. By introducing a change of variables from $\bar{\sigma}$ to $\bar{t} = \bar{\sigma} - \frac{1}{2}V^{-1}\bar{h}$, where $\bar{h}$ is an N-dimensional vector with all components equal to h, the integrand simplifies. The matrix V is then diagonalized via an orthogonal transformation, similar to the Gaussian model. This yields:

$$\bar{\sigma}'V\bar{\sigma} = (a+is)\sum_j \sigma_j^2 - K\sum_{(j,l)}\sigma_j\sigma_l \quad \longrightarrow \quad Z_N = \frac{1}{2\pi}\pi^{N/2}\int_{-\infty}^{\infty} ds\,[\det V]^{-1/2}\exp\left[(a+is)N + \frac{1}{4}\bar{h}'V^{-1}\bar{h}\right]. \tag{1.8}$$

Since V is cyclic and the lattice structure is a hypercube (i.e., $N = L^d$, where L is the number of lattice points per side), the eigenvalues of V are given by [2]:

$$\lambda(\omega_1,\ldots,\omega_d) = a + is - K(\cos\omega_1 + \cdots + \cos\omega_d), \qquad \text{where} \quad \omega_j = 0, \frac{2\pi}{L}, \ldots, \frac{2\pi(L-1)}{L}. \tag{1.9}$$

This provides an alternative form for the eigenvalues of the cyclic matrix encountered in the Gaussian model. The determinant of V, being the product of its eigenvalues, can be expressed for large L[2] as:

$$\ln\det V = N[\ln K + g(z)], \quad \text{where} \tag{1.10}$$

$$z = \frac{a + is - Kd}{K}, \tag{1.11}$$

$$g(z) = (2\pi)^{-d}\int_0^{2\pi}\cdots\int_0^{2\pi} d\omega_1\ldots d\omega_d \ln[z + d - \cos\omega_1 - \cdots - \cos\omega_d]. \tag{1.12}$$

Changing the integration variable from s to z, the partition function becomes:

$$Z_N = \frac{K}{2\pi i}\left(\frac{\pi}{K}\right)^{N/2}\int_{c-i\infty}^{c+i\infty} dz\exp[N\phi(z)], \tag{1.13}$$

$$\phi(z) = Kz + Kd - \frac{1}{2}g(z) + \frac{h^2}{4Kz}, \tag{1.14}$$

$$c = \frac{a - Kd}{K}. \tag{1.15}$$

From (1.9), it follows that all eigenvalues have positive real parts if $a > Kd$, implying $c > 0$. Since $\phi(z)$ is analytic for $\mathrm{Re}(z) > 0$, the integral in (1.15) remains unchanged for any positive c. Applying the saddle-point method for large N[3], the function $\phi(z)$ is analysed for real, positive z. When $K > 0$ and $h \neq 0$, $\phi(z)$ possesses a unique minimum at some positive value z_0. This point z_0 dominates the integral in the large N limit. The free energy per particle is then given by:

$$-\frac{f}{k_BT} = \lim_{N\to\infty}\frac{1}{N}\ln Z_N = \frac{1}{2}\ln\left(\frac{\pi}{K}\right) + \phi(z_0). \tag{1.16}$$

[2]Using $\sum_k f(k)\cdot\frac{2\pi}{L} \to \int f(k)\,dk$ as $N\to\infty$.
[3]Refer to pages 45-47 of [5].

The value of z_0 is determined by the saddle-point condition $\phi'(z_0) = 0$:

$$K - \frac{h^2}{4Kz_0^2} = \frac{1}{2}g'(z_0), \tag{1.17}$$

which has a unique positive solution. This makes f a well-defined function of K and h for $K > 0$ and $h \neq 0$.

1.2.2 Equation of state and evaluation of $g'(z)$

For the general Ising model, the magnetisation M corresponds to the average magnetic moment per site. In the thermodynamic limit, as $N \to \infty$, we have

$$M(H,T) = \frac{\langle \sum_i^N \sigma_i \rangle}{N} = -\frac{\partial f(H,T)}{\partial H}. \tag{1.18}$$

Differentiating the previously derived free energy per particle with respect to h and recalling that $\phi'(z_0) = 0$, it follows that

$$M = \frac{h}{2Kz_0} = \frac{H}{2Jz_0}. \tag{1.19}$$

This expression can be used to eliminate z_0 from (1.17), yielding the exact equation of state for the spherical model, which relates M, H, and T:

$$2J(1 - M^2) = k_B T g'\left(\frac{H}{2JM}\right). \tag{1.20}$$

In the canonical ensemble, the internal energy per particle u is defined as

$$u = \frac{\langle E \rangle}{N} = -T^2 \frac{\partial}{\partial T}\left(\frac{f(H,T)}{T}\right), \tag{1.21}$$

holding J and H fixed. Using (1.19), we find an exact relation between the internal energy and magnetisation:

$$u = \frac{1}{2}k_B T - Jd - \frac{1}{2}H(M + M^{-1}). \tag{1.22}$$

Next, we turn to evaluating the function $g'(z)$, which appears in the equation of state. Using the identity $\lambda^{-1} = \int_0^\infty \exp(-\lambda t)\, dt$,

$$g'(z) = (2\pi)^{-d} \int_0^{2\pi} \cdots \int_0^{2\pi} d\omega_1 \ldots d\omega_d \int_0^\infty dt \exp\left[-t\left(z + d - \sum_{j=1}^{d} \cos\omega_j\right)\right]. \tag{1.23}$$

For $\mathrm{Re}(z) > 0$, the integrals converge and can be rearranged. Using the definition of the Bessel function $J_0(it)$, we get

$$g'(z) = \int_0^\infty \exp[-t(z+d)]\, [J_0(it)]^d\, dt. \tag{1.24}$$

For $\mathrm{Re}(z) > 0$, this integral converges, making $g'(z)$ analytic in the right half-plane. Moreover, $g'(z)$ decreases monotonically as $z \to \infty$. The critical properties of the system depend on the value of $g'(0) - g'(z)$, as will be shown later. Specifically,

$$g'(0) = \infty, \quad \text{if} \quad 0 < d \leq 2, \tag{1.25}$$

$$< \infty, \quad \text{if} \quad d > 2. \tag{1.26}$$

Repeating the same analysis for $g''(z)$ and neglecting terms of relative order t^{-1} for $d < 4$, we find

$$g''(z) \simeq -(2\pi)^{-\frac{d}{2}}\Gamma(2-\frac{d}{2})z^{\frac{d}{2}-2}, \quad \text{for} \quad d<4 \quad \longrightarrow \quad g''(0)=\infty \quad \text{if} \quad 0<d\leq 4, \tag{1.27}$$

while for $d > 4$, the integral for $g''(0)$ converges. For $d = 4$, the limiting value of $g''(z)$ can be calculated exactly as $g''(z) \simeq -(2\pi)^{-2} \ln z$.

