Delft University of Technology

Using the Random-Cluster Model for Ising Model Simulations

Author:
Maxime Delboo

Supervisor:
Dr. E. Pulvirenti
Dr.ir. J.M. Thijssen

4th October 2022

Abstract

This thesis contains a rigorous derivation of the path integral formulation of the Ising model with multiple original proofs. Besides that, thesis also contains various results of simulations of the 2D square lattice Ising Model with nearest-neighbour interactions using the Swendsen-Wang algorithm. Using finite size scaling to find critical exponent γ, we reported a value of $\gamma = 1.748 \pm 0.004$. After calculating the relaxation times τ for various thermodynamic variables, we found the value for the dynamic critical exponent z to be in the range of $z = 0.180 \pm 0.004$ and $z = 0.282 \pm 0.005$.

Contents

1 Introduction

The Ising model is a model in statistical mechanics that displays a lot of features. It is a very simple model that can explain a plethora of real world behaviours, due to its universal nature. Historically, its main application is its accurate description of ferromagnetism, which was also the motivation for its invention by physicist Wilhelm Lenz, together with his student Ernst Ising, in 1920. But since then, many more applications have been discovered. For example, Stauffer [13] reported about the various uses of the Ising model in social economic contexts, which include urban segregation, language change, and economic opinions. Although it seems very strange that these topics have anything to do with magnetism, they are unified by the Ising model.

The Ising model works in a very simple manner. Consider a certain configuration of nodes, which each have an attribute which we call spin, which is restricted to being either up or down. We let each of these points interact with its neighbours in a way that each point prefers to have the same spin as its neighbour. This is because equal neighbours cause the overall energy to go down and the system prefers to have low energy. However, There is an overarching factor that disturbs this tendency, which is temperature. High temperatures trigger disorder, while low temperatures allow the nodes to do what they prefer most, which is copying their neighbours. The behaviour that makes the Ising model stand out most, which also happens what makes it so universal, is its second order phase transition, which occurs at the critical temperature. It is the point where the entropy caused by heat is exactly balanced with the nodes' tendency to align. The symmetry-breaking behaviour that this phase transitions brings is of great interest.

This phase transition has been studied to great detail with different analytical and computational techniques. We will focus on computational techniques for simulations of the Ising model. The usual method for doing Ising model simulations is to use the Metropolis-Hastings algorithm, which flips single spins based on whether it will lower the total energy. It is a generally fast technique, however it suffers from a slow-down around the critical range, that becomes worse with increasing size. This is an important hurdle to overcome, as the critical range is of great interest.

To do this, we need to look at the Ising model from another angle. We can reformulate the Ising model in the language of quantum mechanics, which will allow us to develop a powerful theoretical tool in the form of a path integral representation. This representation gives rise to many reformulations, but we will focus of what is called Fortuin-Kasteleyn representation or the random cluster representation. This reformulation of the Ising model looks at the behaviour of clusters instead of singular spins, which leads to a very powerful algorithm which is way better for dealing with critical behaviour.

In section 2, we will develop the theory of the Ising model and derive its path integral representation along with the theory of random cluster models. In section 3, we will look at different computational techniques to simulate the Ising model. Furthermore, we will also look into how we can use this algorithm to simulate the critical behaviour of the Ising model, using the magnetic susceptibility χ. Lastly, we will investigate how we can

measure the critical slowing-down of the Ising model by calculating critical exponent z. In section 4, we will look at the results of these simulations and draw some conclusions.

2 Theory

2.1 Ising model

This paper is oriented around a specific way of looking at the Ising model. But before we try to understand that, we start by looking at it in the original way. Firstly, we discuss some history about it, then we look at what it does intuitively and finally we go over the mathematics.

2.1.1 Introduction to the Ising model

2.1.2 Classical definition

The mathematical formulation of the model is quite simple as stated before. In Definition 2.1 [6, pp. 88–89], you can find the notation of the model and the definition of the functions that make it up. We will use this notation in the rest of the paper.

Definition 2.1. The model is formulated in terms of a couple of things. Firstly, an undirected graph $(\Lambda, \mathcal{E})$ with unoriented edges $e = \{i, j\} \in \mathcal{E}$ with $i, j \in \mathcal{E}$. Next, each edge has a positive coupling constant $J_{ij} \geqslant 0$ associated with it. The set of all coupling constants is called $\mathbf{J}$. The constant $h \in \mathbb{R}$ represents a magnetic field. Each node $\nu_i \in \{-1, 1\}$ has a spin of either -1 or 1. ν is any spin configuration on Λ, or $\nu \in \Omega_\Lambda = \{-1, 1\}^\Lambda$. The Hamiltonian of the model $\mathbf{H}_\Lambda(\nu)$ is a function on Ω^Λ, where

$$\mathbf{H}_\Lambda(\nu) = -\sum_{(i,j)\in\mathcal{E}} J_{ij}\nu_i\nu_j - h\sum_{i\in\Lambda}\nu_i.$$

Definition 2.2. We define the configuration probability $\mu_\Lambda^{\beta,h}$ on Ω_Λ, where $\beta \geqslant 0$ is the *inverse temperature* and is assumed given, as

$$\mu_\Lambda^{\beta,h} = \frac{1}{\mathcal{Z}_\Lambda(\beta, h)} e^{-\beta\mathbf{H}_\Lambda(\nu)}$$

where $\mathcal{Z}_\Lambda(\beta, h)$ is the partition function, defined as

$$\mathcal{Z}_\Lambda(\beta, h) = \sum_{\nu\in\Omega_\Lambda} e^{-\beta\mathbf{H}_\Lambda(\nu)}. \tag{1}$$

As you might have noticed, this probability distribution is what is generally called the Boltzmann distribution, with the Hamiltonian that is specific to the Ising model. This Hamiltonian comprises of two terms. The first determines the influence of the nodes on each other, and since J_{ij} is always positive, the Hamiltonian always decreases or stays the same if two nodes have the same spin. This is what induces the ferromagnetic behaviour. Should J_{ij} be negative (a case that we excluded in our definition), the Hamiltonian would encourage anti-ferromagnetic behaviour. The second term of the Hamiltonian influences each node to favour one specific direction: if $h > 0$, then $\nu_i = 1$ is preferred, and if $h < 0$, then $\nu_i = -1$ is preferred. The role of h plays the role an external magnetic field would play for the paramagnetic materials. $h = 0$ would mean that there is no external

magnetic field present. It should now be clear why this is good choice of Hamiltonian.

Lastly, we define $\mu_\Lambda^{\beta,h}(\bullet)$ to be the expected value of $\bullet$ under $\mu_\Lambda^{\beta,h}$. In particular,

$$\mu_\Lambda^{\beta,h}(x) = \frac{1}{\mathcal{Z}_\Lambda(\beta,h)} \sum_{\nu\in\Omega_\Lambda} x e^{-\beta \mathbf{H}_\Lambda(\nu)}. \tag{2}$$

2.1.3 Quantum definition

We now want to redefine this model. This means that physically it will describe the same things, but its new mathematical context will allow us to derive what we want in a more straight forward way. In particular, we redefine our model as a quantum system. A good place to start with this, is to create a space where each observable state corresponds to an eigenfunction of the system, as should be in a quantum system. This could be done in many ways, but one of the simplest is to transform each classical ν to an orthonormal basis where each and every classical state corresponds to one eigenfunction of a vector space. Seeing as we had $2^{|\Lambda|}$ possible states before, we now need a $2^{|\Lambda|}$-dimensional vector space.

We start by introducing the spin values ± 1 as the eigenvalues of the Pauli matrix

$$\hat{\sigma}^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{3}$$

with corresponding eigenfunctions

$$\psi_{+1} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \psi_{-1} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{4}$$

We now define the space we want to work on:

$$\mathbb{X}_\Lambda = \bigotimes_{i\in\Lambda} \mathbb{R}^2,$$

which is indeed a $2^{|\Lambda|}$-dimensional vector space. We now 'lift' the classical configurations $\nu \in \Omega_\Lambda$ to this space with the eigenfunctions from (4) as

$$\Psi_\nu = \bigotimes_{i\in\Lambda} \psi_{\nu_i}. \tag{5}$$

The collection of Ψ_ν now forms an orthonormal basis of $\mathbb{X}_\Lambda$, with respect to the scalar product we define as

$$\langle \psi_\nu | \psi_{\nu'} \rangle = \prod_{i\in\Lambda} \langle \psi_{\nu_i} | \psi_{\nu_i'} \rangle_2, \tag{6}$$

with $\langle \bullet | \bullet \rangle_2$ the standard inner product on $\mathbb{R}^2$ and $\nu, \nu' \in \Omega_\Lambda$. Next, we define a set of linear self-adjoint operators $\hat{\sigma}_i^z$ for each i that acts on any ψ_ν just like the Pauli matrix $\hat{\sigma}^z$ would act on the i-th component of that ψ_ν, that is, it gives us the spin of that component. In mathematical terms:

$$\hat{\sigma}_i^z \Psi_\nu = \psi_{\nu_1} \otimes ... \otimes \hat{\sigma}^z \psi_{\nu_i} \otimes ... = \nu_i \Psi_\nu. \tag{7}$$

Furthermore, multiple of these operators can be applied without affecting each other. This can be seen by considering

$$\hat{\sigma}_i^z \hat{\sigma}_j^z \Psi_\nu = \hat{\sigma}_i^z \nu_j \Psi_\nu = \nu_j \hat{\sigma}_i^z \Psi_\nu = \nu_j \nu_i \Psi_\nu = \nu_i \nu_j \Psi_\nu, \tag{8}$$

which holds for any pair $\hat{\sigma}_i^z, \hat{\sigma}_j^z$. It is also clear for this that the operators all commute, seeing as the application of one of them only acts upon a certain component, producing an eigenvalue that can then be extracted by linearity. They influence do not influence each other and can be applied on any number of components in any order.

We now have the ingredients necessary to rewrite the Ising Hamiltonian in the 'quantum way', which we will denote with $\mathcal{H}_\Lambda$. It is defined as

$$\mathcal{H}_\Lambda = -\sum_{(i,j)\in\mathcal{E}} J_{ij} \hat{\sigma}_i^z \hat{\sigma}_j^z - h \sum_{i\in\mathcal{E}} \hat{\sigma}_i^z. \tag{9}$$

Using (7) and (8) it is easy to see that

$$\mathcal{H}_\Lambda \Psi_\nu = \mathbf{H}_\Lambda(\nu) \Psi_\nu,$$

which holds for all configurations ν. This means that $\mathcal{H}_\Lambda$ is diagonal with each element of the diagonal being equal to value of the Ising Hamiltonian for the corresponding classical configuration. We can now rewrite the final quantities we defined in the previous chapter, starting with the partition function $\mathcal{Z}_\Lambda(\beta, h)$, defined in (1). We now write it as

$$\mathcal{Z}_\Lambda(\beta, h) = \sum_{\nu\in\Omega_\Lambda} e^{-\beta \mathbf{H}_\Lambda(\nu)} = \sum_{\nu\in\Omega_\Lambda} \langle \Psi_\nu | e^{-\beta\mathcal{H}_\Lambda} | \Psi_\nu \rangle = \mathrm{Tr}(e^{-\beta\mathcal{H}_\Lambda}), \tag{10}$$

where the middle equality holds because $\mathcal{H}_\Lambda$ is diagonal in the chosen basis. Lastly we can rewrite equation 2 in terms of traces, so that

$$\mu_\Lambda^{\beta,h}(A) = \frac{\mathrm{Tr}(A e^{-\beta\mathcal{H}_\Lambda})}{\mathrm{Tr}(e^{-\beta\mathcal{H}_\Lambda})}, \tag{11}$$

where A is any self-adjoint matrix.

2.2 Poisson limits

This section is dedicated towards developing a way to write the Ising Hamiltonian in a path integral representation by using Poisson limits. We will do this first with a general setup and later apply it to the Ising Hamiltonian. Our general context is as follows:

- $\mathbb{X}$ is an m-dimensional vector space over $\mathbb{R}$ with the scalar product defined in (6) and an orthonormal basis $\{\Psi_i\}$.
- K_1,...,K_m are bounded self-adjoint operators on $\mathbb{X}$.
- λ_1,...,λ_m are positive numbers.

Next, we want to show that we can linearize the Hamiltonian $e^{\beta \sum_{l=1}^{m} \lambda_i K_i}$ in the general case, given $\beta > 0$. In particular, we want

$$e^{\beta \sum_{l=1}^{m} \lambda_l K_l} = e^{\beta \sum \lambda_l} \lim_{\Delta \to 0} \left(\prod_{l=1}^{m} \{(1 - \Delta\lambda_l) I + \Delta\lambda_l K_l\} \right)^{\beta/\Delta}, \tag{12}$$

where I is the identity matrix. For this, we need multiple ingredients. We start by proving the Lie-Trotter formula, similar as in [11] where it is proven for $n = 2$.

Theorem 2.3 (Lie-Trotter product formula)**.** *Let $A_1, A_2, ..., A_k$ be bounded, self-adjoint matrices with $k \in \mathbb{N}$. Then*

$$e^{\sum_{i=1}^{k} A_i} = \lim_{n \to \infty} \left(\prod_{i=1}^{k} e^{A_i/n} \right)^n. \tag{13}$$

The proof for Theorem 2.3 is provided in Appendix A. To show that the linearization in (12) holds, we need to prove one final theorem.

Theorem 2.4. *Let $A_1, ..., A_n$ be bounded self-adjoint operators. Then*

$$\mathrm{e}^{\sum_{i=1}^{n} A_i} = \lim_{\Delta \to 0} \left(\prod_{i=1}^{n} (I + \Delta A_i) \right)^{1/\Delta}$$

Proof. We start by applying the Lie-Trotter formula from Theorem 2.3 as follows:

$$\mathrm{e}^{\sum_{i=1}^{n} A_i} = \lim_{\Delta \to 0} \left(\prod_{i=1}^{n} \mathrm{e}^{\Delta A_i} \right)^{1/\Delta}.$$

Note that we assume w.l.o.g. that $1/\Delta$ is always an integer as $\Delta \to 0$. We now apply the matrix exponential definition to get

$$\lim_{\Delta \to 0} \left(\prod_{i=1}^{n} \mathrm{e}^{\Delta A_i} \right)^{1/\Delta} = \lim_{\Delta \to 0} \left(\prod_{i=1}^{n} [I + \Delta A_i + B_i] \right)^{1/\Delta},$$

where B_i is a rest term that represents all the other terms in the sum, and it satisfies $\|B_i\| = \mathcal{O}(\Delta^2)$. Now, let $T_i = I + \Delta A_i$ so that

$$\lim_{\Delta \to 0} \left(\prod_{i=1}^{n} [I + \Delta A_i + B_i] \right)^{1/\Delta} = \lim_{\Delta \to 0} \left(\prod_{i=1}^{n} [T_i + B_i] \right)^{1/\Delta}$$

If we now work out the product inside the brackets, it is easy to see that one term will depend on only T_i's and all other terms will have at least one factor with some B_i's in it. These B_i's have a norm of order $\mathcal{O}(\Delta^2)$, so all these terms will be of order $\mathcal{O}(\Delta^2)$ or higher. Since the sum of a finite number of terms of order $\mathcal{O}(\Delta^2)$ is still of order $\mathcal{O}(\Delta^2)$, we can rewrite the entire product as

$$\lim_{\Delta \to 0} \left(\prod_{i=1}^{n} [T_i] + B(A_1, ..., A_n) \right)^{1/\Delta},$$

where $B(A_1, ..., A_n) = B$ is a matrix that is a function of the A_i's and has norm of order $\mathcal{O}(\Delta^2)$. We now almost have what we want, we only need to control the B term. In order to do this, we consider

$$\lim_{\Delta \to 0} \left\| \left(\prod_{i=1}^{n} [T_i] + B \right)^{1/\Delta} - \left(\prod_{i=1}^{n} T_i \right)^{1/\Delta} \right\|.$$

If we can show that norm of the difference of these two terms goes to zero, the theorem is proven. To show that this norm goes to zero, we first take $\prod_{i=1}^{n} [T_i] = Q$. Furthermore, remember that $1/\Delta$ is always an integer. We now substitute $k = 1/\Delta$ to avoid confusion in the powers so that we now have the following limit:

$$\lim_{k \to \infty} \left\| (Q + B)^k - Q^k \right\|.$$

Remember that Q and B do not necessarily commute, so we cannot apply the binomial formula to $(Q + B)^k$. Instead we write out the terms one by one as

$$(Q + B)^k = Q^k + Q^{k-1}B + Q^{k-2}BQ + ...$$

We go on to take every permutation of k Q's and B's. Now in the norm, the first and the last term cancel, so we only have the following terms left:

$$\lim_{k \to \infty} \left\| Q^{k-1}B + Q^{k-2}BQ + ... \right\|.$$

Now we can apply the sub-multiplicativity of the matrix norm and the triangle inequality to distribute the norm over every term and product in the term, where we also use that $\|B\| = \mathcal{O}(1/k^2)$, as well as the fact that the '...' contains all permutations of Q's and B's to get that:

$$\begin{aligned}
\lim_{k \to \infty} \left\| Q^{k-1}B + ... \right\| &\leqslant \lim_{k \to \infty} \binom{k}{1} \|Q\|^{k-1} \mathcal{O}(1/k^2) + \binom{k}{2} \|Q\|^{k-2} \mathcal{O}(1/k^{2 \cdot 2}) \| + ... \\
&= \lim_{k \to \infty} \sum_{i=1}^{k} \binom{k}{i} \|Q\|^{k-i} \mathcal{O}(1/k^{2i}) \\
&\leqslant \lim_{k \to \infty} \sum_{i=1}^{k} k^i \|Q\|^{k-i} \mathcal{O}(1/k^{2i}) \\
&= \lim_{k \to \infty} \sum_{i=1}^{k} \|Q\|^{k-i} \mathcal{O}(1/k^i) \\
&= \lim_{k \to \infty} \|Q\|^{k-1} \mathcal{O}(1/k)
\end{aligned}$$

To see that this limit converges to 0, we bound $\|Q\|^{k-1}$. Remember that $Q = \prod_{i=1}^{n} (I + \Delta A_i)$. Writing out this product gives us:

$$\prod_{i=1}^{n} I + \Delta A_i = I + \Delta A_1 + \Delta A_2 + ... + \Delta^2 A_1 A_2 + \Delta^2 A_1 A_3 + ... = I + \Delta S + R$$

where R is the sum of all terms of second order in Δ or higher, so that $\|R\| = \mathcal{O}(1/k^2)$. Recall that $\Delta = 1/k$, so that $Q = I + S/k + \mathcal{O}(1/k^2)$. Now,

$$\lim_{k\to\infty} \|I + S/k + R\|^{k-1} = \lim_{k\to\infty} \frac{\|I + S/k + R\|^k}{\|I + S/k + R\|} \leqslant \lim_{k\to\infty} \frac{(1 + \|S\|/k + \mathcal{O}(1/k^2))^k}{1} = \mathrm{e}^{\|S\|},$$

so that $\|Q\|^{k-1}$ is bounded and $\lim_{k\to\infty} \|Q\|^{k-1}\mathcal{O}(1/k) = 0$. This proves the theorem.