Introducing appropriate positive coefficients A_d, it follows that for small z:

$$g'(0) - g'(z) \simeq A_d z^{\frac{d}{2}-1}, \quad 2 < d < 4, \tag{1.28}$$

$$\simeq A_4 z \ln(1/z), \quad d = 4, \tag{1.29}$$

$$\simeq A_d z, \quad d > 4. \tag{1.30}$$

1.2.3 Existence of a Critical Point for $d > 2$

For a fixed temperature T, and hence a fixed K, we have shown from equation (1.19) that M can be expressed in terms of z_0 as a function of h. To understand the behaviour of this function at specific temperatures, we consider the graphical method[4], shown in figure 1.1, by plotting the two sides of equation (1.17) as functions of positive z. The intersection point P of these two curves represents z_0.

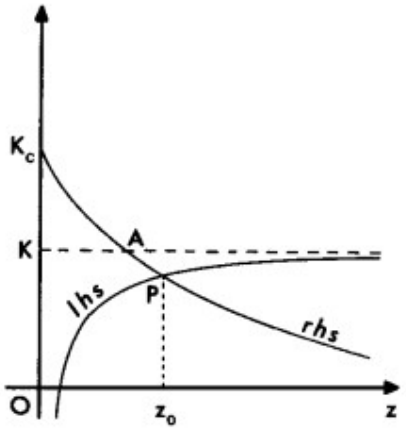

Figure 1.1: Typical sketches of the two sides of (1.17): as h^2 decreases to zero, P moves towards the horizontal line KA.

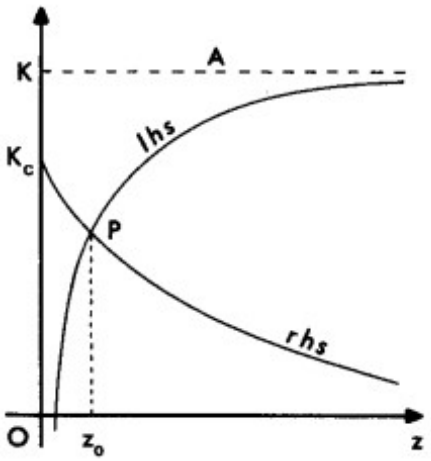

Figure 1.2: Typical sketches of the two sides of (1.17): as h^2 decreases to zero, P moves towards the vertical line OK.

When $h \neq 0$ (and thus $H \neq 0$), z_0 varies smoothly with h. In fact, z_0 is an even analytic function of h, which implies that M is an odd analytic function of H for $H \neq 0$. Now, as h^2 approaches zero, the left-hand side (LHS) curve in Figures 1.1 and 1.2 becomes the step function OKA. Two limit cases arise.

We define the critical point K_c as

$$K_c = \frac{J}{k_B T_c} = \frac{1}{2}g'(0). \tag{1.31}$$

[4]The plots are taken from [1], page 67.

If $T > T_c$ (i.e., $K < K_c = \frac{1}{2}g'(0)$), as $h^2 \to 0$, P moves towards A, and z_0 tends to a non-zero value determined by $\frac{1}{2}g'(z_0) = K$. Treating the term $\frac{h^2}{4Kz_0}$ as a small perturbation, the equation can be solved iteratively, showing that z_0 is a non-zero even analytic function of h. Consequently, M is an odd analytic function of H near $H = 0$.

For $d \leq 2$, both K_c and $g'(0)$ diverge, so K is always less than K_c. Hence, the spherical model does not exhibit spontaneous magnetisation or phase transitions for $d \leq 2$.

However, for $d > 2$, K_c is finite, allowing $K > K_c$ and, thus, the possibility of $T < T_c$. In this scenario, as $h^2 \to 0$, P moves towards the point $(0, K_c)$, and z_0 approaches zero. Moreover, the right-hand side (RHS) of the equation also approaches K_c, leading to

$$\lim_{h\to 0} \frac{|h|}{z_0} = \sqrt{4K(K - K_c)} \quad \longrightarrow \quad \lim_{h\to 0} M = \operatorname{sgn}(H) M_0, \tag{1.32}$$

where

$$M_0 = \left(1 - \frac{T}{T_c}\right)^{1/2}. \tag{1.33}$$

Thus, $M(H)$ exhibits a jump discontinuity at $H = 0$, corresponding to a non-zero spontaneous magnetisation M_0. Therefore, for $d > 2$, the spherical model demonstrates typical ferromagnetic behaviour with a Curie point given by equation (1.31).

1.2.4 Zero-Field Properties: Exponents $\alpha, \beta, \gamma, \gamma', \delta$

Internal Energy and α

As $H \to 0$, the behaviour of magnetisation M depends on the temperature T: for $T < T_c$, M tends to the spontaneous magnetisation M_0 while for $T > T_c$, M tends to zero and H/M approaches $2Jz_0$ (where $z_0 > 0$).

The internal energy u in these cases is given by:

$$u = u_- = \frac{1}{2}k_B T - Jd, \quad \text{for } T < T_c, \tag{1.34}$$

$$u = u_+ = \frac{1}{2}k_B T - Jd - Jz_0, \quad \text{for } T > T_c. \tag{1.35}$$

The singular part of the internal energy, defined as

$$u_s(0, T) = u_+(0, T) - u_-(0, T), \tag{1.36}$$

vanishes near T_c following a power law. The exponent α is defined by the relation $u_s(0, T) \sim t^{1-\alpha}$ as $t \to 0$, with $t = (T - T_c)/T_c$. For $d \neq 4$, using asymptotic behaviour derived from (1.30) and (1.31), we obtain:

$$\alpha = \frac{4 - d}{d - 2}, \quad \text{for } 2 < d < 4, \tag{1.37}$$

$$\alpha = 0, \quad \text{for } d > 4. \tag{1.38}$$

Spontaneous Magnetization and β

The spontaneous magnetisation M_0 calculated in (1.32) implies that the critical exponent β is

$$\beta = \frac{1}{2}. \tag{1.39}$$

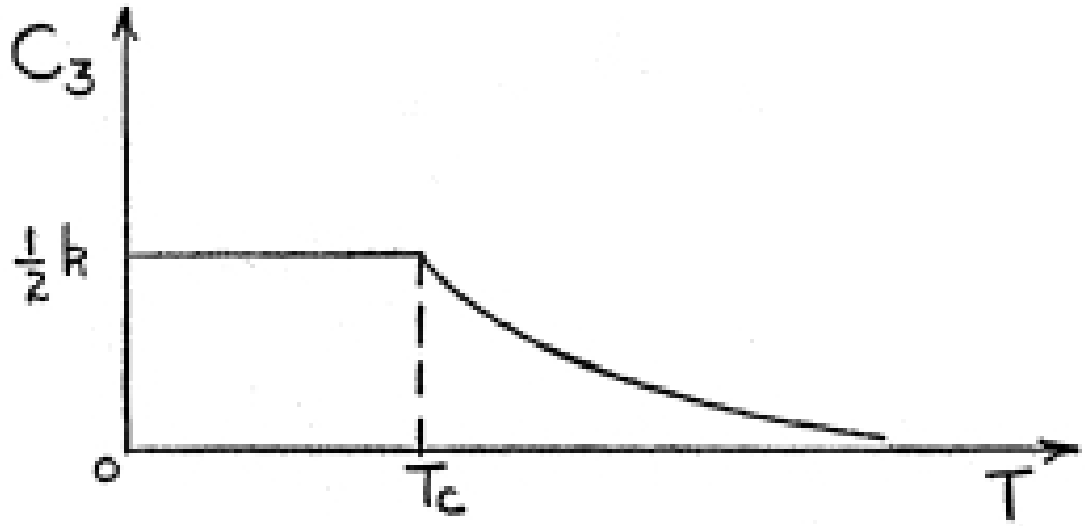

Figure 1.3: The specific heat, normalised by the number of spins, for dimension $d = 3$ is shown in the graph.

Susceptibility and γ, γ'

The susceptibility χ is defined as

$$\chi(H,T) = \frac{\partial M(H,T)}{\partial H}. \tag{1.40}$$

For $H \to 0$ and $T > T_c$, $M \to 0$ and $\chi^{-1} = 2Jz_0$, leading to a divergence of χ as $T \to T_c^+$. The exponent γ is:

$$\gamma = \frac{2}{d-2}, \quad \text{for } 2 < d < 4, \tag{1.41}$$

$$\gamma = 1, \quad \text{for } d > 4. \tag{1.42}$$

For $T < T_c$, $z_0 \to 0$. If $d \leq 4$, $g''(0)$ diverges, and χ diverges for all $T < T_c$, making γ' undefined. For $d > 4$, $g''(0)$ is finite, and

$$\chi^{-1} = -\frac{8JKM_0^2}{g''(0)} \quad \longrightarrow \quad \gamma' = 1 \quad \text{if} \quad d > 4. \tag{1.43}$$

Specific Heat

At zero field, the specific heat remains constant for temperatures lower than the critical temperature for any dimension greater than 2, at a value of $c_0 = \frac{1}{2}Nk_BT$[5], as shown in graph 1.3.

[5]See [2], pages 823-824 where the graph is taken from.

1.2.5 Critical Equation of State

Using equations 1.31 and 1.32 the equation of state is given by:

$$g'(0) - g'(H/2JM) = \frac{2J}{k_B T}(M^2 + t). \tag{1.44}$$

When both H and t are small, the equation simplifies significantly, allowing us to replace T with T_c. Approximating the left-hand side using the expansions from (1.30), we derive the following critical equation of state:

$$H \simeq 2JM\left[\frac{2K_c(M^2+t)}{A_d}\right]^{2/(d-2)}, \qquad \text{for } 2 < d < 4, \tag{1.45}$$

$$H \simeq \frac{4JK_c}{A_4}\frac{M(M^2+t)}{\ln[(M^2+t)^{-1}]}, \qquad \text{for } d = 4, \tag{1.46}$$

$$H \simeq \frac{4JK_c}{A_d}M(M^2+t), \qquad \text{for } d > 4. \tag{1.47}$$

Here, J, K_c, and A_d are constants. This critical equation of state matches the form predicted by the scaling hypothesis:

$$\frac{H}{k_B T_c} = M|M|^{\delta-1}h_s(t|M|^{-1/\beta}), \tag{1.48}$$

where $h_s(x)$ is a dimensionless scaling function. Assuming $h_s(x) = (1+x)^\gamma$, we find consistency with the critical exponents previously calculated: $\beta = \frac{1}{2}$, consistent with spontaneous magnetisation results and the exponent δ is given by:

$$\delta = \frac{d+2}{d-2}, \quad \text{for } 2 < d < 4, \tag{1.49}$$

$$\delta = 3, \quad \text{for } d > 4. \tag{1.50}$$

The scaling hypothesis is verified, and the associated scaling relations are satisfied. Interestingly, the critical exponents exhibit a strong dependence on dimensionality for $2 < d < 4$. In contrast, for $d > 4$, all critical exponents converge to values predicted by the mean-field approximation for the nearest-neighbour Ising model. This confirms that the spherical model provides a smooth interpolation between dimensional-dependent critical behaviour and mean-field universality.

Lastly, we provide the numerical value of the critical temperature T_c for $H = 0$, computed using equation 1.31:

$$T_c = \frac{J}{k_B K_c} = \frac{2J}{k_B g'(0)}. \tag{1.51}$$

Using numerical evaluation of $g'(0)$ in the context of our d-dimensional hyper-cubic lattice, we find:

$$\frac{k_B T_c^3}{J} \approx 4.3, \qquad d = 3, \tag{1.52}$$

$$\frac{k_B T_c^4}{J} \approx 6.5, \qquad d = 4, \tag{1.53}$$

$$\frac{k_B T_c^5}{J} \approx 8.7, \qquad d = 5. \tag{1.54}$$

This value highlights the dependence of the critical temperature on dimensionality and coupling constants, providing a reference for comparing theoretical predictions with numerical simulations.

Chapter 2

The numerical simulation

2.1 Implementation in C++

The spherical model is implemented in C++ using a class named *System*. This class contains all the necessary variables and function definitions to perform three main tasks: creating the system, evolving it, and exporting information about its state at any point in its evolution. These include quantities such as energy, magnetisation, specific heat, and susceptibility.

This design allows the main file to focus solely on initialising the system with the desired parameters and observing its evolution.

2.1.1 Matrix of Interaction

The first task of our `System` class is to create a system represented by a hyper-dimensional lattice. The parameters defining the system are:

- d: the dimension of the system,
- L: the size of the lattice,
- T: the temperature,
- J: the strength of the interaction between spins (in units of $k_B T$, set to 1),
- H: the external magnetic field strength (also in units of $k_B T$),
- θ: a parameter used in the Metropolis algorithm, which will be explained later.

Since we aim to create a class parameterised by the system's dimension, a convenient way to represent the hyper-dimensional lattice follows the approach of Berlin and Kac. They solved the partition function integral by placing all $N = L^d$ spins into a single vector $\bar{\sigma}$. The interaction sum is then computed as the matrix product $\bar{\sigma}^t M \bar{\sigma}$, where the matrix M encodes the interaction contributions.

The quadratic form in equation 1.7 is symmetric and since we're are applying periodic boundaries condition (PBC) it is particularly convenient to write $M = M(c_1, c_2, \ldots, c_N)$ as a cyclic matrix like the following:

$$\begin{pmatrix} c_1 & c_2 & c_3 & \dots & c_{N-1} & c_N \\ c_N & c_1 & c_2 & \dots & c_{N-2} & c_{N-1} \\ c_{N-1} & c_N & c_1 & \dots & c_{N-3} & c_{N-2} \\ . & . & . & \dots & . & . \\ . & . & . & \dots & . & . \\ c_2 & c_3 & c_4 & \dots & c_N & c_1 \end{pmatrix} \tag{2.1}$$

setting $c_{N+i} \equiv c_i$, we see that $M_{i,j} = c_{N-i+1+j}$. M is also symmetric, so we have that $M_{i,j} = M_{j,i}$, or $c_{N-i+1+j} = c_{N-j+1+i}$.
We can determine the values to assign to the matrix M by considering the interaction between the first spin and all other spins. Once this interaction pattern is established, the structure of M becomes evident. At this point, the concept of a cyclic matrix proves particularly useful in capturing the periodic nature of the interactions.