□

We now apply Theorem 2.4 with $A_i = \lambda_i(K_i - I)$ to get that

$$\begin{aligned} e^{\beta \sum_{l=1}^m \lambda_l(K_l - I)} &= \lim_{\Delta\to 0} \left(\prod_{l=1}^m [I + \Delta\lambda_l(K_l - I)] \right)^{\frac{\beta}{\Delta}} \\ &= \lim_{\Delta\to 0} \left(\prod_{l=1}^m [(1 - \Delta\lambda_l)I + \Delta\lambda_l K_i)] \right)^{\frac{\beta}{\Delta}}, \end{aligned}$$

which proves (12).

Next, w.l.o.g, we can assume that $\frac{\beta}{\Delta} \in \mathbb{N}$, as this was a requirement in the proof. Now, for each $l = 1, 2, ..., m$, we consider a sequence $\underline{\xi}_i$ of independent Bernoulli random variables ξ_i

$$\underline{\xi}_l = \{\xi_l(1), \xi_l(2), ..., \xi_l(\beta/\Delta)\},$$

where each $\xi_l(j)$ has a probability of success $p = \Delta\lambda_l$. We assume these $\underline{\xi}_l$ to be independent and let $\mathbb{P}^{\underline{\lambda}}_{\beta,\Delta}$ be the corresponding probability measure on

$$\underbrace{\{0,1\}^{\frac{\beta}{\Delta}} \times ... \times \{0,1\}^{\frac{\beta}{\Delta}}}_{m\text{-times}},$$

where $\underline{\lambda}$ is the vector containing all λ_l. We can now the right side of (12) in terms of this probability measure. The reason why we introduced the Bernoulli random variables in the first place was because you can view the inside of the product in the linearization as a 'process of selecting either I with probability $(1 - \Delta\lambda_l)$ or K_l with a probability $\Delta\lambda_l$'. This is not really what happens, all we do is expand the product, by which you get all permutations of the two terms, and their prefactors are exactly the probabilities of doing the process of selection of either I or K_l, which is of course no coincidence. If we put this underlying idea into mathematics, we get the following:

$$\left(\prod_{l=1}^m [(1 - \Delta\lambda_l)I + \Delta\lambda_l K_l)] \right)^{\frac{\beta}{\Delta}} = \sum_{\underline{a}_1,...,\underline{a}_m} \mathbb{P}^{\underline{\lambda}}_{\beta,\Delta} \left(\bigcap_{l=1}^m \{\underline{\xi}_l = \underline{a}_l\} \right) \mathcal{K}_{\underline{a}}. \tag{14}$$

Here, $\underline{a}_1, ..., \underline{a}_m$ represent realisations of the $\underline{\xi}_1, ..., \underline{\xi}_m$ and $\underline{a}$ the vector containing all those realization, just like $\underline{\xi}$. So on the RHS, the sum cycles through all possible $\underline{a}_1, ..., \underline{a}_m$ separately, thereby cycling through all possible realizations of the earlier described process. $\mathcal{K}_{\underline{a}}$ is the product of the 'selected' K_l's. In mathematical terms:

$$\mathcal{K}_{\underline{a}} \overset{\Delta}{=} \mathcal{K}_{\underline{a}_1,...,\underline{a}_m} = \prod_{j=1}^{\beta/\Delta} \left\{ \prod_{l=1}^m [(1 - a_l(j))I + a_l(j)K_l] \right\} \tag{15}$$

We now want to interpret this series of Bernoulli random variables as a Bernoulli point process and eventually we will take the limit of $\Delta \to 0$ and we will show that we have convergence towards a Poisson point process. The idea behind this is that we will associate arrivals of these point processes with the operators K_l. We start by defining the Bernoulli point processes on the interval $(0, \beta]$. This we do by defining $\xi_l^\Delta(A)$ as follows:

$$\xi_l^\Delta(A) = \sum_{j=1}^{\beta/\Delta} \xi_l(j)\delta_{j\Delta}(A),\ \delta_{j\Delta} = \begin{cases} 1 & \text{if } j\Delta \in A \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

where A is any measurable set in $\mathbb{X}$. It should be clear that ξ_l^Δ is only non-zero wherever $\xi_l(j) = 1$, so we often use the notation $\xi_l^\Delta = \{j\Delta : \xi_l(j) = 1\}$. With these point processes, we give a spatial context to the successes of the underlying Bernoulli random variables.

We now want to define a function $\mathfrak{l}^\Delta$ that takes arrival times $t \in \xi^\Delta \triangleq \cup \xi_i^\Delta$ and gives the arrival 'type', which is the index l of the node at which the arrival was, so that when $t \in \xi_l^\Delta$, we have $\mathfrak{l}^\Delta(t) = l$. This function will be crucial later. However, this function is ill-defined, because two point processes could have an arrival at the same time. To fix this, we need to assume that all ξ_l^Δ are disjoint, which is reasonable considering that this is an event that happens almost surely anyways as $\Delta \to 0$. This fact will be needed later, so we prove it in the lemma below.

Lemma 2.5. *Let $\xi_1^\Delta, ..., \xi_m^\Delta$ be a sequence of Bernoulli processes on $(0, \beta]$ with spacing Δ and probability of arrival $p = \lambda_l \Delta$. Then*

$$\lim_{\Delta \to 0} \mathbb{P}\left(\forall p, q \in \{1, ..., m\} : \xi_p^\Delta \cap \xi_q^\Delta = \emptyset\right) = 1$$

We prove Lemma 2.5 in the appendix B. We will call $\hat{\xi}^\Delta \triangleq \cup \hat{\xi}_l{}^\Delta$ the restricted measures, where all arrival times t must be disjoint, or:

$$t \in \hat{\xi}^\Delta \triangleq \cup \hat{\xi}_l^\Delta = \left\{ j\Delta : \sum_{l=1}^{m} \hat{\xi}_l^\Delta((j\Delta - \Delta/2, j\Delta + \Delta/2)) = 1 \right\}$$

The arrival type function $\mathfrak{l}^\Delta(t)$ is now properly defined. We can now rewrite $\mathcal{K}_{\underline{a}}$ from (15) in terms of this function:

$$\mathcal{K}_{\underline{a}} = \prod_{j=1}^{\beta/\Delta} \left\{ \delta_{\{j\Delta \notin \hat{\xi}^\Delta\}} I + \delta_{\{j\Delta \in \hat{\xi}^\Delta\}} K_{\mathfrak{l}^\Delta(j\Delta)} \right\} \triangleq \prod_{j=1}^{\beta/\Delta} \tilde{K}_j^\Delta \tag{17}$$

This now eliminates the second product of (15) and we are now only multiplying over timesteps Δ. Because of this, we will be able to rewrite the expression as a path integral, by breaking up the product of these operators. To do this, we apply the operators to a inner product with Ψ and Ψ', which are elements of the basis $\{\Psi_i\}$. We can do this by applying a formula, called the Product Expansion Formula.

Theorem 2.6 (Product Expansion Formula)**.** *Let $A_1, A_2, ..., A_n$ be self-adjoint matrices and Ψ, Ψ' two vectors in $\mathbb{X}$. Then,*

$$\langle \Psi | A_1 ... A_n | \Psi' \rangle = \sum_{\Psi_{i_1}, ... \Psi_{i_{n-1}}} \langle \Psi | A_1 | \Psi_{i_1} \rangle \langle \Psi_{i_1} | A_2 | \Psi_{i_2} \rangle ... \langle \Psi_{i_{n-1}} | A_n | \Psi' \rangle, \tag{18}$$

where the Ψ_{i_l} run through all elements of the basis $\{\Psi_i\}$ for all $l = 1, ..., n-1$.

The proof of Theorem 2.6 is provided in Appendix C. Then by the product expansion formula, we have

$$\langle \Psi | \mathcal{K}_{\underline{a}} | \Psi' \rangle = \sum_{\Psi_{i_1}, ... \Psi_{i_{\beta/\Delta - 1}}} \langle \Psi | \tilde{K}_1^\Delta | \Psi_{i_1} \rangle \prod_{j=2}^{\beta/\Delta - 1} \langle \Psi_{i_{j-1}} | \tilde{K}_j^\Delta | \Psi_{i_j} \rangle \langle \Psi_{i_{\beta/\Delta - 1}} | \tilde{K}_{\beta/\Delta}^\Delta | \Psi' \rangle \tag{19}$$

where we also know that

$$\left\langle \Psi_l \left| \tilde{K}_j^\Delta \right| \Psi_k \right\rangle = \begin{cases} \delta_{\{\Psi_l = \Psi_k\}} & \text{if } j\Delta \notin \hat{\xi}^\Delta i^\Delta \\ \langle \Psi_l | \, K_{\mathfrak{l}^\Delta}(j\Delta) | \Psi_k \rangle & \text{if } j\Delta \in \hat{\xi}^\Delta \end{cases} \tag{20}$$

This holds because if $j\Delta \notin \hat{\xi}^\Delta$, then $\tilde{K}_j^\Delta$ is simply the identity matrix by (17) and then we are left with an inner product of two elements of an orthonormal basis, which is 1 if they are the same and 0 if they are different. If $j\Delta \in \hat{\xi}^\Delta$, then $\tilde{K}_j^\Delta$ is simply $K_{\mathfrak{l}^\Delta(j\Delta)}$, so it is clear that (20) holds.

We can now finally put this into continuous time context as follows: to a given sequence $\Psi, \Psi_{i_1}, \ldots, \Psi_{i_{\beta/\Delta - 1}}, \Psi'$ we associate a piecewise constant function $\Psi^\Delta : [0, \beta] \to \{\Psi_j\}$, such that $\Psi^\Delta = \Psi$ on $[0, \Delta)$, $\Psi^\Delta(\beta) = \Psi'$ and

$$\Psi^\Delta = \Psi_{i_j} \text{ on } [j\Delta, (j+1)\Delta) \text{ for } j = 1, \ldots, \beta/\Delta - 1$$

We say that Ψ^Δ is compatible with $\hat{\xi}^\Delta$ or $\Psi^\Delta \sim \hat{\xi}^\Delta$ if all the discontinuities occur at the arrival times of $\hat{\xi}^\Delta$. Then only the compatible functions contribute to the sum of (20). Moreover, in terms of Ψ^Δ, the product expansion reads as follows:

$$\left\langle \Psi \left| \mathcal{K}_{\underline{a}_1, ..., \underline{a}_m} \right| \Psi' \right\rangle = \sum_{\Psi^\Delta \sim \hat{\xi}^\Delta} \prod_{t \in \hat{\xi}^\Delta} \left\langle \Psi^\Delta(t-) \left| K_{\mathfrak{l}^\Delta(t)} \right| \Psi^\Delta(t) \right\rangle \tag{21}$$

The last important step is letting $\Delta \to 0$ and proving that the disjoint Bernoulli processes converge to independent Poisson point processes, so that

$$\left(\hat{\xi}_1^\Delta, \ldots, \hat{\xi}_m^\Delta, \mathfrak{l}^\Delta \right), \to (\xi_1, \ldots, \xi_m, \mathfrak{l})$$

where the ξ_l's are all independent Poisson point processes with intensities λ_l. We define $\mathfrak{l}(t)$ to still map arrival times t of the Poisson point processes ξ_l to l, which is now a well-behaved map. We present a proof of these claims in Theorem 2.8. However, before we go over that proof, we present an important theorem which gives us sufficient conditions for convergence in distribution of point processes. This is found in Leadbetter, Lindgren and Rootzen [7, Theorem A.1] and it is stated as follows:

Theorem 2.7 (Convergence of point processes)**.** *Let $N_n, n = 1, 2, \ldots,$ and N be point processes on the semi-closed interval S in the real line, N being simple. Suppose that*

(i) $\mathbb{E}\left(N_n((c,d])\right) \to \mathbb{E}(N((c,d]))$ *for all* $-\infty < c < d < \infty$ *such that* $[c,d] \subset S$*, and*

(ii) $\mathbb{P}\left\{N_n(B) = 0\right\} \to \mathbb{P}\{N(B) = 0\}$ *for all B of the form* $\bigcup_1^k (c_i, d_i]$*, with* $[c_i, d_i] \subset S$*, for* $i = 1, \ldots, k; k = 1, 2, \ldots$.

Then $N_n \xrightarrow{d} N$.

We will omit the proof of this theorem in this thesis. Instead, we will use these known criteria to prove that $\hat{\xi}^\Delta \xrightarrow{d} \xi$. We do this in the following theorem.

Theorem 2.8. *Let $\xi_1^\Delta, \ldots, \xi_m^\Delta$ be Bernoulli point processes with spacing Δ on the interval $(0, \beta]$ and with success probability $\Delta\lambda_l$, $l \in \{1, ..., m\}$. Next, define $\hat{\xi}_1^\Delta, \ldots, \hat{\xi}_m^\Delta$ to be the same as $\xi_1^\Delta, \ldots, \xi_m^\Delta$, but conditioned to be disjoint everywhere. Then the following statements are true:*

(i) $(\hat{\xi}_1^\Delta, \ldots, \hat{\xi}_m^\Delta) \xrightarrow{d} (\xi_1, \ldots, \xi_m)$ *as* $\Delta \to 0$*, where $\xi_1, \ldots, \xi_m$ are independent Poisson point processes on $(0, \beta]$ with rate parameters $\lambda_1, \ldots, \lambda_m$.*

(ii) The arrival times of the point processes $\xi_1, \ldots, \xi_m$ are disjoint $\mathbb{P}$-a.s.

(iii) Let $\xi = \bigcup_{l=1}^m \xi_l$*. Then* $\mathbb{P}(\mathfrak{l}(t) = l | t \in \xi) = \frac{\lambda_l}{\lambda_1 + \ldots + \lambda_m}$

We will only prove *(i)* here, the rest of the proof can be found in Appendix D.

Proof. (i) We will prove this using Theorem 2.7. First of all, we check if we meet the right conditions to use the theorem. Recall that we let Δ converge to zero such that $1/\Delta$ is an integer. Next, we know that $\hat{\xi}_l$ are in fact simple point processes, as they have weights 1 everywhere. Furthermore, we are working on the semi-closed interval $(0, \beta]$. Before proving the conditions of Theorem 2.7, we first prove a very important intermediate step. Let any $c, d \in (0, \beta]$ with $c < d$. We will start by showing that

$$\lim_{\Delta \to 0} \mathbb{E}\left(\xi_l^\Delta((c,d])\right) = \lim_{\Delta \to 0} \mathbb{E}\left(\hat{\xi}_l^\Delta((c,d])\right). \tag{22}$$

We now let D_Δ be the event that all ξ_l^Δ are disjoint, then we can write the statement above as

$$\lim_{\Delta \to 0} \mathbb{E}\left(\xi_l^\Delta((c,d])\right) = \lim_{\Delta \to 0} \mathbb{E}\left(\xi_l^\Delta((c,d]) | D_\Delta\right).$$

Next, we know from the law of total expectation that

$$\mathbb{E}\left(\xi_l^\Delta((c,d])\right) = \mathbb{E}\left(\xi_l^\Delta((c,d]) | D_\Delta\right) \mathbb{P}\left(D_\Delta\right) + \mathbb{E}\left(\xi_l^\Delta((c,d]) | D_\Delta^c\right) \mathbb{P}\left(D_\Delta^c\right)$$

where D_Δ^c represents the complement of D_Δ, meaning that $\cap \xi_l^\Delta \neq \emptyset$. We now aim at controlling every factor in both of the terms. The easiest are the probabilities, as we have already proven in Lemma 2.5 that $\lim_{\Delta \to 0} \mathbb{P}\left(D_\Delta\right) = 1$ so if we now prove that the second term converges to 0, we prove (22). To get started on that, notice that we can write

$$\mathbb{E}\left(\xi_l^\Delta((c,d]) | D_\Delta^c\right) \mathbb{P}\left(D_\Delta^c\right) = \mathbb{E}\left(\xi_l^\Delta((c,d]) \mathbf{1}_{D_\Delta^c}\right).$$

Next, we split up the event into 2 cases: either the specific ξ_l^Δ we are considering has an arrival time that is joined with some other set, or the other sets have the joined arrival times. Notation-wise, we will achieve this by calling $D^c_{\Delta p,q}$ the event that 2 processes p and q share an arrival time. We can now write that

$$\mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{D^c_\Delta}\right) \leqslant \sum_{\substack{p=1,q=1\\p\neq q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{D^c_{\Delta p,q}}\right) + \sum_{\substack{q=1\\q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{D^c_{\Delta l,q}}\right). \tag{23}$$

The reason why we have an inequality is because on the RHS we count the cases where there are two or more pairs of joined arrival times multiple times. We will find an upper bound to these sums one by one. We start with the first sum, which is the case where the we do not have a joined arrival on ξ_l^Δ.

$$\sum_{\substack{p=1,q=1\\p\neq q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{D^c_{\Delta p,q}}\right) = \sum_{\substack{p=1,q=1\\p\neq q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])|D^c_{\Delta p,q}\right)\mathbb{P}\left(D^c_{\Delta p,q}\right)$$

Because the event $D^c_{\Delta p,q}$ is independent from whatever happens at node l, the conditioning does not change the expectation, so we get:

$$\sum_{\substack{p=1,q=1\\p\neq q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])|D^c_{\Delta p,q}\right)\mathbb{P}\left(D^c_{\Delta p,q}\right) = \mathbb{E}\left(\xi_l^\Delta((c,d])\right)\sum_{\substack{p=1,q=1\\p\neq q\neq l}}^{m} \mathbb{P}\left(D^c_{\Delta p,q}\right)$$

The expectation is easy to calculate, it is simply the number of points $j\Delta, j \in 1, ..., \beta/\Delta$ that are in the interval $(c,d]$. That number can at maximum be $\left\lceil\frac{d-c}{\Delta}\right\rceil$, which we will call N_Δ, where we remember that it is of order $\mathcal{O}(1/\Delta)$. Next, the probability of $D^c_{\Delta p,q}$ can easily bounded from above by letting $\lambda = \max(\lambda_1, ..., \lambda_m)$. Then the probability of this event, is simply the chance for 2 arrivals to coincide in time, i.e. $(\Delta\lambda)^2$, multiplied by the number of independent times where it can happen, i.e. β/Δ. The sum is simply a sum on the permutations of p and q over $m-1$ types, so we get:

$$\mathbb{E}\left(\xi_l^\Delta((c,d])\right)\sum_{\substack{p=1,q=1\\p\neq q\neq l}}^{m} \mathbb{P}\left(D^c_{\Delta p,q}\right) \leqslant N_\Delta\Delta\lambda_l\binom{m-1}{2}\Delta^2\lambda^2\frac{\beta}{\Delta} = \mathcal{O}(\Delta),$$

which means that this part of the sum converges to zero. The next sum we look at is the other case, being the one where l does have a joined arrival on it. First, we split up the indicator function as follows:

$$\mathbf{1}_{D^c_{\Delta l,q}} \leqslant \sum_{j=1}^{\beta/\Delta} \mathbf{1}_{\{\xi_l(j\Delta\pm\Delta)=1\}}\mathbf{1}_{\{\xi_q(j\Delta\pm\Delta)=1\}},$$

where $j\Delta \pm \Delta = (j\Delta - \Delta/2, j\Delta + \Delta/2]$. Again we have an inequality because we are double counting the cases where we have 2 or more joined arrivals on j and q. This means that the second sum in (23) is now bounded as follows

$$\sum_{\substack{q=1\\q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{D^c_{\Delta l,q}}\right) \leqslant \sum_{j=1}^{\beta/\Delta}\sum_{\substack{q=1\\q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{\{\xi_l(j\Delta\pm\Delta)=1\}}\mathbf{1}_{\{\xi_q(j\Delta\pm\Delta)=1\}}\right).$$

Notice now that the event $\{\xi_q(j\Delta \pm \Delta) = 1\}$ is independent from the rest, so we can again separate its probability out:

$$\sum_{j=1}^{\beta/\Delta} \sum_{\substack{q=1 \\ q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{\{\xi_l(j\Delta\pm\Delta)=1\}}\mathbf{1}_{\{\xi_q(j\Delta\pm\Delta)=1\}}\right) =$$

$$\sum_{j=1}^{\beta/\Delta} \sum_{\substack{q=1 \\ q\neq l}}^{m} \mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{\{\xi_l(j\Delta\pm\Delta)=1\}}\right)\mathbb{P}\left(\{\xi_q(j\Delta \pm \Delta) = 1\}\right)$$

By calculating out the probabilities, this last sum can be simplified to the following expression:

$$(m-1)\Delta\lambda \sum_{j=1}^{\beta/\Delta} \mathbb{E}\left(\xi_l^\Delta((c,d])\mathbf{1}_{\{\xi_l(j\Delta\pm\Delta)=1\}}\right)$$

where $\lambda = \max(\lambda_1, ..., \lambda_m)$.

Lastly, we can pull the sum over j back inside the expectation to get

$$(m-1)\Delta\lambda\mathbb{E}\left(\xi_l^\Delta((c,d]) \sum_{j=1}^{\beta/\Delta} \mathbf{1}_{\{\xi_l(j\Delta\pm\Delta)=1\}}\right)$$

Notice how the sum of these indicators is nothing more than counting the number of arrivals on $(0,\beta]$, which is exactly what $\xi_l^\Delta((0,\beta])$ is.

$$(m-1)\Delta\lambda\mathbb{E}\left(\xi_l^\Delta((c,d]) \sum_{j=1}^{\beta/\Delta} \mathbf{1}_{\{\xi_l(j\Delta\pm\Delta)=1\}}\right) \leqslant (m-1)\Delta\lambda\mathbb{E}\left(\xi_l^\Delta((0,\beta])^2\right)$$

The expression in the expectation is now nothing more than the second moment of a binomial $\text{bin}(n,p)$, which is a well known calculation that is equal to $n(n-1)p^2 + np$. We can apply this formula to get that

$$\begin{aligned}(m-1)\Delta\lambda\mathbb{E}\left(\xi_l^\Delta((0,\beta])^2\right) &= (m-1)\Delta\lambda\left(\frac{\beta}{\Delta}\left(\frac{\beta}{\Delta}-1\right)(\Delta\lambda_l^2) + \frac{\beta}{\Delta}\lambda_l\right) \\ &= (m-1)\Delta\lambda\left(\beta^2\lambda_l^2 - \beta\lambda_l^2\Delta + \beta\lambda_l\right) \\ &= \mathcal{O}(\Delta).\end{aligned}$$

This means that both the sums of (23) are of order Δ, which then implies that as $\Delta \to 0$, the limit of (22) also holds. We can apply condition *(i)* of Theorem 2.7 to the independent Bernoulli processes ξ_l^Δ instead of $\hat{\xi}_l^\Delta$. Proving *(i)* is now quite trivial. We want to show that

$$\lim_{\Delta\to 0} \mathbb{E}\left(\xi_l^\Delta((c,d])\right) = \mathbb{E}\left(\xi_l((c,d])\right).$$

We can calculate the left hand side with a squeeze theorem argument:

$$\lim_{\Delta\to 0}\lfloor(d-c)/\Delta\rfloor)\Delta\lambda_l \leqslant \lim_{\Delta\to 0} \mathbb{E}\left(\xi_l^\Delta((c,d])\right) \leqslant \lim_{\Delta\to 0}\lceil(d-c)/\Delta\rceil)\Delta\lambda_l.$$

Now we have that $\lfloor (d-c)/\Delta \rfloor) = (d-c)/\Delta - \text{frac}((d-c)/\Delta)$, where $\text{frac}(\cdot)$ is a function that gives the fractional part of the number in the argument, and we also have that $\lceil (d-c)/\Delta \rceil) = (d-c)/\Delta + 1 - \text{frac}((d-c)/\Delta)$. We then have that

$$\begin{aligned}\lim_{\Delta\to 0}\left(\frac{d-c}{\Delta} - \text{frac}\left(\frac{d-c}{\Delta}\right)\right)\Delta\lambda_l &\leqslant \lim_{\Delta\to 0}\mathbb{E}\left(\xi_l^{\Delta}((c,d])\right) \\ &\leqslant \lim_{\Delta\to 0}\left(\frac{d-c}{\Delta} + 1 - \text{frac}\left(\frac{d-c}{\Delta}\right)\right)\Delta\lambda_l,\end{aligned}$$

which by the squeeze theorem implies that

$$\lim_{\Delta\to 0}\mathbb{E}\left(\xi_l^{\Delta}((c,d])\right) = (d-c)\lambda_l = \mathbb{E}(\xi_l((c,d])).$$

Now all that remains is to prove condition *(ii)* of Theorem 2.7. Let B be a set such as in that condition and let $\text{Leb}(B)$ be the Lebesgue measure of that set so that $\text{Leb}(B) = \sum_{i=1}^{k}(d_i - c_i)$. Then we can write that

$$\lim_{\Delta\to 0}\mathbb{P}\left(\hat{\xi}_l^{\Delta}(B) = 0\right) = \lim_{\Delta\to 0}\mathbb{P}\left(\xi_l^{\Delta}(B) = 0 | D_{\Delta}\right).$$

Here, conditioning on the event D_Δ means that all the processes, except l, must be disjoint, because we are calculating the probability on an event where we do not have any arrivals at l anyways. Let us call this event $D_{\Delta,l}$. Furthermore, that event is independent from process l anyways, so we can multiply their probabilities. Therefore we can say that

$$\begin{aligned}\lim_{\Delta\to 0}\mathbb{P}\left(\xi_l^{\Delta}(B) = 0 | D_{\Delta}\right) &= \lim_{\Delta\to 0}\frac{\mathbb{P}\left(\xi_l^{\Delta}(B) = 0 \cap D_{\Delta}\right)}{\mathbb{P}(D_{\Delta})} \\ &= \lim_{\Delta\to 0}\frac{\mathbb{P}\left(\xi_l^{\Delta}(B) = 0\right)\mathbb{P}\left(D_{\Delta,l}\right)}{\mathbb{P}(D_{\Delta})}\end{aligned}$$

The probabilities $\mathbb{P}(D_\Delta)$ and $\mathbb{P}\left(D_{\Delta,l}\right)$ can be found using Lemma 2.5. Since D_Δ is simply the event that all processes $\xi\Delta_1, ..., \xi\Delta_m$ are disjoint, its probability is simply 1. Next, since $D_{\Delta,l}$ is simply the same, but with one less process, the lemma still holds, so it has probability 1 as well. For the probability $\mathbb{P}\left(\xi_l^{\Delta}(B) = 0\right)$, it comes down to a counting problem again. We can use the same squeeze theorem argument as before to say that the number of points $j\Delta$ in an interval $(a,b]$, converges to $(b-a)/\Delta$. Remember that B is a union $\bigcup_{i=1}^{k}(c_i, d_i]$, but these intervals could be overlapping. However, without loss of generality, we can say that we can rewrite this as a union of disjoint intervals $\bigcup_{i=1}^{k}(a_i, b_i]$ such that $\text{Leb}\left(\bigcup_{i=1}^{k}(a_i, b_i]\right) = \sum_1^k \text{Leb}\left((a_i, b_i]\right) = \text{Leb}(B)$. Now the counting problem gives us that the total number of points $j\Delta$ in B as $\Delta \to 0$ is $\sum_1^k \frac{b_i - a_i}{\Delta} = \frac{\text{Leb}(b_i - a_i)}{\Delta}$Now, we can finally say that

$$\lim_{\Delta\to 0}\mathbb{P}\left(\xi_l^{\Delta}(B) = 0\right) = \lim_{\Delta\to 0}(1-\Delta\lambda_i)^{\text{Leb}(B)/\Delta} = \mathrm{e}^{-\text{Leb}(B)\lambda_i} = \mathbb{P}(\xi_l(B) = 0)$$

Part *(ii)* and *(iii)* are proven in Appendix D. □

By Theorem 2.8 *(i)*, we now have that point processes $\hat{\xi}_l \xrightarrow{d} \xi_l$ and ξ_l are Poisson point processes. We call $\mathbb{P}_\beta^\lambda$ the distribution of these Poisson processes $\xi_1, ..., \xi_m$. We can now let $\Delta \to 0$ in (14), as well as in (21), where $\hat{\xi}^\Delta \to \xi$ and $\mathfrak{l}^\Delta \to \mathfrak{l}$. We know that $\mathfrak{l}$ now behaves properly by Theorem 2.8 *(ii)* and *(iii)*. If we now take two basis elements Ψ^1 and Ψ^2, the inner product of those vectors applied to (14) becomes

$$\frac{\left\langle \Psi^1 \left| \mathrm{e}^{\beta \sum \lambda_i K_i} \right| \Psi^2 \right\rangle}{\exp\left\{\beta \sum_i \lambda_i\right\}} = \int \mathbb{P}_\beta^\lambda \left(\mathrm{d}\xi\right) \sum_{\Psi \sim \xi} \prod_{t \in \xi} \left\langle \Psi(t-) \left| K_{\mathfrak{l}(t)} \right| \Psi(t) \right\rangle, \tag{24}$$

where the sum and product run over all functions Ψ that are compatible to the given realization of ξ. Here, the Ψ's are essentially just like what Ψ^Δ was in relation to $\hat{\xi}^\Delta$. They are still piecewise constant with jumps only on successes in ξ and their boundaries are $\Psi(0) = \Psi^1, \Psi(\beta) = \Psi^2$. However, the jump times are now on an interval instead of on discrete points. There are only a finite number of these compatible functions, because $\mathbb{X}$ is finite dimensional and the Poisson processes are on a finite time interval. We have now successfully turned the expectation of an 'Ising Hamiltonian-like exponential' into a path integral over the interval $[0, \beta]$. With this path integral formulation on the time axis $(0, \beta]$, we can also express some other quantities. The trace, which is equal to the partition function, can be expressed as follows

$$\frac{\mathrm{Tr}\left(\mathrm{e}^{\beta \sum \lambda_i K_i}\right)}{\exp\left\{\beta \sum_i \lambda_i\right\}} = \int \mathbb{P}_\beta^\lambda(\mathrm{d}\xi) \sum_{\Psi \sim \xi} \langle \Psi(0) \mid \Psi(\beta) \rangle \prod_{t \in \xi} \left\langle \Psi(t-) \left| K_{\mathfrak{l}(t)} \right| \Psi(t) \right\rangle, \tag{25}$$

where we added the factor $\langle \Psi(0) \mid \Psi(\beta) \rangle$ to make sure that we only pick diagonal elements. We can also add any adjoint matrix A in front of the exponent, and by the definition of adjoint operators, we can pull it inside the integral and into the inner product. We then get the following expression for the expectation of A

$$\frac{\mathrm{Tr}\left(A\mathrm{e}^{\beta \sum \lambda_i K_i}\right)}{\exp\left\{\beta \sum_i \lambda_i\right\}} = \int \mathbb{P}_\beta^\lambda(\mathrm{d}\xi) \sum_{\Psi \sim \xi} \langle \Psi(0)|A|\Psi(\beta) \rangle \prod_{t \in \xi} \left\langle \Psi(t-) \left| K_{\mathfrak{l}(t)} \right| \Psi(t) \right\rangle. \tag{26}$$

This is the path integral we wanted to achieve, as well as reformulation of the relevant quantities. We derived it in an even more general formulation than we had earlier with the Ising model related operators $\hat{\sigma}_{ij}^z$ and $\hat{\sigma}_i^z$. In the next section we will apply this to the Ising model and decompose the Hamiltonian in order to derive what we call the FK-representation of the Ising model.

2.2.1 FK-representation

The Ising Hamiltonian like it is written in (9) is the most simplified form to write it, but we can also refactor it in different ways that are useful to us. To get the FK-representation, named after the mathematicians Fortuin and Kasteleyn who originally derived it, we need to refactor it in the following way:

$$-\mathcal{H}_\Lambda = -\left(\sum_{(i,j)} J_{ij} + \sum_i h\right)\mathrm{I} + \sum_{(i,j)} 2J_{ij}\frac{\mathrm{I} + \hat{\sigma}_i^z\hat{\sigma}_j^z}{2} + \sum_i 2h\frac{\mathrm{I} + \hat{\sigma}_i^z}{2}. \tag{27}$$

It is not hard to see that they are in fact the same Hamiltonian. However, this way of factoring now allows us to interpret the model in a totally different way. More specifically, we apply the path integral representation to this Hamiltonian with the Poisson processes ξ_{ij} representing the arrival of operators $K_{ij} \triangleq \frac{I+\hat{\sigma}_i^z\hat{\sigma}_j^z}{2}$ with intensities J_{ij} and ξ_i representing the arrival of operators $\frac{I+\hat{\sigma}_i^z}{2}$ with intensities $2h$. We call the entire Poisson process $\xi = \{\xi_{ij}, \xi_i\}$.

Arrivals of these operators have a certain influence of the model, which we will now investigate. Suppose ν and ν' are two classical configurations, with corresponding elements Ψ_ν and $\Psi_{\nu'}$ from $\mathbb{X}$. Then we get the following when either operator arrives:

$$\langle \Psi_\nu | K_{ij} | \Psi_{\nu'} \rangle = \delta_{\{\nu=\nu'\}}\delta_{\{\nu_i=\nu_j\}} \qquad \text{and} \qquad \langle \Psi_\nu | K_i | \Psi_{\nu'} \rangle = \delta_{\{\nu=\nu'\}}\delta_{\{\nu_i=1\}}. \tag{28}$$

The operators that arrive act as a sort of filter that narrows down the allowed spin configurations, such that only specific configurations are let through. This effect is mathematically enacted by the delta-functions. For the K_{ij} type operator, they insure that ν and ν' are identical and that ν_i and ν_j are the same spin. Why this happens is simply the way K_{ij} is defined. It can be an identity matrix when it node i and j have the same spin, or a null matrix when they are opposite. Suppose they are the same, then all that's left is an inner product between Ψ_ν and $\Psi_{\nu'}$, which gives us the other delta function. The arrival of operators K_i is the same story but by the way it's designed, it forces a specific node to be 1.

We now need to define the overarching Ψ which determines compatibility. It has to be a piecewise constant function with jumps at the arrival times of the Poisson processes. On its constant pieces, it always equals a certain Ψ_ν, so that $\Psi : [0, \beta) \mapsto \{\Psi_\nu\}$. We can also rewrite this by letting the ν depend on time. We write $\Psi_{\nu(\bullet)}$ where $\nu(\bullet) : [0, \beta) \mapsto \Omega_\Lambda$. We can now draw some conclusions from this. Firstly, given that the arrival of the any operator has a delta-function imposes a condition of equality of the classical configuration for the Ψ_ν's 'between' which it arrives (in the language of equation 21). This means that there is no operator that can arrive that can change spins over time, not even the type K_i, which does force a specific node to be of spin $\nu_i = 1$, but before that it also needed to be 1 when it arrived. The conclusion to draw from this is that $\Psi_{\nu(\bullet)}$ is in fact constant in time, so we might as well call it Ψ_ν from now on. This means that it doesn't actually matter when an operator arrives, and the effect of the arrival is relevant before and after it arrives. An arrival of any K_i forces it's corresponding spin to be $\nu_i = 1$ as stated before and K_{ij} forces $\nu_i = \nu_j$. We can represent these two types of arrivals with bridges and $\times$-marks, like in Figure 1.