For the **1D case**, we can visualise the system as a string that folds onto itself. In this scenario, the first spin interacts only with the spins at positions 2 and N. Thus, in the first row of the matrix M, we have $c_2 = 1$ and $c_N = 1$, while all other entries are zero.

For the **2D case**, we can visualise a rectangular lattice (composed of n_1 columns and n_2 rows, so $N = n_1 n_2$) with periodic boundary conditions (PBC) by folding it into a torus shape. Let us represent the i-th site with the pair (p, q) and by the spatial coordinates (x, y). If a is the spacing between adjacent lattice points, we set:

$$x = (p-1)a, \qquad p = 1, 2, \dots, n_1, \tag{2.2}$$

$$y = qa, \qquad q = 0, 1, \dots, n_2 - 1, \tag{2.3}$$

$$i = p + qn_1 \in \{1, 2, \dots, N = n_1 n_2\}. \tag{2.4}$$

With this setup, we can examine how the index i changes as we move in the x- or y-direction. Specifically, the site $(x, y) = (0, 0)$ interacts with the sites $(a, 0)$, $(-a, 0)$, $(0, a)$, and $(0, -a)$, which correspond to the pairs (p, q): $(2, 0)$, $(n_1, 0)$, $(1, 1)$, and $(1, n_2 - 1)$, respectively.

Recalling the periodicity of the lattice, where $c_0 = c_N$ and $c_{1-n_1} = c_{N-n_1+1}$, the first row of the interaction matrix M takes the form:

$$c_2 = c_N = c_{n_1+1} = c_{N-n_1+1} = 1, \tag{2.5}$$

with all other entries c_i being zero.

Similarly, we get that for a tridimensional lattice (with sizes n_1, n_3 and n_3):

$$c_2 = c_{n_1+1} = c_{n_1 n_2+1} = c_{N-n_1 n_2+1} = c_{N-n_1+1} = c_N = 1 \tag{2.6}$$

In our case, we simulate only squared hypercubes, so we set $n_1 = n_2 = \cdots = n_d = L$. This allows for a crucial generalisation, enabling us to create the system in any desired dimension. Specifically, we can state that the only c_i coefficients that are non-zero are given by:

$$c_{L^{k-1}+1} = c_{N-L^{k-1}+1} = 1, \qquad k = 1, 2, \dots, d. \tag{2.7}$$

Thus, there will be $2d$ non-zero coefficients in our $N \times N$ interaction matrix.

2.1.2 Adaptive Metropolis Monte Carlo

After initialising the interaction matrix using the armadillo[1] class *SpMat* (class intended for storing large matrices, where most of the elements are zeros), we initialise the system to a random[2] configuration by sampling each spin from a uniform distribution and by normalising it to the spherical constraint: $\sum \sigma_i^2 = N$.

The energy is then computed as

$$E = -J \sum_{\{i,j\}} \sigma_i \sigma_j - H \sum_i \sigma_i = -J \bar{\sigma}^t M \bar{\sigma} - H \sum_i \sigma_i \tag{2.8}$$

where the first sum is over neighbouring spins so, as described above, we use a matrix product with our interaction matrix M, which is optimised with armadillo functions.
While the magnetisation is computed as

$$M = \sum_i \sigma_i \tag{2.9}$$

To evolve the system we use the metropolis algorithm like shown in figure 2.1.

The algorithm starts by initializing the system $\mathbf{s}$ and an initial rotation angle θ_0.
For each Monte Carlo cycle, a new spin configuration $\mathbf{s}'$ is proposed by rotating the original spin configuration $\mathbf{s}$ by an angle θ towards a random vector $\mathbf{u}$ that lies on the hypersphere and is orthogonal to the vector $\mathbf{s}$.
Then the change in energy ΔE is calculated and using the Metropolis acceptance criterion, the proposal is accepted or rejected based on a probabilistic rule: if, taken a random number $r \in [0, 1]$, $r < e^{-\Delta E/kT}$ then the proposal configuration is accepted. This means that a configuration $\mathbf{s}'$ that has a lower energy ($\Delta E < 0$) is always accepted; but if $\mathbf{s}'$ has a higher energy, it can still be accepted with the said probability.

At the beginning we tried to use a more standard method to find a proposal spin configuration, and it was to just change one component of the spin vector, but this meant that, due to the spherical condition 1.6, we had to renormalise the system making this a non-local move and a possible high correlation between the spins.
So we then tried with the different method described before of rotating the vector on the hypersphere. Since we only wanted to have small movements in the space of vectors, we set at the beginning a low value of $\theta = 0.1$rad (or even smaller); this would lead to the stabilisation of the system but sometimes the values found were not correct and slow to reach, leading us to think that the system may got stuck in a local minimum. We then tried with a higher value of θ, but this time the system would fail to reach a stabilisation on a long run.
This lead us to change adaptively the value of θ over the values of Monte Carlo cycles (each cycle runs $N = L^d$ times the Metropolis algorithm) in a way that the system is at first allowed to explore all the space, but then reaches a stabilisation towards the end of the run.[3] So the rotation angle $\theta(t)$ is updated adaptively after each cycle, following the function:

$$\theta(t) = \theta_0 (1 - \frac{t}{N_{MC}})^2, \tag{2.10}$$

where $t = 1, \ldots, N_{MC}$ is the number of Monte Carlo cycle into the run and we chose $\theta_0 = 0.4\text{rad} \simeq 23°$ (see figure 2.2).

[1] See the armadillo library here: [6].
[2] We used the built-in armadillo random generator based on the 64-bit version of the pseudorandom generator Mersenne Twister (*MT19937-64)*).
[3] The more mathematical reason why it works is shown by Heikki Haario [4]