Each vertical axis on the figure represents the time evolution of a particular node. For the i-th node, we call $\mathbb{S}_\beta^i = [0, \beta)$ its time interval. Furthermore, we will also refer to these as 'sticks'. We make the plot as follows: if there is an arrival of ξ_i at time t, then we would plot an $*$-mark at the coordinate (i, t) and if there is an arrival of any ξ_{ij} at time t, we would plot a bridge between points (i, t) and (j, t). Furthermore, we also consider nodes with red *times*-marks to be connected to a so called 'ghost node' $\mathfrak{g}$. We now want to split up any realization of ξ into connected components. We call sticks $\mathbb{S}_\beta^i$ and $\mathbb{S}_\beta^j$ connected if $\xi_{ij} \neq \emptyset$. This connection property is transitive, so suppose $\mathbb{S}_\beta^1$ and $\mathbb{S}_\beta^2$ are connected and $\mathbb{S}_\beta^2$ and $\mathbb{S}_\beta^3$ are connected, then $\mathbb{S}_\beta^1$ and $\mathbb{S}_\beta^3$ are connected as well. If we have a connection of this sort, we like to consider them as all being in one big connected component. This way, we can split up any instance of ξ into maximally connected components. We call

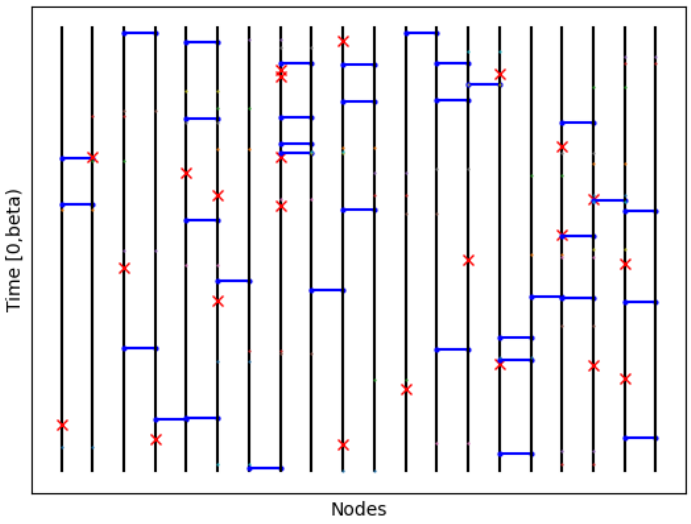


Figure 1: This is a graphical representation of the Poisson process ξ. The vertical axis is the time axis running from $[0, \beta]$, the horizontal represents the different nodes on $\{1, 2, ..., 20\}$. In this case there are a lot of arrivals of both types. Firstly, a blue line or bridge between node i and j represents an arrival in ξ_{ij}. Secondly, a red $\times$-mark on node i represents an arrival in ξ_i.

each connected component $\mathcal{C}_l$, and we call $\mathcal{A}_l$ the corresponding subset of Λ that has the indices of the sticks that are connected, so that

$$\mathcal{C}_l = \bigcup_{i \in \mathcal{A}_l} \mathbb{S}^i_\beta.$$

We call a component wired if any node in the component is connected to the ghost node or any $\xi_i \neq 0$ for some $i \in \mathcal{A}_l$. It is convenient to think of all of these components as being linked.

Let us now consider any ξ with the corresponding arrived operators K_i and K_{ij}, in accordance with measure $\mathbb{P}^{\mathbf{J},h}_{\beta,\Lambda}$. Let $\#_w(\xi)$ be the number of maximally connected components *not* wired to the ghost site, then the number of classical trajectories that are compatible to (28) is precisely $2^{\#_w(\xi)}$, as each of these non-wired components can be either of spin 1 or -1. Note that nodes that do not have any arrivals, have their own component. For each of these trajectories,

$$\prod_{t \in \xi} \left\langle \Psi_\nu(t-) \left| K_{\mathfrak{l}(t)} \right| \Psi_\nu(t) \right\rangle = \prod_{t \in \xi} \left\langle \Psi_\nu \left| K_{\mathfrak{l}(t)} \right| \Psi_\nu \right\rangle = 1.$$

With this information, we can calculate the trace of $e^{-\beta \mathcal{H}_\Lambda}$ as follows, where we use the results of (25) and (27) to get that

$$
\begin{aligned}
\operatorname{Tr}\left(\mathrm{e}^{-\beta\mathcal{H}_A}\right) &= e^{\sum_{(i,j)} J_{ij}+\sum_i h} \int \mathbb{P}^{\mathbf{J},h}_{\beta,\Lambda}(\mathrm{d}\xi) \sum_{\Psi\sim\xi} \langle \Psi(0) \mid \Psi(\beta)\rangle \prod_{t\in\xi} \left\langle \Psi(t-) \left| K_{\mathfrak{l}(t)} \right| \Psi(t) \right\rangle \\
&= e^{\sum_{(i,j)} J_{ij}+\sum_i h} \int \mathbb{P}^{\mathbf{J},h}_{\beta,\Lambda}(\mathrm{d}\xi) \sum_{\Psi\sim\xi} 1 \\
&= e^{\sum_{(i,j)} J_{ij}+\sum_i h} \int \mathbb{P}^{\mathbf{J},h}_{\beta,\Lambda}(\mathrm{d}\xi) 2^{\#_w(\xi)} \\
&= \mathrm{e}^{\beta\left(\sum_{(i,j)} J_{ij}+\sum_i h\right)} \mathbb{E}\left[2^{\#\mathrm{w}(\xi)}\right]. \qquad (29)
\end{aligned}
$$

We now define a new measure $\widetilde{\mathbb{P}}^{\mathbf{J},h}_{\beta,\Lambda}$ on the same ξ as follows

$$
\widetilde{\mathbb{P}}^{\mathbf{J},h}_{\beta,\Lambda}(\mathrm{d}\xi) \triangleq \frac{2^{\#\mathrm{w}(\xi)} \mathbb{P}^{\mathbf{J},h}_{\beta,\Lambda}(\mathrm{d}\xi)}{\mathbb{E}\left[2^{\#\mathrm{w}(\xi)}\right]}, \qquad (30)
$$

where the expected value is taking with respect to the old measure $\mathbb{P}^{\mathbf{J},h}_{\beta,\Lambda}$. The reason why we do this, is because

$$
\mu^{\beta,h}_{\Lambda}(\nu_i) = \widetilde{\mathbb{P}}^{\mathbf{J},h}_{\beta,\Lambda}(i \longleftrightarrow \mathfrak{g}) \quad \text{and} \quad \mu^{\beta,h}_{\Lambda}(\nu_i\nu_j) = \widetilde{\mathbb{P}}^{\mathbf{J},h}_{\beta,\Lambda}(i \longleftrightarrow j), \qquad (31)
$$

where $(i \longleftrightarrow \mathfrak{g})$ is the event that node i is connected to the ghost node and $(i \longleftrightarrow j)$ is the event that node i is connected to node j. Why this holds will be shown below.

Proof. We start with the first statement of equation 31.

$$
\begin{aligned}
\mu^{\beta,h}_{\Lambda}(\nu_i) &= \frac{\operatorname{Tr}\left(\hat{\sigma}^z_i \mathrm{e}^{-\beta\mathcal{H}_A}\right)}{\operatorname{Tr}\left(\mathrm{e}^{-\beta\mathcal{H}_A}\right)} && \text{(by equation 11)} \\
&= \frac{\displaystyle\int \mathbb{P}^{\mathbf{J},h}_{\beta,\Lambda}(\mathrm{d}\xi) \sum_{\Psi\sim\xi} \langle \Psi(0) \mid \hat{\sigma}^z_i \mid \Psi(\beta)\rangle \prod_{t\in\xi} \left\langle \Psi(t-) \left| K_{\mathfrak{l}(t)} \right| \Psi(t) \right\rangle}{\mathbb{E}\left[2^{\#\mathrm{w}(\xi)}\right]} \\
&= \frac{\mathbb{P}^{\mathbf{J},h}_{\beta,\Lambda}(i \longleftrightarrow \mathfrak{g}) 2^{\#\mathrm{w}(\xi)}}{\mathbb{E}\left[2^{\#\mathrm{w}(\xi)}\right]} \\
&= \widetilde{\mathbb{P}}^{\mathbf{J},h}_{\beta,\Lambda}(i \longleftrightarrow \mathfrak{g})
\end{aligned}
$$

The penultimate step holds because of the fact that if node i is connected to the ghost nodes, the sum will give $2^{\#\mathrm{w}(\xi)}$ as in (29), and if it is not connected, the $\hat{\sigma}^z_i$ will contribute once 1 and once -1 for all ξ, since there will always be two configurations that are compatible in that case.

The second statement of equation 31 will now be proven.

$$\begin{aligned}
\mu_{\Lambda}^{\beta,h}(\nu_i\nu_j) &= \frac{\mathrm{Tr}\left(\hat{\sigma}_i^z\hat{\sigma}_j^z \mathrm{e}^{-\beta\mathcal{H}_A}\right)}{\mathrm{Tr}\left(\mathrm{e}^{-\beta\mathcal{H}_A}\right)} && \text{(by equation 11)} \\
&= \frac{\int \mathbb{P}_{\beta,\Lambda}^{\mathbf{J},h}(\mathrm{d}\xi) \sum_{\Psi\sim\xi} \langle\Psi(0) \mid \hat{\sigma}_i^z\hat{\sigma}_j^z \mid \Psi(\beta)\rangle \prod_{t\in\xi} \left\langle \Psi(t-)\left|K_{\mathfrak{l}(t)}\right|\Psi(t)\right\rangle}{\mathbb{E}\left[2^{\#\mathrm{w}(\xi)}\right]} \\
&= \frac{\mathbb{P}_{\beta,\Lambda}^{\mathbf{J},h}(i \longleftrightarrow j) 2^{\#\mathrm{w}(\xi)}}{\mathbb{E}\left[2^{\#\mathrm{w}(\xi)}\right]} \\
&= \widetilde{\mathbb{P}}_{\beta,\Lambda}^{\mathbf{J},h}(i \longleftrightarrow j)
\end{aligned}$$

The penultimate step holds because of a similar argument to the first part of the proof. □

2.2.2 Random cluster measures

We now have a measure $\widetilde{\mathbb{P}}_{\beta,\Lambda}^{\mathbf{J},h}$ that produces the same results as the Ising model, but in a very different way. The $\widetilde{\mathbb{P}}_{\beta,\Lambda}^{\mathbf{J},h}$ is called an FK-measure or a random cluster measure. They are a class of measures that are more general in their formulation. We will briefly display the definition of a random cluster model [3, pp. 4–6].

We start on a finite graph $G = (\Lambda, \mathcal{E})$. An edge e between node i and j is written as $\mathrm{e} = \langle i, j\rangle$. We define a state space Ω for our measures as $\Omega = \{0,1\}^{|\mathcal{E}|}$ with members $\omega \in \Omega$. We say an edge e is open in ω, if $\omega(e) = 1$ and closed if $\omega(e) = 0$. The σ-algebra on which the measures are defined, is the power set of omega $2^{\Omega} = \mathcal{F}$. Similar to the previous section, we also make connected components with these open or closed edges. We visualize them as gates in this case, so that areas of the graph with open gates form clusters. We call the number of maximally connected components $k(\omega)$.

The random cluster measures $\Phi_{p,q}$ are a set of measures that are distinguished by parameters $p \in [0,1]$ and $q \in (0,\infty)$. They are given by:

$$\phi_{p,q}(\omega) = \frac{1}{Z_{\mathrm{RC}}} \left\{ \prod_{e\in E} p^{\omega(e)}(1-p)^{1-\omega(e)} \right\} q^{k(\omega)}, \tag{32}$$

where Z_{RC} is the partition function, defined as follows:

$$Z_{\mathrm{RC}} = Z_{\mathrm{RC}}(p,q) = \sum_{\omega\in\Omega} \left\{ \prod_{e\in E} p^{\omega(e)}(1-p)^{1-\omega(e)} \right\} q^{k(\omega)}. \tag{33}$$

It should be clear that setting $q = 2$ and $p = \mathbb{P}_{\beta,\Lambda}^{\mathbf{J},h}(i \longleftrightarrow j)$ gives us exactly the same distribution as $\widetilde{\mathbb{P}}_{\beta,\Lambda}^{\mathbf{J},h}$, although a bit more narrow, as the λ_i's of the previous chapter would have to be constant, as well as $h = 0$ so that there are no ghost nodes. It is in that sense less general, but one could easily make p vary per node, so that degree of freedom can be added to the random cluster measure. The degree of freedom in q is the main feature of this model however. Setting it to 1 gives the famous percolation model, and setting it to values of 3 or higher gives the Potts-model [3, pp. 6–7]. What were after in the first place though, was $q = 2$, which gives an Ising model, with the aforementioned simplifications, that is.

3 Simulations of the Ising model

In this section we introduce the methods to analyse the Ising model using simulations. First of all we will take a quick look at what to expect from Ising model simulations and how it is usually simulated. Next, we will analyze whether or not Ioffe's stochastic geometrical interpretation [6] is useful for simulations. It will quickly be obvious that Ioffe's interpretation is not readily applicable in simulations, but with small tweaks, we can modify it to get a working algorithm.

3.1 Introduction to Ising model simulations

The Ising model is a commonly studied model in statistical mechanics. As in all statistical mechanical models, you want to study at the thermodynamic behaviour of microscopic models. This is hard, as the phase space is large. All states in the phase space have a certain probability of occurrence, but only a limited set of them are relevant. It is usually impossible to find these by directly evaluating the partition function as this essentially entails a sum over the phase space. However, there exist various analytical and numerical approaches.

An exact solution is always the best option. The problem is that it usually does not exist or is not yet found. For example, the Ising model has only been solved in one and two dimensions for specific lattice configurations with no external magnetic field.

When an exact solution is not available, physicists can resort to simulations, which are usually very effective as well. Simulations of statistical mechanics problems always include some form of Monte-Carlo method, used to sample the complex underlying probability distribution. With these samples, one can calculate the thermodynamic quantities of the model. It should also be noted that one can often find very useful solutions to approximations of statistical models. For example, mean field theory can produce various useful results regarding the Ising model [12].

The most interesting behaviour of the Ising model is its phase transition in the total magnetization, without external magnetic field. It switches for not being magnetized, to being magnetized at a very specific point and it does so in a non-analytic, but continuous fashion. It is not obvious that this phase transition should even exist. Ernst Ising himself was convinced that the model exhibits no phase transitions, because the 1D model does not show them. The first to prove that there are in fact phase transitions in higher dimensional Ising models was Rudolph Peierls [10].

We studied the behavious of the model near the phase-transition, results of which will be presented later on. However, there are many more aspects to study in the Ising model. For example, topology and dimension of the lattice and the range of the coupling can be varied. We shall focus here on the square-lattice in two dimensions with neirest neighbour coupling $J_{ij} = J = 1$. We study the model near the critical phase transition, i.e. $h = 0$ and β is close to $\beta_{crt} \approx 0.441$.

3.1.1 Metropolis-Hastings algorithm

The Metropolis-Hastings algorithm is the standard tool for Ising model simulations [8, pp. 46–49], and for good reasons: it is easy to implement and it is fast for many applications. Since this algorithm is not the main focus of this thesis, we will not go in depth into why this algorithm works, but we will explain it briefly, along with its main weakness.

The algorithm is quite simple: starting from any configuration, apply the following steps:

1. Let μ be the current state of the model.
2. Pick any site, chosen uniformly, and check whether or not flipping its spin will increase or decrease the total energy of the system. The system with the selected spin flipped, is called ν
3. If it decreases the energy, flip the spin. If it increases the energy, flip the spin with probability $\mathrm{e}^{-\beta(H_\nu - H_\mu)}$
4. Repeat from step 1.

After some time, this procedure will lead to a quasi-stable magnetization, only influenced by thermal excitations, see $\beta = 1/k_B T$ in step 3. As stated before, this algorithm works well most of the time. However its major flaw is that it slows down quite harshly when run close to critical temperatures [8, pp. 53–57]. That is because the correlation length between nodes diverges at the critical temperature. The high correlation length near criticality will cause thermal fluctuations at large scales. Because the algorithm is local, it has difficulties in converging to such a state with high correlation length efficiently. The main thing to look at when studying critical slowing-down of an algorithm, is the dynamic critical exponent z, which will be explained in more detail in section 3.2.2. According to literature, the value for this exponent z for a 2D lattice, is $z = 2.125$ [15]. All algorithms suffer from this long correlation length to some degree, but some overcome this difficulty better than others.

3.2 Simulations of cluster models

Theory section 2.2.1 considers the FK representation of the Ising model. It gives us insight into how we can map the Ising model onto a cluster-based model. Moreover, it gives us a very good idea as to how we can sample the Ising model using a clustering approach. The FK representation tells us that the probability to find a certain Ising configuration can interpreted as an expectation of the sum over all compatible bond configurations, which are defined by Poisson processes ξ_{ij}, ξ_i on $(0, \beta]$ with intensities $2J_{ij}$ and $2h$. Arrival of ξ_{ij} means that nodes i and j are bridged, which can only happen if their spins are equal. Note that the specific arrival time is not of importance, only whether there is an arrival or not. Therefore the probability of two neighbours being bridged, is $1 - \mathrm{e}^{-2\beta_{ij}}$. An arrival of ξ_i forces node i to be of a certain spin. However, this in not of interest to us since we always consider $h = 0$, which excludes arrivals of this operator. The crux is

then that the bridging process forms clusters and that when each clusters is weighted by a factor of 2, we get that this clustering shows the same statistics as the Ising model. With this information, we have a good idea of what an algorithm using this clustering process should look like. However, to put all of it together in just the right way, it still takes a genius. Luckily for us, Swendsen and Wang did the brilliant part already. Their algorithm is presented in the next section.

3.2.1 The Swendsen-Wang Algorithm

The right way to incorporate the FK-representation into simulations is by using the Swendsen-Wang algorithm [15]. Note that this version of the algorithm only works for $h = 0$, so we assume this to be the case when using this algorithm. The setup of the algorithm is as follows: let $\sigma_1, \sigma_2, ..., \sigma_n \in \{-1, 1\}$ represent the spin of nodes 1 to n. Note that σ_j denotes the spin at site number j, which is on a square lattice in our case. Furthermore, let $b_{ij} \in \{0, 1\}$ represent a bond between node i and j where i and j are nearest neighbours, where $b_{ij} = 0$ represents an open bond and $b_{ij} = 1$ represents a closed bonds. Each pair of nearest neighbours i, j has a coupling strength J_{ij}. The Swendsen-Wang algorithm is then applied as follows:

1. Assign random spin values to each node
2. Go through all nearest neighbours i, j and apply the following logic:
   - If the spins are different, let $b_{ij} = 0$
   - If the spins are equal, let $b_{ij} = 0$ with probability $e^{-2\beta J_{ij}}$ and $b_{ij} = 1$ with probability $1 - e^{-2\beta J_{ij}}$
3. Identify the clusters made by closed bonds ($b_{ij} = 1$) in the previous step and assign a random spin to the entire cluster with equal probability of being either spin.
4. All bonds are erased.
5. Repeat form step 2.

It should be very clear that this algorithm is inspired by the FK representation we derived. We will now prove that with this algorithm, the system saturates at the desired distribution. For this to hold, 2 conditions must be met: ergodicity and detailed balance. Ergodicity is trivial. One can see that from any spin state, a bond configuration with all bonds open can be reached, from which point all other Ising configuration can be reached with finite probability. Showing detailed balance is also not that hard. To show detailed balance, we need to show that

$$\mathbb{P}(\sigma)\mathbb{P}(\sigma \to \sigma') = \mathbb{P}(\sigma')\mathbb{P}(\sigma' \to \sigma)$$

where σ and σ' are two spin configurations. Note that the algorithm also passes through some bond configuration when going from σ to σ'. Therefore we fix a particular valid

bond configuration, which we call b, that is a possible intermediate step between σ and σ'. Then we can now rewrite the detailed balance condition as follows:

$$\frac{\mathbb{P}(\sigma'|b)\mathbb{P}(b|\sigma)}{\mathbb{P}(\sigma|b)\mathbb{P}(b|\sigma')} = \frac{\mathbb{P}(\sigma')}{\mathbb{P}(\sigma)}.$$

The probability to be in any configuration, given what the originating bond configuration was, is the same for any one of them, because the cluster spins are all chosen with equal probabilities. This means that $\mathbb{P}(\sigma|b) = \mathbb{P}(\sigma'|b) = p$. The other side of the equality is easily calculated by remembering that the probability of a certain Ising configuration is simply given by the Boltzmann distribution so that we have

$$\frac{\mathbb{P}(\sigma')}{\mathbb{P}(\sigma)} = \frac{\frac{\exp(-\beta\mathbf{H}(\sigma'))}{\mathcal{Z}}}{\frac{\exp(-\beta\mathbf{H}(\sigma))}{\mathcal{Z}}} = \mathrm{e}^{-\beta\Delta E},$$

where $\mathbf{H}$ is the Ising Hamiltonian and ΔE is the energy difference between the spin configurations σ' and σ.

Finally we need to consider $\mathbb{P}(b|\sigma)$ and $\mathbb{P}(b|\sigma')$. Each given bond configuration will always have a certain number of closed bonds and the rest is open. The key observation is that only the open bonds between nodes of equal spin have relevance, seeing as the other ones are closed with probability 1. The bonds between equal spin nodes are controlled by Bernoulli random variables with $p = 1 - \mathrm{e}^{-2\beta J}$. Hence, the total probability of a particular bond configuration occurring depends on the number of failed Bernoulli bond connections. To write this out mathematically, we let n_c be the number closed bonds in b and $n_o(\sigma)$ is the number of closed bonds in b between equal spins in the prior configuration σ. Then $\mathbb{P}(b|\sigma) = (1-p)^{n_o(\sigma)}p^{n_c}$ and $\mathbb{P}(b|\sigma') = (1-p)^{n_o(\sigma')}p^{n_c}$ so that

$$\frac{\mathbb{P}(b|\sigma')}{\mathbb{P}(b|\sigma)} = \frac{(1-p)^{n_o(\sigma')}}{(1-p)^{n_o(\sigma)}} = (1-p)^{n_o(\sigma')-n_o(\sigma)}$$

The next step is to see that the number of closed bonds only dependent on b and not on σ or σ', we can therefore write

$$n_o(\sigma') - n_o(\sigma) = (n_o(\sigma') + n_c) - (n_o(\sigma) + n_c) = \sum_{<i,j>} \delta_{\sigma'_i,\sigma'_j} - \sum_{<i,j>} \delta_{\sigma_i,\sigma_j},$$

where $< i, j >$ loops through nearest neighbours i and j. We now have that

$$\frac{\mathbb{P}(b|\sigma')}{\mathbb{P}(b|\sigma)} = \mathrm{e}^{-2\beta J\left(\sum_{<i,j>}\delta_{\sigma'_i,\sigma'_j} - \delta_{\sigma_i,\sigma_j}\right)}$$

As a last step, notice that

$$\begin{aligned}
\Delta E &= -\sum_{<i,j>} J\left(\sigma'_i\sigma'_j - \sigma_i\sigma_j\right) \\
&= -\sum_{<i,j>} J\left[\delta_{\sigma'_i,\sigma'_j} - \left(1 - \delta_{\sigma'_i,\sigma'_j}\right) - \delta_{\sigma_i,\sigma_j} + \left(1 - \delta_{\sigma_i,\sigma_j}\right)\right] \\
&= -2\sum_{<i,j>} J\left(\delta_{\sigma'_i,\sigma'_j} - \delta_{\sigma_i,\sigma_j}\right)
\end{aligned}$$

such that

$$\frac{\mathbb{P}(b|\sigma')}{\mathbb{P}(b|\sigma)} = \mathrm{e}^{-\beta\Delta E},$$

which proves detailed balance.

As opposed to the Metropolis algorithm, the Swendsen-Wang algorithm suffers a lot less from critical slowing-down. As published by Swendsen and Wang themselves, an upper bound for the critical exponent is $z = 0.35$ [15] as opposed to the Metropolis algorithm $z = 2.14$. This is a result that we will attempt to reproduce.

3.2.2 Critical exponent z

A full understanding of the critical exponents would require many pages of theory and is not that much of interest to this thesis. However, a good source to read up on the matter would be Computational Physics [16, pp. 186-192]. It is however important to understand how the critical exponent z is defined.

Critical exponents are a tool to describe the behaviour of a model near a phase transition. They are universal, which means that they do not depend on the details of the interaction. For example, one could add next-nearest neighbour interactions without changing the exponents. However, there are also dynamic critical exponents, which are dependent on the algorithm used to simulate the model, instead of the model itself. These exponents are dependent on dynamical quantities, as well as perhaps other critical exponents. The critical exponent z is defined by the following relation [2]:

$$\tau \sim \xi^z \tag{34}$$

τ is called the relaxation time and ξ is the correlation length. This τ is of great interest to us, as it is in fact a number that tells us how many steps of the algorithm it takes for a quantity that is not in equilibrium to return to equilibrium. ξ on the other hand is the correlation length, and it is a measure for how far the nodes will 'respond' to what other nodes are doing. The correlation length is defined in terms of the two-point correlation function between two spins σ_i and σ_j:

$$\langle(\sigma_i - \langle\sigma_i\rangle)(\sigma_j - \langle\sigma_j\rangle)\rangle = e^{-|i-j|/\xi}$$

Note here that the correlation length ξ is also determined by a critical exponent relation, namely $\xi \sim |\beta - \beta_{crt}|^{-\nu}$, where $\nu = 1$. This is the reason why it converges around criticality. At high temperatures, we have totally chaotic states, so one can imagine this length being very low. On the other hand, at low temperatures we also have a low correlation length, perhaps surprisingly. That is because nodes in that regime are so strongly connected to their neighbours, that there is hardly any long range interaction. At criticality however, it is infinite, meaning that every spin flip will be felt and reacted to in every other node. Since we are obviously simulating on a finite grid, the correlation length cannot not infinite, so close to the critical point, it is replaced by the length of the grid L. This gives us the relation we use for simulation purposes:

$$\tau \sim L^z. \tag{35}$$

Notice that this quantity explicitly size scaling to the relaxation time, and to be clear, a low relaxation time is exactly what you want from an algorithm, as it means that you converge to an equilibrium state in approximately that amount of timesteps. Therefore, a low value of z means favorable size scaling. With this in mind, it should be clear that Swendsen and Wang's proposed $z = 0.35$ is a massive improvement over the standard Metropolis-Hastings method with $z = 2.14$.

This τ turns out to be quite difficult to pin down, as there are multiple definitions for it. We look at the two definitions for τ that we will be using. The first is the integrated correlation time τ_{int}. For a quantity A, the integrated correlation time is defined as:

$$\tau_{int} = \frac{1}{2} + \sum_{t=1}^{\infty} \rho_A(t), \quad \rho_A(t) = \frac{c_{AA}(t)}{c_{AA}(0)}, \tag{36}$$

where ρ is called the auto-correlation function and $c_{AA}(t)$ is the time correlation function of A with itself and it is defined as $c_{AA}(t) = \langle A_n A_{n+t} \rangle - \langle A_n \rangle^2$. This definition is very handy, as it is easy to extract out of simulation data. Notice here that the longer the simulation, the more accurate this sum will be. However, there is a slight problem with its calculation and that is that we must truncate the sum, since the statistics are bad for large t. It should be the case that ρ_A is monotonically decreasing in t, as the algorithm does not produce any loops. However, for large t there is a sort of aliasing effect, which makes the value of ρ_A increase. Therefore a cut-off point t_{max} to the sum of equation 36, suggested by [5], is the point where $\rho_A(t_{max} + 1) > \rho_A(t_{max})$, that being the point where ρ stops being monotonic.

Another interesting definition for the relaxation time is τ_{exp} or the exponential relaxation time. This one is defined as follows:

$$\tau_{exp} = \lim_{t \to \infty} \frac{t}{\rho_A(t)} \tag{37}$$

This quantity represents the relaxation time of the slowest decaying mechanism in the system [5, 16]. We can approximate this quantity by curve-fitting to the following relationship:

$$\rho_A(t) \sim \mathrm{e}^{-t/\tau_{exp}},$$

where we make sure that t is as large as possible, but not exceeding t_{max} from the previous part.

Once values for τ_{exp} and/or τ_{int} for multiple sizes have been obtained, it is a simple matter of curve-fitting the data to equation 35 to find z.

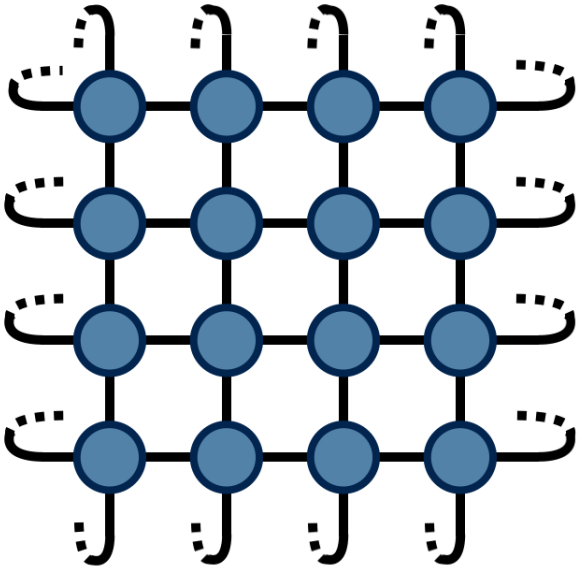


Figure 2: Graphical representation of periodic boundary conditions.

3.3 Programming cluster-based algorithms

We will now go over the process of programming a cluster-based algorithm, like the Swendsen-Wang algorithm. For the results of this paper, Python was used, so the the explanation will be biased towards using that language and code snippets will be in Python. Assumptions that will be made in this section, are that $J_{ij} = J = 1$ and the lattice that will be used is square and has periodic boundary conditions, meaning that bonds can form between opposite edges of the system. We will also assume that the external magnetic field $h = 0$. Furthermore, we will focus on a 2D system, but it is very easy to generalize the code to higher dimensions. We make these assumptions since most of these assumptions will also be made in for the results.

The strategy is simple: first, initialize the space we work on, then program the necessary tools to deal with clusters and lastly, run the cluster algorithm.

To initialize the environment, we start by setting all parameters of the model we want to use, being the inverse temperature β, the couplings which choose as $J = 1$. Since the behaviour is determined by βJ, we can do this without loss of generality. Next, we choose the shape of the model, which in our case is a square lattice of size L. Next, since we are working on a 2D lattice, we can store the spins in an L array and the bonds in a 2 matrix, where the first dimension stores the bonds for each cardinal direction.

An important remark to make is that the type of boundary conditions will influence the outcome of simulations. The most common one is periodic, for which the opposite ends of the lattice are connected by bonds. This is visualized in figure 2.

One can choose different boundary condition, but this will affect the behaviour of the parts of the lattice that are less than a correlation length away from an edge.

Next, the most important tool of a cluster-based algorithm, is an algorithm that can identify clusters of equal spins. This is not that trivial, since we only have information about connections between neighbours, and nothing more. The problem of finding connected components or clusters in a lattice (or any undirected graph) is well-studied and there exist many algorithms which do this. The one we use is called the 'depth first search' algorithm or DFS. The procedure is quite simple:

- Initialize each node as 'not visited'.
- Pick any node that is not visited and add it to a stack.
- Next, repeat the following procedure
    1. Mark the top node of the stack as visited and remove it from the stack
    2. Create a list of all nodes that are connected to the node that was just removed from the stack
    3. Add any non-visited node from that list to the top of the stack

Once the stack is empty, you have spanned out the cluster the original node was in. One can repeat this process for every node to get a list of all the clusters.

With this tool, we can easily carry out the steps of the Swendsen-Wang algorithm as described in section 3.2.1.

3.4 Overview of the simulations

Now that we have established how the simulations work, we can look at what kind of simulations that were carried out. First of all we present the dynamics of the Swendsen-Wang algorithm by simply looking at plots of a few timesteps of the algorithm. Afterwards we do a series of simulations on different sized lattices to investigate how well we can approximate the theoretical values of thermodynamic variables. Finally we do a number of very long simulations to find the critical exponent z.

It is important to set expectations of these simulations in order to judge the results properly. The 2D Ising model has been studied extensively. From the analytical solution [9] we know that the critical temperature $\beta_{crt} = \frac{\ln(1+\sqrt{2})}{2} \approx 0.4407$ and we know that below this temperature there is an ordered phase with spontaneous magnetization, and above the critical temperature is a disordered, chaotic phase. The solution of Yang [17] tells us that this spontaneous magnetization is described by the following relationship:

$$M(\beta) = \left(1 - \sinh\left(2\beta J\right)^{-4}\right)^{\frac{1}{8}} \tag{38}$$

Note that the spontaneous magnetization is the absolute value of the average spin per site. This equation only holds as the number of nodes in the systems goes to infinity, so we can expect the simulations to converge towards this distribution as the size increases. Notice here that M is the absolute value of the average of the spins. We prefer to

avoid working with absolute values of quantities, therefore we consider a closely related quantity, being the magnetic susceptibility χ' per site. This quantity can be calculated as the variation of the average spin $S = \frac{1}{L^2}\sum_i \sigma_i$:

$$\chi' = \mathrm{var}(S) = \frac{1}{L^4}(\langle S^2\rangle - \langle S\rangle^2)$$

Note that in the low temperature regime, $\langle S\rangle$ will constantly swap signs during the simulation. Therefore taking the average over time will equal 0. This is not supposed to happen according to thermodynamics, so it is more or less a flaw of the simulation method. This quantity is hard to control, so we prefer not to use it. Instead we consider the unsubtracted magnetic susceptibility per site χ

$$\chi = \frac{1}{L^4}(\langle S^2\rangle)$$

However, instead of computing the magnetic susceptibility like this, we can use the so-called improved estimator for this quantity [16, pp 495-496], which has the same average, but a smaller variance. It is defined as

$$\chi = \frac{1}{L^4}\left\langle \sum_c N_c^2 \right\rangle \tag{39}$$

where c are all clusters on the lattice and N_c represents the area or number of nodes in each cluster. This estimator is based on a principle described by Sweeny [14], namely that averaging over the clusters instead of the spins improves the statistics dramatically. That is because each cluster could have been entirely of spin 1 or spins -1 with equal probability, so a bond configuration essentially holds information of $2^{\#c}$ possible outcomes. Anything which is measured from the bond configuration directly, represents an average over all possible realisations of spin configurations.

Lastly, it is important to know that the susceptibility is supposed to behave as $\chi \sim (\beta_{crt} - \beta)^{-\gamma}$ for $\beta < \beta_{crt}c$ where γ is a critical exponent, which is $\frac{7}{4}$ for the 2D Ising model. This something we will check with simulations.

Lastly, the original paper of Swendsen and Wang stated that an upper bound to the critical exponent of the Swendsen-Wang algorithm is $z = 0.35$, but there are many studies that show lower values ranging from 0.17 to 0.28 and it has been conjectured that this exponent should grow to 0 [4] for larger and larger lattice sizes.

4 Results

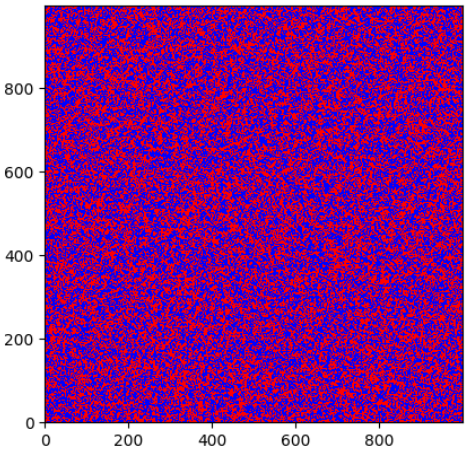
4.1 Swendsen-Wang dynamics

The first results we present, is a basic simulation of 10 timesteps of an Ising model simulation, using the Swendsen-Wang algorithm. The results can be found in figure 3. The simulations was done on a 1000×1000 lattice at $\beta = 0.6$, so a subcritical temperature. This means we expect a highly ordered system. And this is exactly what we observe: we start with randomly chosen spin states for each node and after a few timesteps, we evolve to an ordered system. Some other things to note are that we are clearly in the magnetized phase, but even in timestep 10, we still have small specks of blue, so it is not yet fully magnetized. Furthermore, notice that every timestep from 4 and higher, the dominant color flips. This is more or less a coincidence, however it should be noted that a drastic number of spin flips is expected behaviour in Swendsen-Wang dynamics. Recall that every timestep, the spin is chosen again per cluster, which is allowed, because the Ising Hamiltonian $\mathbf{H}_\Lambda$ from definition 2.1 is symmetric under spin reversal when $h = 0$. Lastly, the critical eye will notice that this simulation was not done with periodic boundaries, but with free boundaries. This has no particular reason other than that it was just the way the algorithm was implemented at the time of creating these images. All later simulations following this will be with periodic boundaries, as that is generally favoured to reduce finite-size effects.