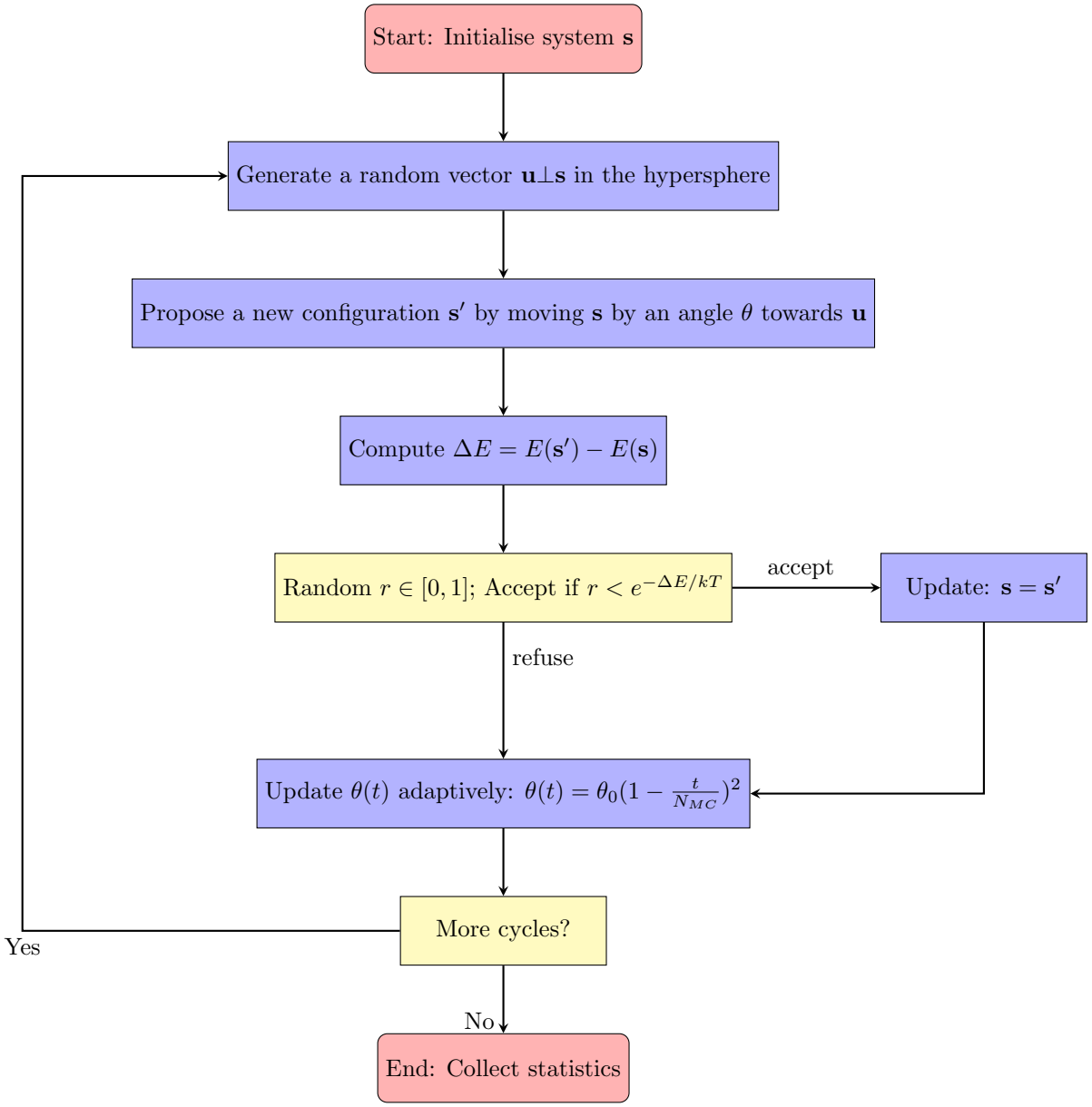

Figure 2.1: Flowchart of the adaptive Metropolis Monte Carlo algorithm for simulating the spherical model.

This method still presents some challenges and requires careful handling.
First, it is crucial to consider only the tail of the recorded values, as the first higher values of θ cause the system to evolve rapidly. Additionally, very small values of θ lead to slow system dynamics, resulting in skewed statistics and unreliable results.

To overcome these issues, we adopt an adaptive approach: during the first half of the simulation, the parameter θ is adjusted dynamically to allow the system to explore the configuration space more effectively, facilitating larger energy jumps. In the second half, we stabilise the system by fixing θ at a low value of 0.1 rad, ensuring convergence to equilibrium.

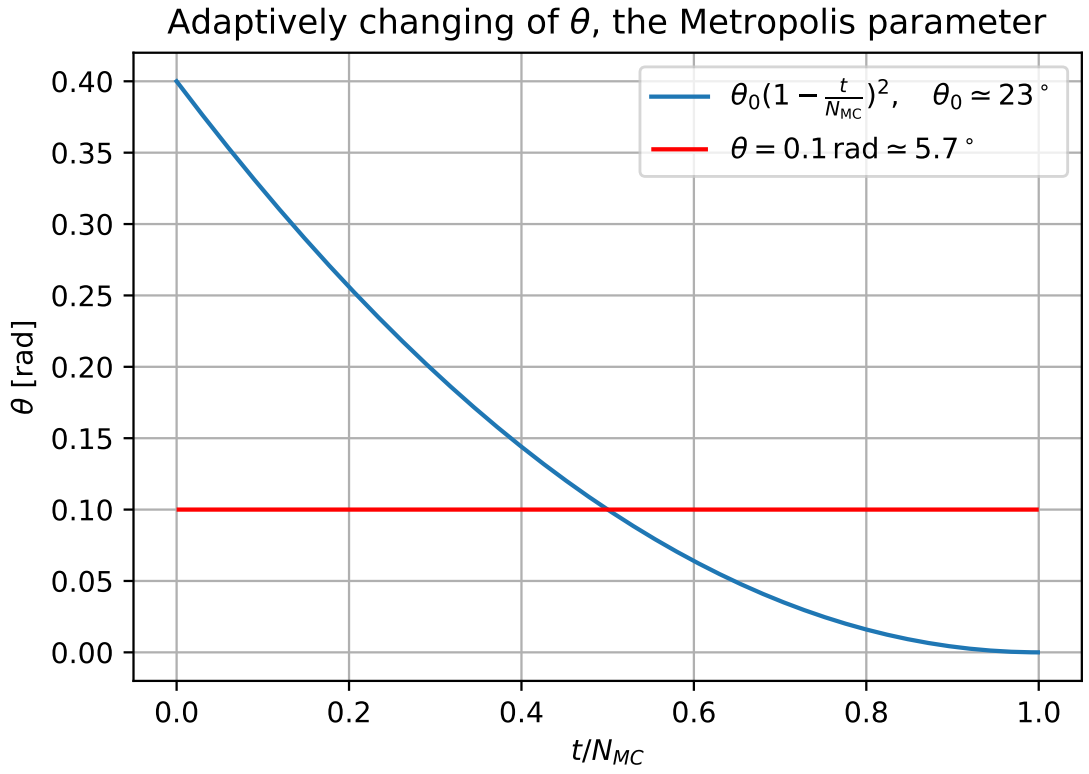

Figure 2.2: The figure shows the two choices of θ, the constant value at $\theta = 0.1\text{rad} \simeq 5.7°$ and the adaptive $\theta = \theta_0(1 - \frac{t}{N_{MC}})^2$ with $\theta_0 = 0.6\text{rad} \simeq 34°$

This final method involves slightly more calculus than the very first one, but it converges faster and more reliably, making it ultimately more efficient and preferable.

Specifically, the first single-spin change method was approximately 30% faster in completing the same number of Monte Carlo cycles. However, it required at least 80'000 cycles to converge, whereas the final method converges in fewer than 30'000 cycles.

It is worth noting that, towards the end, the convergence exhibits exponential decay. We terminate the process once the decay becomes negligible. An alternative approach could involve performing an exponential fit to determine the parameter value as $t \to \infty$.

Finally, we aim to compute and export the values for the specific heat c_V and the susceptibility

χ as given in (2.12):

$$c_v = \frac{\mathrm{Var}[E]}{Nk_BT^2} \quad \text{specific heat per number of spins,} \tag{2.11}$$

$$\chi = \frac{\mathrm{Var}[M]}{Nk_BT} \quad \text{susceptibility per number of spins,} \tag{2.12}$$

For this, we need to collect the values of energy and magnetisation throughout the Monte Carlo cycles and then compute their variance.
As mentioned earlier, the energy and magnetisation values exhibit an exponential decay before stabilising. Therefore, we choose to compute the variance only from the final values, storing the data only after a burn-in period.