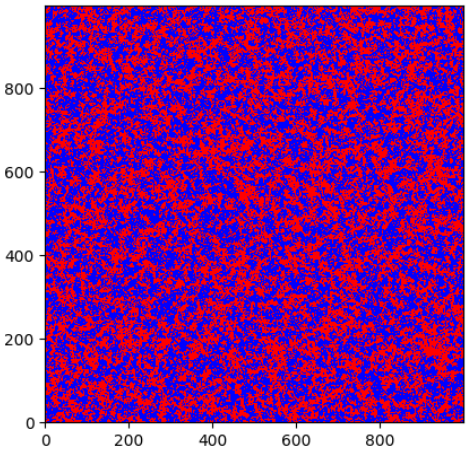
4.2 Magnetic susceptibility of the Ising model

In this section we present the results of a series of simulations where we use simulations to investigate how closely we can approximate the unsubtracted magnetic susceptibility χ, which we will simply call susceptibility from now on, of the Ising model. We will estimate this quantity using the improved estimator we showed in (39). To do this we run the model at various ranges of inverse temperature β and at each timestep we calculate the susceptibility and we take the average of all those to get an estimate of the expected magnetization at that temperature. Note here that we omit the first 30 timesteps, in order to have no influence from the initial configuration. The choice of the number of timesteps to omit is somewhat arbitrary, but it should at least be a few times the decorrelation time τ. The maximum value we measured for this quantity (results of which will be shown later) was around 7.5, so we chase to omit $4 \cdot 7.5 = 30$ timesteps. Furthermore, we will compare the results of the simulations with the theoretical values for the Ising model in the thermodynamical limit. We know that (38) gives us the magnetization for β values larger than the β_{crt} and that for the subcritical values, it should be 0. Since the unsubtracted susceptibility we are working with is simply the square of the magnetization, we can compare our results to the square of that function.

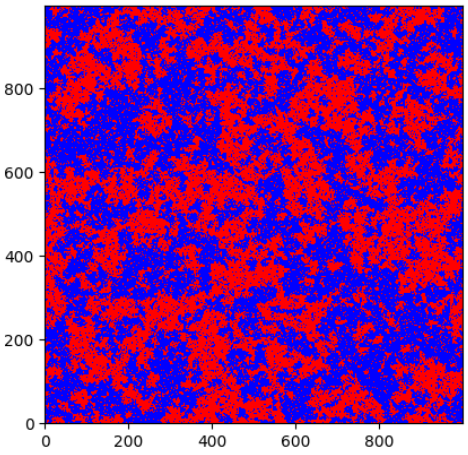
An important remark is that in order to get good estimates of the uncertainty in an averaged quantity, like the susceptibility, we cannot simply apply the normal formula to calculate the sample standard deviation. That is because the observables we are working with, is are correlated to their neighbours. In order to decorrelate the data, we take a so



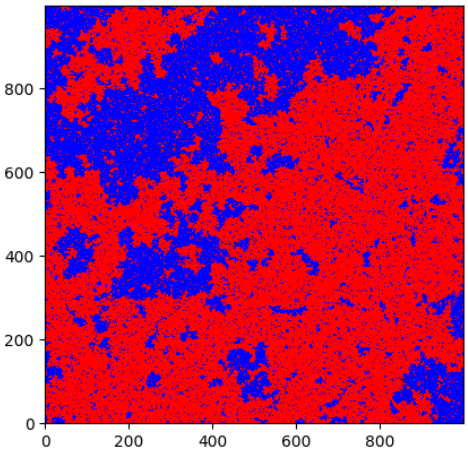
(a) Timestep 1



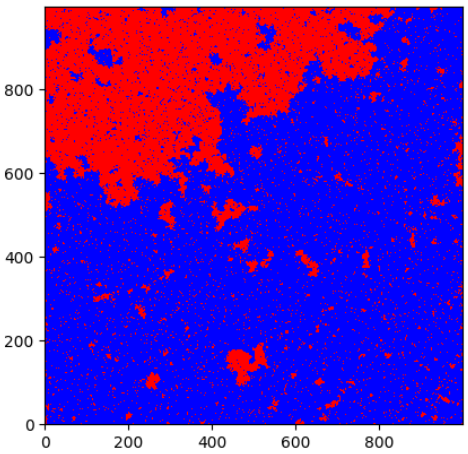
(b) Timestep 2



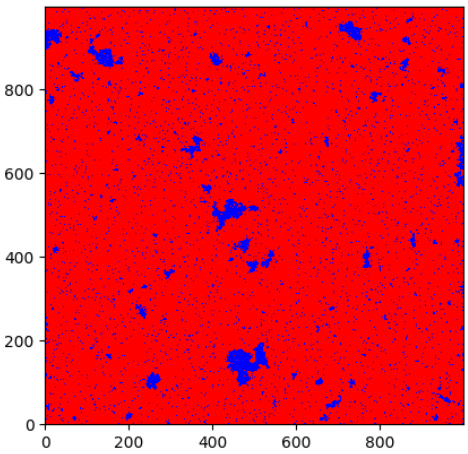
(c) Timestep 3



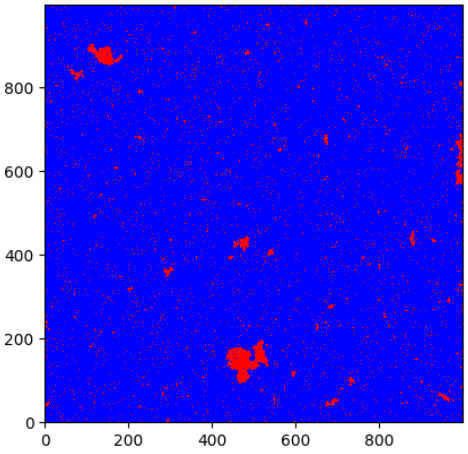
(d) Timestep 4



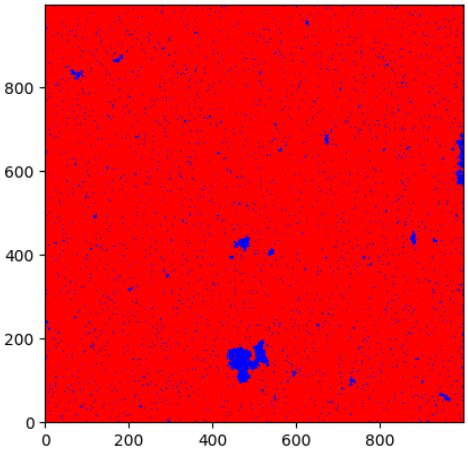
(e) Timestep 5



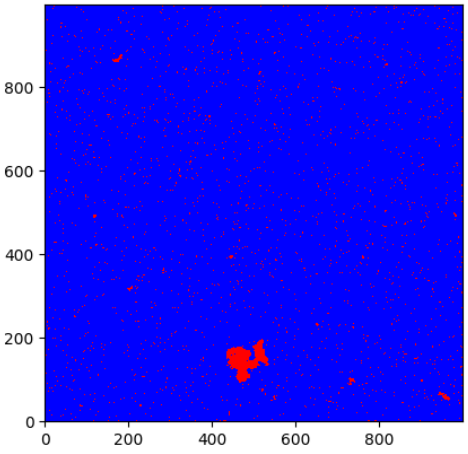
(f) Timestep 6



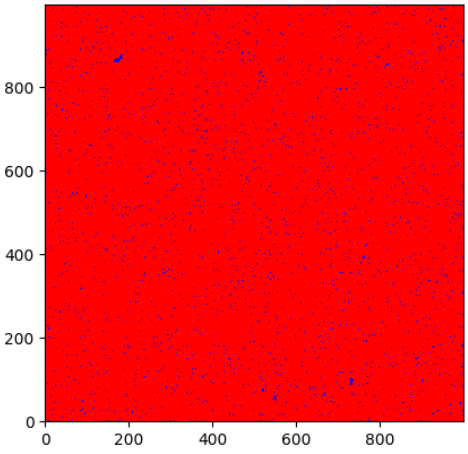
(g) Timestep 7



(h) Timestep 8



(i) Timestep 9



Timestep 10

Figure 3: First 10 iterations on a Swendsen-Wang simulation. This simulation was done on a 1000x1000 lattice with an inverse temperature $\beta = 0.6$.

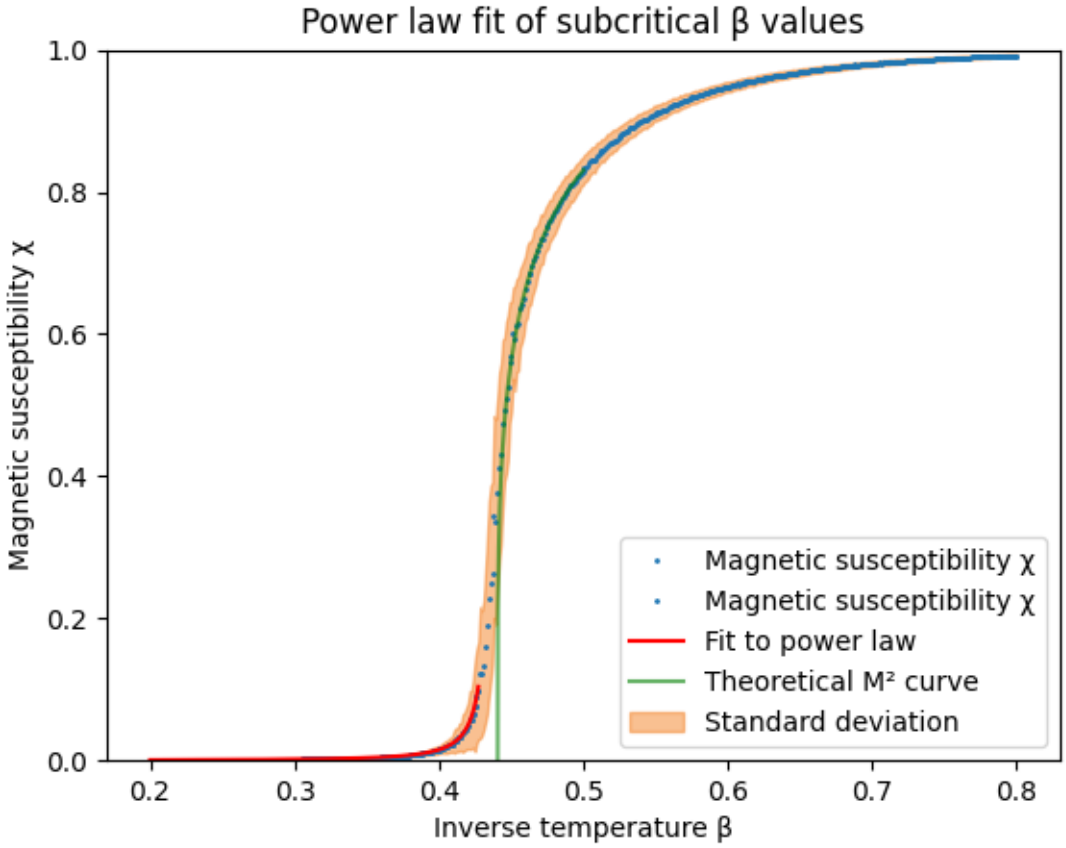


Figure 4: Simulation of the susceptibility over a range of temperatures. The simulations were done on a 64×64 lattice for 1000 timesteps. The β values range between 0.2 and 0.8 with 500 points in between. The data left of the phase transition was fitted to a power law, while the data right of the critical point has the theoretical curve for M^2 as a reference. The orange area represents the uncertainty in the measurements and has a width of 1 standard deviation.

called block average, where we split the data into blocks of a certain size and then we calculate the sample standard deviation with that formula. Note that the block length must be bigger than the decorrelation time, but small enough so that there are enough statistics. In our case, all calculated standard deviations are with a block average of width 20.

We start by presenting the results of a simulation over a range of β-values in figure 4. This plot is the result of many different simulations on 64×64 lattices on 500 equally spaced temperatures between $\beta = 0.2$ and $\beta = 0.8$. Each simulation was done for 1000 timesteps. It is clear that for β's that are a lot lower or higher than the critical temperature. However, right around the critical point, the results are quite bad compared to the results of the thermodynamical limit. We have obvious finite-size scaling effects and large standard deviations. As discussed before, this is what is expected to happen. We expect a power law relationship for subcritical β's that scales according to

$$\chi \sim a(\beta_{crt} - \beta)^{-\gamma}, \quad \beta < \beta_{crt}. \tag{40}$$

According to the theory of critical exponents, $\gamma = \frac{7}{4}$. We filled our data for the

susceptibility to a power law, seen in the figure as the red curve, to find out how close we can get to the theoretical $\gamma = \frac{7}{4}$. The value we fitted in this simulation, was $\gamma = 1.732 \pm 0.001$. Notice however that the fit is cut off at a certain point. That is because there is small region, close to the critical temperature, where the values are no longer described by the power law and also not yet on the theoretical curve. This is the region where the correlation size is greater that the lattice length. Since this region is not described by a power law anymore, we want to exclude this region from the fit. To find a good estimate for this cut-off point, we use another critical exponent scaling law for the correlation length. We know that the correlation length of the 2D Ising model scales like

$$\xi = c \cdot (\beta_{crt} - \beta)^{-1}. \tag{41}$$

The point where the finite size starts to heavily influence the results, is when $\xi = L/2$. Plugging this into this relationship, we get that

$$\beta_{cutoff} = \beta_{crt} - \frac{2a}{L}.$$

Without calculating the correlation length and then fitting it to (41), we cannot get a good estimate of what this constant c should be. The best we can do is to visually find some values such that cut-off point matches with the point where the graph starts to bend off the fitted power law. The conclusion was that for $c \geqslant 1$, the fit parameter for γ stayed nearly constant, but for values $c \leqslant 1$, the results got visibly worse. Therefore, taking $a = 1$ appeared to be a good choice.

With the results of the last simulation in mind, our goal is now to do a series of simulations on increasing lattice sizes and find out whether the power law scaling stays at a similar exponent and also to see how much the finite-size scaling effects decrease with larger simulation sizes.

To do this, we ran simulations on sizes $L \in \{32, 64, 128, 256, 512, 1024\}$. The simulations were all done on $[0.2, 0.5]$ with 500 equally spaced temperatures and they all ran for 500 timesteps. The range of temperatures was decreased as it was obvious from the previous simulation that extreme temperatures are not of interest to this experiment. The results of these simulations were combined in one plot so that the effect of finite size scaling is clearly visible. This plot is shown in figure 5 and the least square fit parameters for these simulations are given in table 1.

L	a	γ
32	2.44e-4 $\pm$0.01e-4	1.715 ± 0.002
64	5.94e-5 $\pm$0.01e-5	1.730 ± 0.001
128	1.479e-5 $\pm$0.001e-5	1.7327 ± 0.0006
256	3.693e-6 $\pm$0.002e-6	1.7334 ± 0.0003
512	9.224e-7 $\pm$0.003e-7	1.7340 ± 0.0002
1024	2.3067e-7 $\pm$0.0006e-7	1.7337 ± 0.0002

Table 1: These are the results of the least square fits of the simulations with different lattice sizes L fitted to (40).

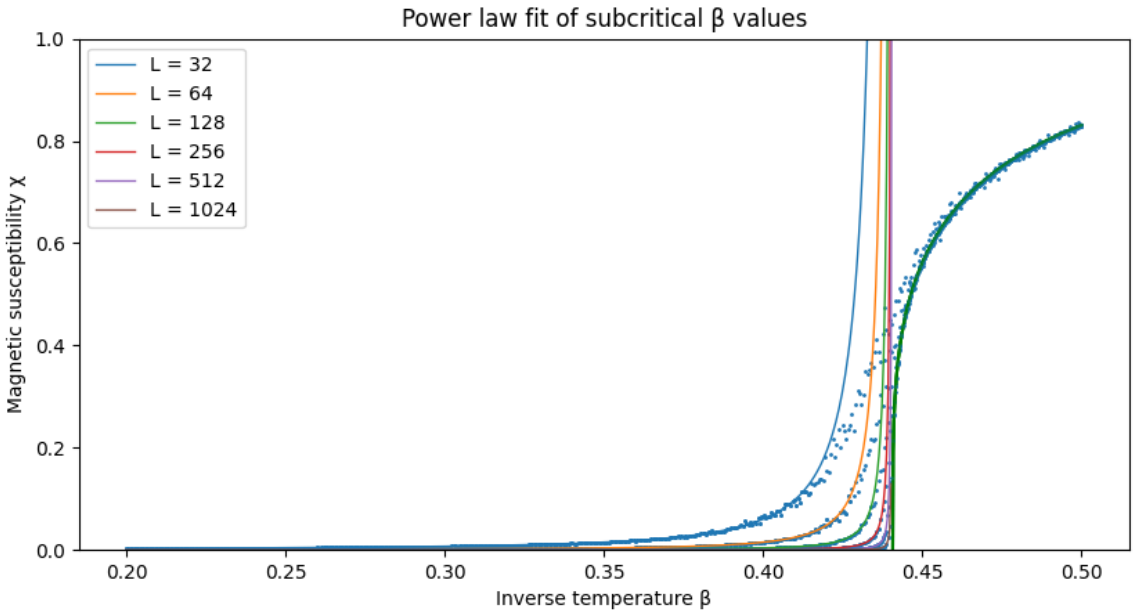


Figure 5: Results of simulations of magnetic susceptibility χ on lattice sizes 32×32, 64×64, 128×128, 256×256, 512×512, 1024×1024. The used β values range between 0.2 and 0.5 and each simulation ran for 500 timesteps.

First, we note that the amplitude a of each fit scales with $1/L^2$, which is expected as we are measuring the susceptibility *per spin*. Next, notice that there is a clear sign of convergence towards the theoretical distribution with increasing size. The results become more and more precise and there is very little spread between the last three values. However, instead of converging to the theoretical value of 1.75, it stagnates at around 1.734, which is 1% off the theoretical value of 1.75. It appears that we can get close to it, but not quite there. Due to the clear sign of convergence, we cannot explain this as a finite-size effect. There are a different number of possible reasons for this. One of them is it has to do with the way the boundaries are set. As discussed before, we chose an upper bound β_{cutoff}, which was definitely a necessity to produce consistent results. However, we the only lower bound that was set, was the minimal β-value of all simulations, which is 0.2. So the range on which the power law was fitted, was $[0.2, \beta_{cutoff}(L)]$. To investigate the effect, we increased the lower bound by removing more and more β-values and fitted to the power law at each step. In figure 6, there is a plot of these γ-values on reduced intervals. The data that was used for this, is from the 512×512 simulations. These results seem to get us a lot closer to $\gamma = 1.75$, but the data clearly becomes unreliable after a certain point, which is due to a lack of data after cutting off most of the data.