2.1.3 Optimisation techniques

When running larger simulations, optimising the code becomes essential. For this reason, we made extensive use of Armadillo's optimised functions wherever possible (for instance, using `arma::norm()` to meet the spherical constraint) and paid close attention to minor details to avoid unnecessary calculations. These small inefficiencies, if not properly addressed, can accumulate and significantly increase the simulation time.

To further improve the simulation's runtime, we parallelised the loop over the temperature steps using *OpenMP* (see [3]). This allowed the machine to compute multiple independent simulations at different temperatures concurrently:

```
#pragma omp parallel for schedule(dynamic)
```

However, we had to be cautious with the handling of the random number generator's seed. Initially, we assigned a different seed to each thread, but this led to non-reproducible results due to the use of `schedule(dynamic)`. As a result, we decided to assign a different seed to each temperature step instead.

Another aspect to be cautious about when parallelising code is memory allocation.
This is a general issue when calling classes or functions multiple times that may allocate variables and data in the same memory space. Therefore, it is crucial to ensure proper memory deallocation for each object that might be reused in the same memory space.
For instance, to compute the specific heat and susceptibility as 2.12 we need to store the values of energy E and magnetisation M over the course of Monte Carlo cycles and then compute their variance. To achieve this, we used the functionality `track_E.push_back(E)` from the `std::vector<double>` class. However, this can lead to issues if the memory is not cleared once it is no longer needed, which can be done using `track_E.clear()`.
Another more obvious example is the use of static variables, which can be very useful for computations that need to be performed only once per temperature. However, when dealing with multiple temperature steps, caution is required, as static variables persist across function calls and may lead to unintended side effects if not managed properly.

2.2 Simulation setup and parameters

The first objective of this simulation is to identify phase transition behaviour as predicted by theory and to determine the expected critical temperatures for various dimensions.

We will simulate systems of different sizes and dimensions across a range of temperatures. Table 2.1 summarises the simulation parameters.

symbol	**variable type**	**description**
d	$int > 0$	dimension of the system
L	$int > 0$	leght of the system ($N = L^d$)
T_i	$double > 0$	initial temperature of the simulation
T_f	$double > 0$	final temperature of the simulation
J	$double$	Energy strength of interaction
H	$double$	Magnetic field
n	$int > 0$	number of temperature steps between T_i and T_f
θ	$double$	metropolis' adaptive variable, in radians
n_MC	$int > 0$	number of Markov Chain Monte Carlo cycles
burnin	$double \in [0, 1]$	percentage of burn-in time
out_num	int	number to change the name of the output file

Table 2.1: The table lists all the parameters required for the program to function. These parameters must be stored in an external file and provided as input to the program.

In Table 2.1, the first six parameters (d, L, T_i, T_f, J, and H) are used to initialise the system, while the remaining parameters configure the simulation and influence the system's evolution. The exception is the final parameter, `out_num`, which formats the output file.

While, in principle, any values could be chosen for these parameters, practical constraints limit our choices. The primary challenge is that this simulation is computationally intensive.

In contrast to the classical Ising model, where each spin takes discrete values, in the spherical model each spin takes a continuous real value. Consequently, there are no pre-computation shortcuts for the energy difference between the current and proposed configurations. Instead, the system's energy must be recalculated N times per Monte Carlo cycle, involving costly matrix operations.
Additionally, the system must satisfy the spherical constraint (1.6). Every time the spin vector's length changes to generate a new proposal, the vector must be renormalised to meet this constraint.

Another significant consideration is that, while the *2D* case is of interest in the Ising model, the critical temperature for the spherical model in *2D* is zero. Consequently, we focus on higher dimensions, such as *3D*, *4D*, and *5D*. However, as expected, the simulation time grows exponentially with increasing system size and dimensionality, making it impractical to simulate a sufficiently large system.

With our system specifications, detailed in Table 2.2, we are only able to simulate the evolution for $d = 3$ on a hyper-lattice of maximum size $L = 15$, with at least 8 temperature steps and 30'000 Monte Carlo cycles per step. For $d = 5$, however, a single temperature step for a lattice of size $L = 5$ takes over 8 hours, making such simulations unprocessable.

Component	Specifications
CPU	Intel Core i7-6600U @ 2.60 GHz, 2 Cores / 4 Threads, 4MB Cache
RAM	3.8 GiB (Available: 3.1 GiB)
Storage	1 TB Disk
GPU	Intel HD Graphics 520 (Driver Version: 31.0.101.2111)
GPU Memory	3.9 GiB (Shared)

Table 2.2: System Specifications

Chapter 3

Results

3.1 Algorithm convergence

The first point we want to discuss is how the different Metropolis approaches, as discussed in Section 2.1.2, converge differently.

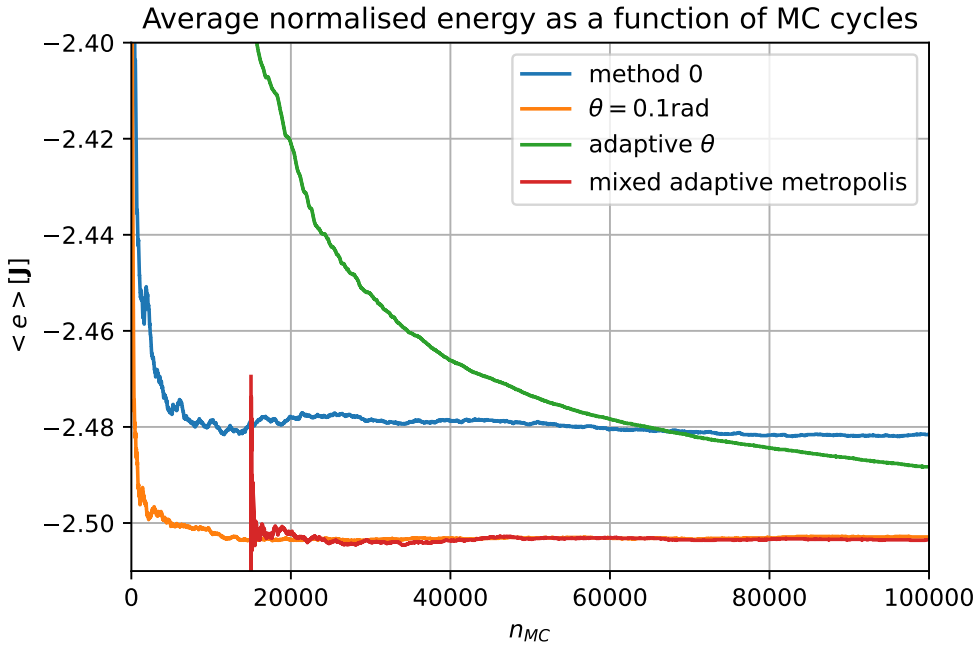

Figure 3.1: This plot shows the energy per spin as a function of Monte Carlo (MC) cycles. The system under consideration is a 3-dimensional hyper-lattice of size $L = 5$, with a fixed temperature of $T = 1$. The energy values are normalised by the number of spins, N, to facilitate comparison. This plot highlights how each method approaches equilibrium and the relative speed of convergence.

In figure 3.1, we observe that the methods approach equilibrium in different ways.
The first observation is that they do not converge to exactly the same equilibrium energy. This discrepancy could arise from the fact that different ways to propose the new configuration might sample from a different probability distribution. Alternatively, it might indicate that the system is trapped in a local minimum. However, the difference in equilibrium energy between the methods is less than 1% but we still note that for $d = 3$ and $T = 1$, from 1.35 the correct value at zero field is $e = \frac{E}{N} = -2.5J$.

Another key aspect is the convergence speed. The green curve appears not to converge within the simulation time. This behaviour can be attributed to two factors: first, the adaptive parameter θ continuously evolves throughout the simulation according to Equation 2.10, so it stays high for a long period. Second, the statistical average is computed using all energy values from the start of the simulation, including non-equilibrium values and, since with this method, non-equilibrium values can differ a lot from the desired value and can also be the majority of the data, they have a big impact.

A more effective approach with the adaptive Metropolis method involves computing statistics only after θ has stabilised at a low value. The red curve illustrates this mixed strategy: for the first 15'000 MC cycles, θ evolves according to Equation 2.10 (with $N_{\text{MC}} = 30'000$). After 15'000 cycles, θ has decreased to 0.1, at which point we begin averaging the energy. As a result, the red curve converges to the same equilibrium energy as the orange curve.

We can thus conclude that the standard Metropolis method requires at least 80'000 MC cycles to reach a thermalised value, while the θ-method achieves this in only 30'000 cycles.

For the mixed adaptive method, the required number of cycles is even lower. Specifically, thermalisation can be achieved in 15'000 cycles, where the parameter θ is adjusted adaptively for the first 7,500 cycles and then held constant for the remainder. Since the adaptive θ-method allows the system to explore the configuration space more efficiently, it reaches the same equilibrium results as those obtained with a much higher number of cycles in the constant θ-method.

3.2 Energy and Magnetisation

For $H = 0$, the internal energy is determined by equation 1.35.

The magnetization, on the other hand, follows equation 1.32 for temperatures $T < T_C$, while it should vanish for $T > T_C$.

In the energy graph (3.1), we can see how the system perfectly follows the theoretical line $\frac{T}{2} - d$ before the critical temperature. At the critical temperature, the only system that deviates slightly from the line is the one with dimension 5.
We could also follow the behaviour after the critical temperature and compute the critical exponent α from 1.36.

From graph 3.3, we observe that the systems still follow the line $(1 - \frac{T}{T_C})^{1/2}$ for $T < T_C$, with the line for $d = 5$ being the closest. An interesting aspect is that the magnetisation does not seem to approach zero as expected, but this may simply be due to the fact that higher temperatures need to be reached for a clearer convergence; or maybe because the systems were too small.

3.3 Specific Heat and Susceptivity

Figure 3.4 displays the normalised specific heat for four different systems. Here, the critical temperatures of each system begin to emerge more distinctly. Clearly, all systems exhibit a lower critical temperature than theoretically expected, which can be attributed to the small size of the simulated systems.

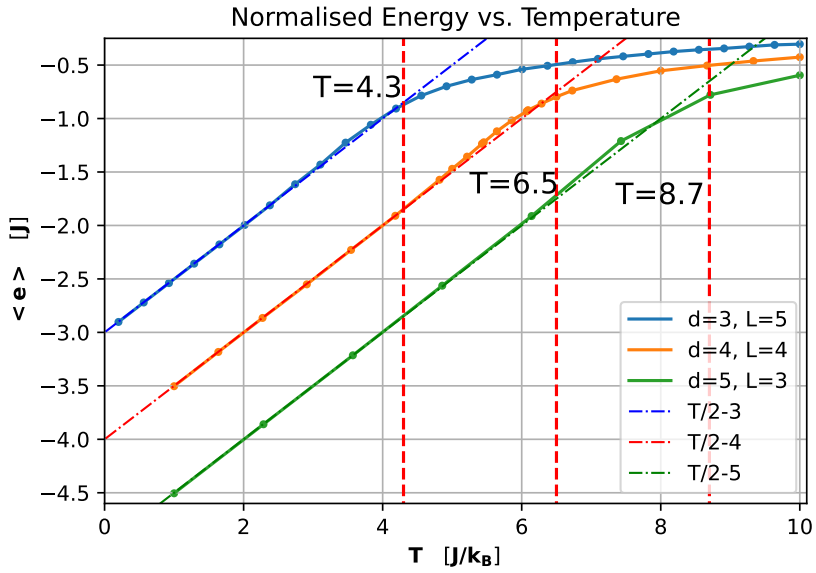

Figure 3.2: The graph illustrates the internal energy, normalised by the total number of spins, as a function of temperature for systems in dimensions ranging from 3 to 5. Additionally, it shows the theoretical behaviour given by equation (1.35) for $T < T_C$ and indicates the expected critical temperature values.

This effect is more evident in the $d = 3$ case, where two simulations were performed with $L = 5$ and $L = 10$. The larger system, $L = 10$, aligns more closely with the theoretical critical temperature.[1] Finally, we examine the susceptibility graph in figure 3.5, which underscores the significance of larger system sizes. As the temperature approaches T_C, the susceptibility χ should diverge. This divergence is more pronounced in the larger ($d = 3$, $L = 10$) system. Additionally, the peak of the black curve, corresponding to the larger system, approaches the theoretical critical temperature $T_C = 4.3$ more accurately.

[1] This was not highlighted in the previous sections as the values for both simulations were nearly overlapping.

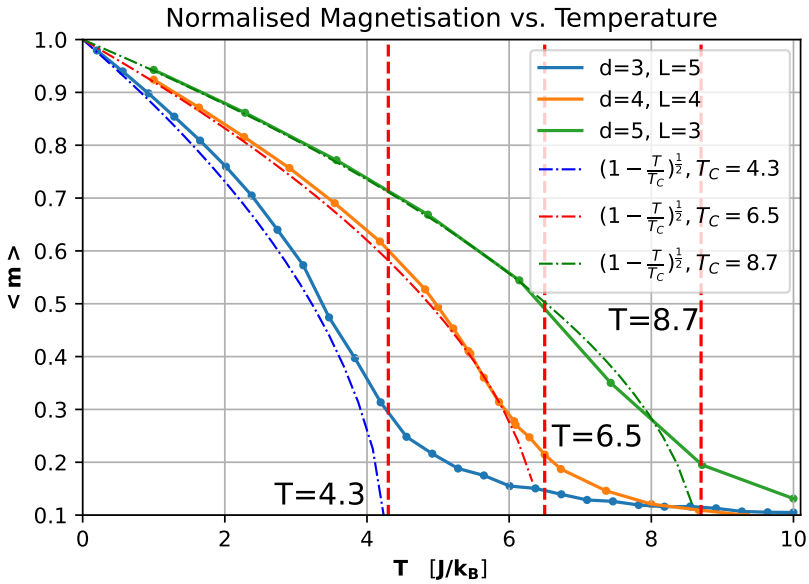

Figure 3.3: The graph illustrates the magnetisation, normalised by the total number of spins, as a function of temperature for systems in dimensions ranging from 3 to 5. Additionally, it shows the theoretical behaviour given by equation (1.32) for $T < T_C$ and indicates the expected critical temperature values.