In order to optimize this lower bound even further, we made more simulations for β in the range of $[0.4, \beta_{cutoff}(L)]$ with 200 timesteps in this interval and on a 512×512 grid. The same process of removing one β at a time, that was done for figure 6, was applied to this data with the results displayed in figure 7. It is clear that in this plot, we converge even closer to the theoretical value. Furthermore, the theoretical value is already within the standard deviation for a lower bound of $\beta = 0.415$, which is clearly still within a

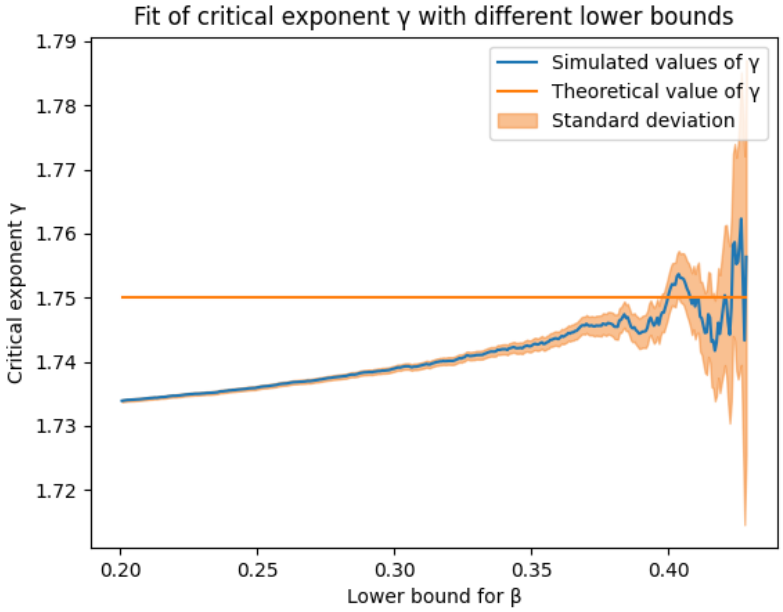


Figure 6: This plot shows the estimate of parameter γ, when the lower bound for β is steadily increased. The horizontal line shows the theoretical value for γ. The data used for this plot is from the previously done simulation with size $L = 512$.

region of stable uncertainties. More precisely, with a lower bound of $\beta = 0.415$, we find $\gamma = 1.748 \pm 0.004$. We can therefore conclude that if we take data that is simulated in an appropriately close range to the critical temperature, we can find a critical exponent of $\gamma = 1.748 \pm 0.004$, which is within one standard deviation from the true value of $\gamma = 1.75$.

One could say that perhaps there is a confirmation bias in these results, since the lower bound is chosen quite arbitrarily and perhaps so that it meets our expectations. However, it should be noted that the critical exponents are supposed to hold only in the neighbourhood of the critical point. Furthermore, the fit for γ stays around 1.748 even after the point we chose as a minimum. The point where it starts to move away from 1.75, is clearly also the range where a lack of data starts to heavily influence the results. Therefore still accept this as a valid result.

4.3 Determining the dynamic critical exponent z

As explained in section 3.2.2, we need to fit a number of relaxation times τ to find the critical exponent z so we started by doing a number of very long simulations on different sizes. The simulations that were done, range between lattice size $L = 16$ to $L = 256$ and are all done at critical temperature. The precise specifications and results are presented in table 2. Note that the results are all an average of 3 simulations per size and for all these simulations, the values were calculated for τ_{int} andτ_{exp} for both the total energy E and magnetization M, giving us 4 different relaxation times.

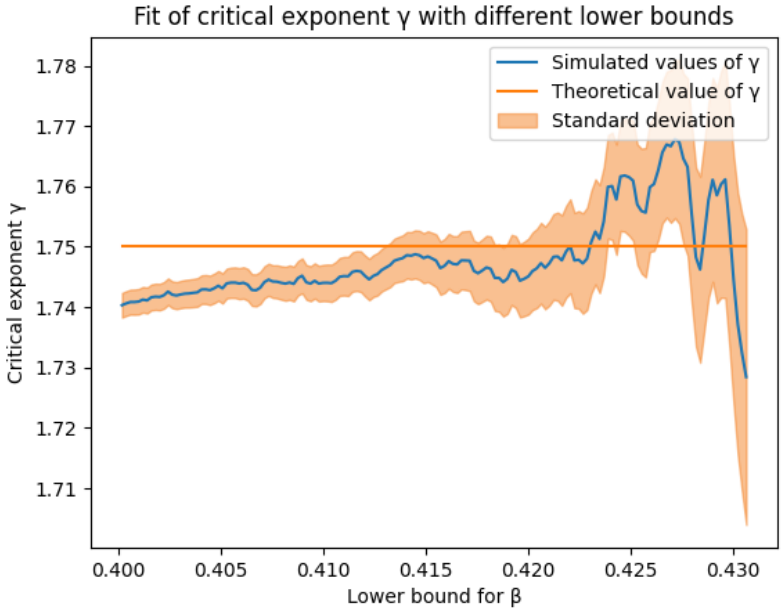


Figure 7: This plot shows the estimate of parameter γ, when the lower bound for β is steadily increased. The horizontal line shows the theoretical value that we are supposed to have. This time, we used improved the data by using a smaller timescale on a more concentrated interval.

Lattice size L	Timesteps	$\tau_{int,E}$	$\tau_{exp,E}$	$\tau_{int,M}$	$\tau_{exp,M}$
16	$5 * 10^5$	3.32 ± 0.05	3.29 ± 0.05	3.05 ± 0.05	3.24 ± 0.05
32	$5 * 10^5$	4.07 ± 0.05	4.17 ± 0.05	3.51 ± 0.05	4.12 ± 0.05
50	$5 * 10^5$	4.58 ± 0.05	4.60 ± 0.05	3.79 ± 0.05	4.53 ± 0.05
64	$5 * 10^5$	4.96 ± 0.05	5.11 ± 0.05	3.99 ± 0.05	4.99 ± 0.05
100	$3 * 10^6$	5.51 ± 0.05	5.80 ± 0.05	4.33 ± 0.05	5.62 ± 0.05
128	$2 * 10^5$	5.93 ± 0.05	6.22 ± 0.05	4.56 ± 0.05	6.03 ± 0.05
256	$7 * 10^5$	7.02 ± 0.05	7.56 ± 0.05	5.07 ± 0.05	7.20 ± 0.05

Table 2: This table contains the results of simulations to calculate the relaxation times τ_{int} and τ_{exp} for the energy E and the magnetization M. Each result for those τ's is the average of 3 separate simulations and the number of timesteps in each of those simulations is given in column 2.

Note that each value of τ has an uncertainty of 0.05. It does not represent the standard deviation within each simulation size, because we only had 3 simulations per τ. Instead, it was assumed that all simulations have a similar deviation. Therefore we took the uncertainty of all simulated values to be the standard deviation of the difference of each simulation with its mean value in the table. This gave us a data set of size 72 values instead of 24 sets of size 3. Even though this data was mostly generated in to measure dynamic critical exponent z, it has an important meaning of its own, as explained in section 3.2.2 on critical exponents. We used this data in order to get an estimate of the block size average and to find how many values should be omitted at the start of a simulation.

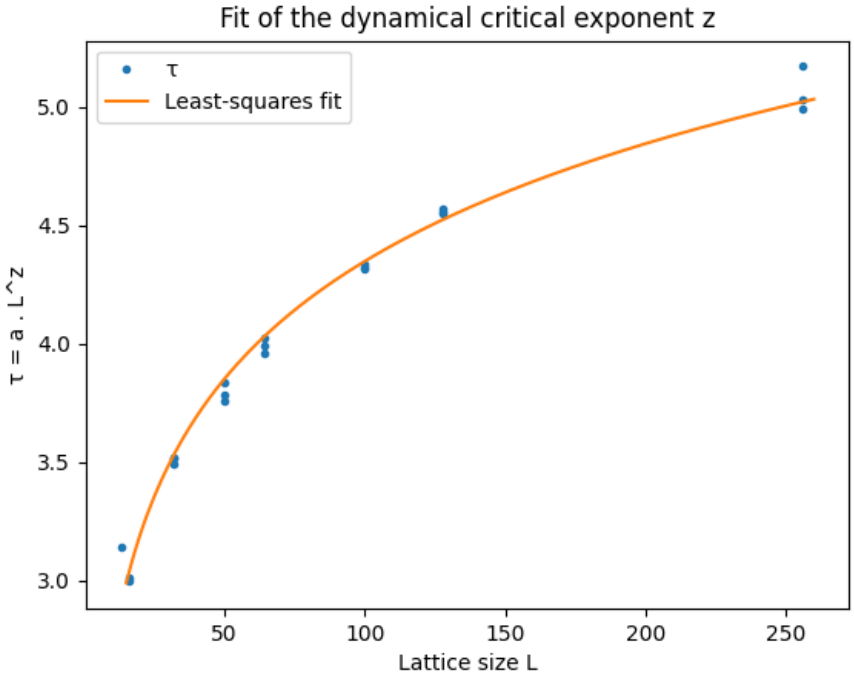


Figure 8: Plot of the curve fit in order to find critical exponent z. The data of $\tau_{int,M}$ was used for this particular fit. The resulting value of z was 0.184±0.004.

We now go over to the results of the simulations to find the critical exponent z. As explained before, this is done by fitting the τ values to

$$\tau = aL^z$$

where a is a constant of proportionality. In figure 8 you can see the results one such fits for $\tau_{int,M}$. This result is definitely a success, as we can see a clear power law scaling. Furthermore, the found value of z is 0.184, which is a lot lower than the value of 0.35 which was originally reported by Swendsen and Wang [15].

The rest of the fits for z are not shown as a plot, but we present the values in table 3. This data agrees well with results found in [1], which used very similar techniques. Furthermore, it is clear that the value of $z = 0.35$ reported by Swendsen and Wang [15] is too high. It appears to be the case that the larger the range of lattice sizes L in the simulations are, the smaller the z value becomes.

	z
$\tau_{int,E}$	0.263 ± 0.004
$\tau_{exp,E}$	0.292 ± 0.005
$\tau_{int,M}$	0.180 ± 0.004
$\tau_{exp,M}$	0.277 ± 0.005

Table 3: Dynamic critical exponents z for the integrated and exponential relaxation times for both energy E and magnetization M.

Another remarkable thing to note is that the z value for $\tau_{int,M}$ is very different compared to the rest of the z values. Even though there is no reason why they should all be very close, it is still interesting to see this. The conclusion to be made is that somehow the magnetization relaxes, on average, a lot faster that other observables, although the exponential relaxation time scales just like the rest of the relaxation times.

It has also been hypothesized by Heermann and Burkitt [4] that the integrated and exponential relaxation times scale logarithmically, which has since been partly corroborated by Du, Zheng and Wang [2]. They found that the $\tau_{exp,E}$ scales logarithmically, and that $\tau_{int,E}$ stay practically constant from a certain size of around $L = 256$ and onward. However, their method is different as they measure the initial decay towards equilibrium starting from either an ordered state or a random state, the choice of which appears to have very little influence on the results. We sadly cannot compare results as they used simulation sizes up to 8192, which is not feasible to reproduce on the home computer, that was used for our simulation.

5 Conclusion

In this thesis, we introduced the path integral representation of exponential operators as a tool to derive the FK-representation for the Ising model. Along the way, we made sure to include proofs, most of which are original. These results are related to the Swendsen-Wang algorithm, which is based on the FK-representation. With this algorithm, we ran many simulations with the goal of calculating two critical exponents. The first one was γ which is related to the magnetic susceptibility χ and the second one was z, which is the dynamical critical exponent with which we showed that this algorithm is more efficient near criticality.

The results of the simulation of γ were generally good, even though we had to take our data from a specific neighbourhood around the critical temperature. At first, the result appeared to be $\gamma = 1.7337(3)$ which was the result for simulations of lattice size $L = 256$, $L = 512$ and $L = 1024$. However, forcing the lower bound closer and closer to the critical point, gave us increasingly better results. This is because the critical exponent scaling relation only holds close to the critical point. In the end we concluded that when letting the lower bound approach critical point, while ensuring that there is enough data to take an accurate fit, the simulated critical exponent γ approaches its true value of 1.75. With this in mind, we did extra simulations with increased resolution around the critical point, and we found a value of $\gamma = 1.748 \pm 0.004$.

Next, we did a range of very long simulations in order to find different relaxation times τ. We used different definitions of τ, namely τ_{int} and τ_{exp}, to find the relaxation times of the magnetization M and total energy E. These results themselves were not particularly of great interest to our study. We did however find that the values for $\tau_{int,M}$ were considerably lower than the rest, which means that, on average, the magnetization decorrelates faster than other observables. The main purpose of these simulations was to estimate the dynamic critical exponent z. For this we found values ranging between 0.180 and 0.292, depending on the thermodynamic variable used to generate them. This is lower than the value originally reported by Swendsen and Wang, but is in line with other more recent studies of this subject. It has been conjectured that this exponent should be 0 and that the scaling is either logarithmic or constant, depending on the variable considered. However, many more orders of magnitude need to be considered to get an accurate depiction of whether this is true. Therefore, this study cannot deny nor corroborate this hypothesis.

References

[1] Paul D. Coddington and Clive F. Baillie. *The Dynamical Critical Exponents of the Swendsen-Wang Algorithm for the 2D Ising Model*.

[2] Jianqing Du, Bo Zheng and Jian-Sheng Wang. 'Dynamic critical exponents for Swendsen–Wang and Wolff algorithms obtained by a nonequilibrium relaxation method'. In: *Journal of Statistical Mechanics: Theory and Experiment* 2006.05 (May 2006), P05004–P05004. URL: https://doi.org/10.1088%2F1742-5468%2F2006%2F05%2Fp05004.

[3] Geoffrey R. Grimmett. '1.2 Random-cluster model'. In: *The random-cluster model*. Springer, 2006.

[4] Dieter W. Heermann and Anthony N. Burkitt. 'System size dependence of the autocorrelation time for the Swendsen-Wang Ising model'. In: *Physica A: Statistical Mechanics and its Applications* 162.2 (1990), pp. 210–214. ISSN: 0378-4371. URL: https://www.sciencedirect.com/science/article/pii/037843719090439Y.

[5] M. Hennecke and U. Heyken. 'Critical dynamics of cluster algorithms in the dilute Ising model'. In: *Journal of Statistical Physics* 72.3-4 (Aug. 1993), pp. 829–844.

[6] Dmitry Ioffe. 'Stochastic Geometry of Classical and Quantum Ising Models'. In: Mar. 2009. ISBN: 978-3-540-92795-2.

[7] M.R. Leadbetter, G. Lindgren and H. Rootzen. *Extremes and Related Properties of Random Sequences and Processes*. Springer Series in Statistics. Springer New York, 2012. ISBN: 9781461254492. URL: https://books.google.be/books?id=-ofTBwAAQBAJ.

[8] M.E.J. Newman and G.T. Barkema. *Monte Carlo Methods in Statistical Physics*. Clarendon Press, 1999. ISBN: 9780198517979. URL: https://books.google.be/books?id=kQN6DwAAQBAJ.

[9] L Onsager. 'A two-dimensional model with an order-disorder transition (crystal statistics I)'. In: *Phys. Rev* 65.3-4 (1944).

[10] Rudolf Peierls. 'On Ising's model of ferromagnetism'. In: *Mathematical Proceedings of the Cambridge Philosophical Society*. Vol. 32. 3. Cambridge University Press. 1936.

[11] M. Reed and B. Simon. 'I: Functional Analysis'. In: Methods of Modern Mathematical Physics. Elsevier Science, 1981, p. 295. ISBN: 9780080570488. URL: https://books.google.nl/books?id=rpFTTjxOYpsC.

[12] Dalton A R Sakthivadivel. 'Magnetisation and Mean Field Theory in the Ising Model'. In: *SciPost Phys. Lect. Notes* (2022), p. 35. URL: https://scipost.org/10.21468/SciPostPhysLectNotes.35.

[13] D. Stauffer. 'Social applications of two-dimensional Ising models'. In: *American Journal of Physics* 76.4 (Apr. 2008), pp. 470–473.

[14] Mark Sweeny. 'Monte Carlo study of weighted percolation clusters relevant to the Potts models'. In: *Phys. Rev. B* 27 (7 Apr. 1983), pp. 4445–4455. URL: https://link.aps.org/doi/10.1103/PhysRevB.27.4445.

[15] Robert H. Swendsen and Jian-Sheng Wang. 'Nonuniversal critical dynamics in Monte Carlo simulations'. In: *Phys. Rev. Lett.* 58 (2 Jan. 1987), pp. 86–88. URL: https://link.aps.org/doi/10.1103/PhysRevLett.58.86.

[16] Jos Thijssen. *Computational Physics*. 2nd ed. Cambridge University Press, 2007.

[17] C. N. Yang. 'The Spontaneous Magnetization of a Two-Dimensional Ising Model'. In: *Phys. Rev.* 85 (5 Mar. 1952), pp. 808–816. URL: https://link.aps.org/doi/10.1103/PhysRev.85.808.