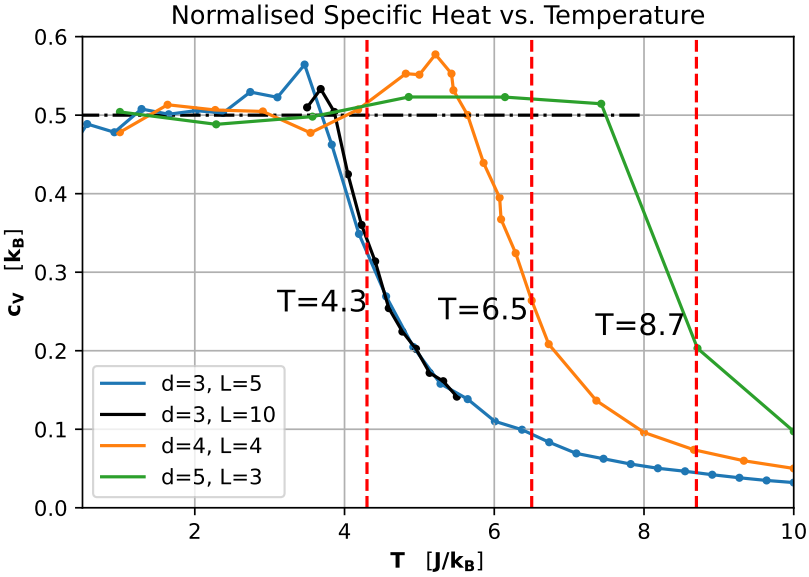

Figure 3.4: The graph illustrates the specific heat c_V, normalised by the total number of spins, as a function of temperature for systems in dimensions ranging from 3 to 5. Additionally, it shows the theoretical constant value $c_V = \frac{1}{2}k_B$, for $T < T_C$ and indicates the expected critical temperature values.

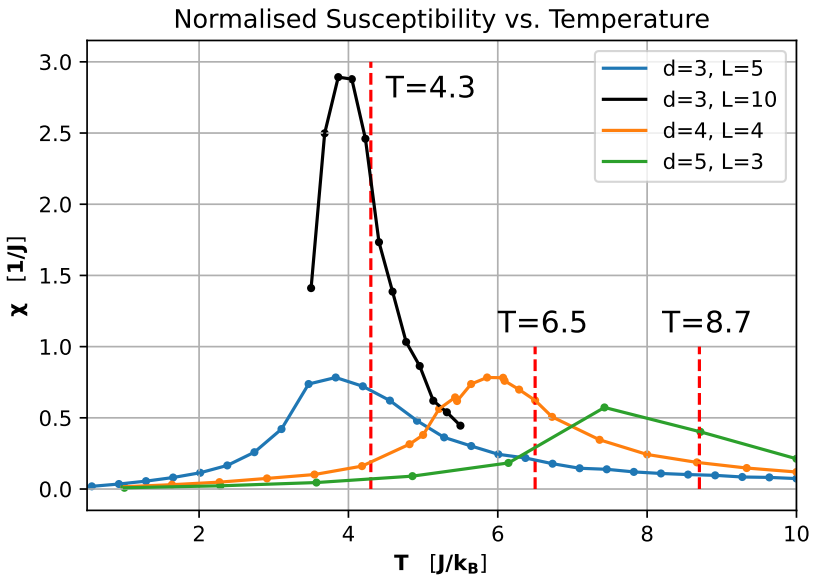

Figure 3.5: The graph illustrates the susceptivity χ, normalised by the total number of spins, as a function of temperature for systems in dimensions ranging from 3 to 5. Additionally, it indicates the expected critical temperature values.

Conclusions

In conclusion, we can say that the algorithmic analysis reveals different convergence behaviours among the different Metropolis approaches. Notably, the adaptive methods, particularly the mixed θ-strategy, significantly enhance convergence speed compared to the standard Metropolis method. As illustrated in figure 3.1, the θ-method achieves thermalisation in roughly one-third the cycles required by the conventional method, underscoring its efficiency in exploring the configuration space. Moreover, the mixed strategy further optimises convergence by stabilising the parameter θ at an earlier stage, highlighting its practical advantage in computational simulations.

The investigation into thermodynamic quantities meets theoretical predictions. The internal energy per spin adheres closely to the theoretical line $\frac{T}{2} - d$ for $T < T_C$, as seen in figure 3.2. However, deviations at critical temperatures suggest the influence of finite-size effects, particularly in higher-dimensional systems. Similarly, the magnetisation curves (figure 3.3) align with the theoretical prediction $(1-T/T_C)^{1/2}$ below the critical temperature. The observed persistence of non-zero magnetisation at higher temperatures likely stems from either the finite system size or insufficient sampling in this temperature range.

The analysis of specific heat (figure 3.4) and susceptibility (figure 3.5) further proves these findings. The shift in critical temperature observed in specific heat plots accentuates the role of system size, with larger systems demonstrating better alignment with theoretical T_C values. This trend is the most evident in the susceptibility plots, where larger systems exhibit sharper peaks near T_C, reflecting the expected divergence behaviour.

This work provides a strong foundation for further research into the spherical model. Future studies could delve deeper into the critical phenomena by investigating critical exponents, autocorrelation functions, and the system's ferromagnetic response to variations in the external magnetic field. These directions could offer a better understanding of the spherical model's behaviour and extend its applicability to broader contexts in statistical mechanics.

In conclusion, this work has demonstrated the utility of adaptive Metropolis algorithms in accelerating convergence within the spherical model framework. The thermodynamic properties evaluated align well with theoretical expectations, although with minor discrepancies attributable to finite-size effects. These results affirm the spherical model's capacity to capture critical phenomena while underscoring the importance of system size and algorithmic efficiency in numerical simulations.

Bibliography

[1] R. J. Baxter. *Exactly solved models in statistical mechanics*. 1982.

[2] T. H. BERLIN and M. KAC. "The spherical model of a ferromagnet". In: *Physical Review, 86, 821* (15 June 1952).

[3] Rohit Chandra et al. *Parallel programming in OpenMP*. Morgan kaufmann, 2001.

[4] Heikki Haario, Eero Saksman, and Johanna Tamminen. "An adaptive Metropolis algorithm". In: *Bernoulli* 7.2 (2001), pp. 223–242.

[5] R.K. Pathria and Paul D. Beale. *Statistical mechanics*. Elsevier BH, Amsterdam, 2011.

[6] Conrad Sanderson and Ryan Curtin. "Practical Sparse Matrices in C++ with Hybrid Storage and Template-Based Expression Optimisation." In: *Journal of Open Source Software, Vol. 1, No. 2, pp. 26* (2016).

[7] H. E. Stanley. "Spherical Model as the Limit of Infinite Spin Dimensionality". In: *Phys. Rev.* 176 (2 Dec. 1968), pp. 718–722.