Appendix A

Proof. (based on [11]) We start by setting

$$T_n = \mathrm{e}^{\sum_{i=1}^{k} A_i/n} \quad \text{and} \quad S_n = \prod_{i=1}^{k} e^{A_i/n}.$$

Then,

$$T_n^n - S_n^n = \sum_{l=0}^{n-1} \left(T_n^{n-l} S_n^l - T_n^{n-l-1} S_n^{l+1} \right) = \sum_{l=0}^{n-1} T_n^{n-l-1} \left(T_n - S_n \right) S_n^l$$

This is a clever telescoping sum to rewrite the difference $T_n^n - S_n^n$. Now, we can bound this with:

$$\begin{aligned} \left\| S_n^n - T_n^n \right\| &\leqslant n \left(\max \left\{ \left\| S_n \right\|, \left\| T_n \right\| \right\} \right)^{n-1} \left\| S_n - T_n \right\| \\ &\leqslant n \left\| S_n - T_n \right\| \exp(\sum_{i=1}^{k} \left\| A_i \right\|) \end{aligned}$$

This holds because

$$\begin{aligned} \left\| S_n \right\|^{n-1} = \left\| \prod_{i=1}^{k} e^{A_i/n} \right\|^{n-1} &\leqslant \left(\prod_{i=1}^{k} \left\| e^{A_i/n} \right\| \right)^{n-1} && \text{(matrix norm is sub-multiplicative)} \\ &\leqslant \left(\prod_{i=1}^{k} e^{\left\| A_i \right\|} \right)^{n-1/n} && \text{(triangle inequality)} \\ &\leqslant e^{\sum_{i=1}^{k} \left\| A_i \right\|} \end{aligned}$$

and

$$\begin{aligned} \left\| T_n \right\|^{n-1} = \left\| \mathrm{e}^{\sum_{i=1}^{k} A_i/n} \right\|^{n-1} &\leqslant \left(\mathrm{e}^{\sum_{i=1}^{k} \left\| A_i \right\|} \right)^{(n-1)/n} && \text{(triangle inequality)} \\ &\leqslant \mathrm{e}^{\sum_{i=1}^{k} \left\| A_i \right\|}. \end{aligned}$$

Finally,

$$\left\| T_n - S_n \right\| = \left\| \sum_{l=0}^{\infty} \frac{1}{l!} \left(\sum_{i=1}^{k} \frac{A_i}{n} \right)^l - \prod_{i=1}^{k} \left\{ \sum_{l=0}^{\infty} \frac{1}{l!} \left(\frac{A_i}{n} \right)^l \right\} \right\| = \mathcal{O}\left(\frac{1}{n^2} \right)$$

We now see that $\left\| S_n^n - T_n^n \right\| \to 0$ in the limit, which proves the theorem.

□

Appendix B

In this appendix we prove Lemma 2.5.

Lemma 2.5. *Let $\xi_1^\Delta, ..., \xi_m^\Delta$ be Bernoulli processes on $(0, \beta]$ with spacing Δ and probability of arrival $p = \lambda_l \Delta$. Then*

$$\lim_{\Delta \to 0} \mathbb{P}\left(\forall p, q \in \{1, ..., m\} : \xi_p^\Delta \cap \xi_q^\Delta = \emptyset\right) = 1.$$

Proof. We start by rewriting the event to have disjointedness to an equivalent event, namely

$$\mathbb{P}\left(\forall p, q \in \{1, ..., m\} : \xi_p^\Delta \cap \xi_q^\Delta = \emptyset\right) = \mathbb{P}\left(\bigcap_{j=1}^{\beta/\Delta} \left[\sum_{l=1}^{m} \xi_l^\Delta((j\Delta - \Delta/2, j\Delta + \Delta/2)) \leqslant 1\right]\right).$$

Here we split the disjointedness condition in time and say that at each time, no more than one arrival can occur, which is an equivalent way to describe disjointedness of the arrival times. Notice that all timesteps are independent, so we have

$$\mathbb{P}\left(\bigcap_{j=1}^{\beta/\Delta} \left[\sum_{l=1}^{m} \xi_l^\Delta((j\Delta - \Delta/2, j\Delta + \Delta/2)) \leqslant 1\right]\right) = \prod_{j=1}^{\beta/\Delta} \mathbb{P}\left(\left[\sum_{l=1}^{m} \xi_l^\Delta((j\Delta - \Delta/2, j\Delta + \Delta/2)) \leqslant 1\right]\right).$$

Next, we take $\lambda = \min(\lambda_1, ..., \lambda_m)$, so now we can give a lower bound to this probability

$$\begin{aligned}\prod_{j=1}^{\beta/\Delta} \mathbb{P}\left(\left[\sum_{l=1}^{m} \xi_l^\Delta((j\Delta - \frac{\Delta}{2}, j\Delta + \frac{\Delta}{2})) \leqslant 1\right]\right) &\geqslant \prod_{j=1}^{\beta/\Delta} \left((1-\Delta\lambda)^m + \binom{m}{1}\Delta\lambda(1-\Delta\lambda)^{m-1}\right) \\ &= \left((1-\Delta\lambda)^m + m\Delta\lambda(1-\Delta\lambda)^{m-1}\right)^{\beta/\Delta} \\ &= L(\Delta),\end{aligned}$$

where $L(\Delta)$ is just a shorter way to write the function we ended up with. We now take the limit of this function and apply L'Hopital's rule to find that the limit goes to 1. The calculations are done as follows

$$\begin{aligned}&\lim_{\Delta\to 0} L(\Delta) = \exp\left(\ln\left(\lim_{\Delta\to 0}\left((1-\Delta\lambda)^m + m\Delta\lambda(1-\Delta\lambda)^{m-1}\right)^{\beta/\Delta}\right)\right) \\ &= \exp\left(\lim_{\Delta\to 0} \beta \frac{\ln\left((1-\Delta\lambda)^m + m\Delta\lambda(1-\Delta\lambda)^{m-1}\right)}{\Delta}\right) \\ &\overset{L'H.}{=} \exp\left(\lim_{\Delta\to 0} \beta \frac{-m\lambda(1-\Delta\lambda)^{m-1} - m(m-1)\Delta\lambda^2(1-\Delta\lambda)^{m-2} + m\lambda(1-\Delta\lambda)^{m-1}}{(1-\Delta\lambda)^m + m\Delta\lambda(1-\Delta\lambda)^{m-1}}\right) \\ &= \exp\left(\lim_{\Delta\to 0} \beta \frac{-m(m-1)\Delta\lambda^2(1-\Delta\lambda)^{m-2}}{(1-\Delta\lambda)^m + m\Delta\lambda(1-\Delta\lambda)^{m-1}}\right) = \exp(0) = 1.\end{aligned}$$

Recall that this was only a lower bound to the probability of the lemma, but since a probability cannot be greater than 1, we have proven the lemma. □

Appendix C

Theorem 5.1 (Product Expansion Formula)**.** *Let $A_1, A_2, ..., A_n$ be self-adjoint matrices and Ψ, Ψ' two vectors in $\mathbb{X}$. Then,*

$$\langle\Psi|A_1...A_n|\Psi'\rangle = \sum_{\Psi_{i_1},...\Psi_{i_{n-1}}} \langle\Psi|A_1|\Psi_{i_1}\rangle\langle\Psi_{i_1}|A_2|\Psi_{i_2}\rangle...\langle\Psi_{i_{n-1}}|A_n|\Psi'\rangle, \tag{42}$$

where the Ψ_{i_l} run through all elements of the basis $\{\Psi_i\}$ for all $l = 1, ..., n-1$.

Proof. We will prove this by induction, starting at $n = 2$, so that

$$\langle\Psi|A_1A_2|\Psi'\rangle = \sum_{\Psi_{i_1}\in\{\Psi_i\}} \langle\Psi|A_1|\Psi_{i_1}\rangle\langle\Psi_{i_1}|A_2|\Psi'\rangle,$$

For this case, first notice that

$$\begin{aligned} A\Psi &= A\sum_{\Psi_i\in\{\Psi_i\}} \langle\Psi|\Psi_i\rangle\Psi_i \\ &= \sum_{\Psi_i\in\{\Psi_i\}} \langle A\Psi|\Psi_i\rangle\Psi_i \\ &= \sum_{\Psi_i\in\{\Psi_i\}} \langle\Psi|A|\Psi_i\rangle\Psi_i, \end{aligned}$$

where the last equality holds because A is self-adjoint. Next, we can transpose this statement to get

$$\langle\Psi|A = \sum_{\Psi_i\in\{\Psi_i\}} \langle\Psi|A|\Psi_i\rangle\langle\Psi_i|.$$

Now taking $A = A_1$ and multiplying by $A_2|\Psi'\rangle$ gives the statement we wanted to prove, so the case for $n = 2$ holds.

Next, we assume $n = k$ holds and we prove $n = k + 1$. So we have

$$\langle\Psi|A_1...A_k|\Psi'\rangle = \sum_{\Psi_{i_1},...\Psi_{i_{k-1}}} \langle\Psi|A_1|\Psi_{i_1}\rangle\langle\Psi_{i_1}|A_2|\Psi_{i_2}\rangle...\langle\Psi_{i_{k-1}}|A_k|\Psi'\rangle, \tag{43}$$

and we want to show that

$$\langle\Psi|A_1...A_kA_{k+1}|\Psi'\rangle = \sum_{\Psi_{i_1},...\Psi_{i_k}} \langle\Psi|A_1|\Psi_{i_1}\rangle\langle\Psi_{i_1}|A_2|\Psi_{i_2}\rangle...\langle\Psi_{i_{k-1}}|A_k|\Psi_{i_k}\rangle\langle\Psi_{i_k}|A_{k+1}|\Psi'\rangle.$$

Notice here that we have no restrictions on Ψ, so the proof for the case $n = 2$ is valid for any $\Psi_\alpha \in \mathbb{X}$. We now take $\langle\Psi|A_1...A_{k-1} = \langle\Psi_\alpha|$. Then by 43

$$\langle\Psi_\alpha| = \sum_{\Psi_{i_1},...\Psi_{i_{k-1}}} \langle\Psi|A_1|\Psi_{i_1}\rangle\langle\Psi_{i_1}|A_2|\Psi_{i_2}\rangle...\langle\Psi_{i_{k-1}}|.$$

Now we apply the case of $n = 2$ to $\langle \Psi_\alpha | A_k A_{k+1} | \Psi' \rangle$ to get that

$$\langle \Psi_\alpha | A_k A_{k+1} | \Psi' \rangle = \sum_{\Psi_{i_k} \in \{\Psi_i\}} \langle \Psi_\alpha | A_k | \Psi_{i_k} \rangle \langle \Psi_{i_k} | A_{k+1} | \Psi' \rangle.$$

Then simply substitute Ψ_α back out and we get

$$\begin{aligned} \langle \Psi_\alpha | A_k A_{k+1} | \Psi' \rangle &= \langle \Psi | A_1 ... A_k A_{k+1} | \Psi' \rangle \\ &= \sum_{\Psi_{i_1}, ... \Psi_{i_k}} \langle \Psi | A_1 | \Psi_{i_1} \rangle ... \langle \Psi_{i_{k-1}} | A_k | \Psi_{i_k} \rangle \langle \Psi_{i_k} | A_{k+1} | \Psi' \rangle, \end{aligned}$$

which is exactly what we wanted to show. This proves the theorem. □

Appendix D

We now prove the rest of Theorem 2.8.

Proof of (ii). We want to prove that arrival times are disjoint. To do this we consider the following limit

$$\lim_{\epsilon \to 0} \mathbb{P}\left(\sum_{l=1}^{m} \underline{\xi}_l(t-\epsilon, t+\epsilon] = 1 \middle| \exists t \in \xi : \sum_{l=1}^{m} \underline{\xi}_l(t-\epsilon, t+\epsilon] \geqslant 1 \right)$$

By definition of conditional probability, we can rewrite this probability as follows

$$\mathbb{P}\left(\sum_{l=1}^{m} \underline{\xi}_l(t-\epsilon, t+\epsilon] = 1 \middle| \exists t \in \xi : \sum_{l=1}^{m} \underline{\xi}_l(t-\epsilon, t+\epsilon] \geqslant 1 \right) = \frac{\mathbb{P}\left(\sum_{l=1}^{m} \underline{\xi}_l(t-\epsilon, t+\epsilon] = 1 \cap \exists t \in \xi : \sum_{l=1}^{m} \underline{\xi}_l(t-\epsilon, t+\epsilon] \geqslant 1 \right)}{\mathbb{P}\left(\exists t \in \xi : \sum_{l=1}^{m} \underline{\xi}_l(t-\epsilon, t+\epsilon] \geqslant 1 \right)}.$$

It is clear that for the event in the numerator, the left part of the intersection is a subset of the right part, so that the intersection should be just the left part. Note that to keep the expressions a bit more compact, we substitute $t_\epsilon = (t-\epsilon, t+\epsilon]$ We have now simplified the limit to the following expression

$$\frac{\mathbb{P}\left(\sum_{l=1}^{m} \underline{\xi}_l(t_\epsilon) = 1 \right)}{\mathbb{P}\left(\exists t \in \xi : \sum_{l=1}^{m} \underline{\xi}_l(t_\epsilon) \geqslant 1 \right)} \triangleq \frac{\mathbb{P}(A)}{\mathbb{P}(B)}$$

We now calculate the probabilities of the numerator and denominator separately. We start with the numerator

$$\begin{aligned} \mathbb{P}(A) &= \mathbb{P}\left(\sum_{l=1}^{m} \underline{\xi}_l(t_\epsilon) = 1 \right) \\ &= \sum_{l=1}^{m} \mathbb{P}\left(\underline{\xi}_1(t_\epsilon) = 0, ..., \underline{\xi}_{l-1}(t_\epsilon) = 0, \underline{\xi}_l(t_\epsilon) = 1, \underline{\xi}_{l+1}(t_\epsilon) = 0, ..., \underline{\xi}_m(t_\epsilon) = 0 \right). \end{aligned}$$

By independence of these events, we can write this as

$$\begin{aligned}
\mathbb{P}(A) &= \sum_{l=1}^{m} \mathbb{P}\left(\underline{\xi}_l(t_\epsilon) = 1\right) \prod_{\substack{k=1 \\ k\neq l}}^{m} \mathbb{P}\left(\underline{\xi}_k(t_\epsilon) = 0\right) \\
&= \sum_{l=1}^{m} 2\epsilon\lambda_l \exp\left(-2\epsilon\lambda_l\right) \cdot \exp\left(-2\epsilon \sum_{k\neq l} \lambda_k\right) \\
&= 2\epsilon \left(\sum_{l=1}^{m} \lambda_l\right) \exp\left(2\epsilon \left(\sum_{l=1}^{m} \lambda_l\right)\right)
\end{aligned}$$

Let us now call the sum of all λ_l's, $L = \sum_{l=1}^{m} \lambda_l$, so that we get:

$$\mathbb{P}(A) = 2\epsilon L \mathrm{e}^{-2\epsilon L}$$

Now we evaluate the denominator:

$$\begin{aligned}
\mathbb{P}(B) &= \mathbb{P}\left(\exists t \in \xi : \sum_{l=1}^{m} \underline{\xi}_l(t_\epsilon) \geqslant 1\right) \\
&= 1 - \mathbb{P}\left(\sum_{l=1}^{m} \underline{\xi}_l(t_\epsilon) = 0\right) \\
&= 1 - \prod_{l=1}^{m} \mathbb{P}\left(\underline{\xi}_l(t_\epsilon) = 0\right) \\
&= 1 - \prod_{l=1}^{m} \mathrm{e}^{-2\epsilon\lambda_l} \\
&= 1 - \mathrm{e}^{-2\epsilon L}
\end{aligned}$$

where L is once again $\sum_{l=1}^{m} \lambda_l$. Now, we can evaluate the limit using L'Hopital's rule:

$$\lim_{\epsilon\to 0} \frac{\mathbb{P}(A)}{\mathbb{P}(B)} = \lim_{\epsilon\to 0} \frac{2\epsilon L \mathrm{e}^{-2\epsilon L}}{1 - \mathrm{e}^{-2\epsilon L}} \overset{L'H.}{=} \lim_{\epsilon\to 0} \frac{2L\mathrm{e}^{-2\epsilon L} - 4\epsilon L^2 \mathrm{e}^{-2\epsilon L}}{-2L\mathrm{e}^{2\epsilon L}} = \frac{2L}{2L} = 1$$

This proves that we have disjoint arrival times almost surely. □

Proof of (iii). From *(ii)*, we know that arrival types are well defined. Next, we bring our attention to the following limit

$$\lim_{\epsilon\to 0} \mathbb{P}\left(\xi_l(t-\epsilon, t+\epsilon] = 1 | \exists\ l' : \xi_{l'}(t-\epsilon, t+\epsilon] = 1\right).$$

Here it is important to note that for the conditioning, we do not require the $\exists!$ operator, nor do we have to consider the case where $\xi_{l'}(t-\epsilon, t+\epsilon] \geqslant 1$, because in the previous part, we have already proven that arrival times are disjoint almost surely. We now use the same strategy as in the proof of the previous part. Next, we substitute $(t-\epsilon, t+\epsilon] = t_\epsilon$ and we call $[\xi_l(t_\epsilon) \geqslant 1] = A$ and $[\exists\ l' : \xi_{l'}(t_\epsilon) \geqslant 1] = B$ and use that

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}.$$

Now notice that A is a subset of B, and therefore $A \cap B = A$, so that

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A)}{\mathbb{P}(B)}.$$

We start by calculating $\mathbb{P}(A)$:

$$\begin{aligned}\mathbb{P}(A) &= \mathbb{P}(\xi_l(t_\epsilon) = 1) \\ &= 1 - \mathbb{P}(\xi_l(t_\epsilon) = 0) \\ &= 1 - \mathrm{e}^{-2\epsilon\lambda_l}\end{aligned}$$

Next, we look at $\mathbb{P}(B)$:

$$\begin{aligned}\mathbb{P}(B) &= \mathbb{P}(\exists\ l' : \xi_{l'}(t_\epsilon) = 1) \\ &= 1 - \mathbb{P}(\forall\ l' : \xi_{l'}(t_\epsilon) = 0) \\ &= 1 - \prod_{l=1}^{m} \mathrm{e}^{-2\epsilon\lambda_l} \\ &= 1 - \mathrm{e}^{-2\epsilon\sum_{l=1}^{m}\lambda_l}\end{aligned}$$

Now, we look at the limit of their quotient and apply L'Hopital's rule to find the following:

$$\begin{aligned}\lim_{\epsilon\to 0}\frac{\mathbb{P}(A)}{\mathbb{P}(B)} &= \lim_{\epsilon\to 0}\frac{1-\mathrm{e}^{-2\epsilon\lambda_l}}{1-\mathrm{e}^{-2\epsilon\sum_{l=1}^{m}\lambda_l}} \\ &\overset{L'H.}{=} \lim_{\epsilon\to 0}\frac{-2\lambda_l\mathrm{e}^{-2\epsilon\lambda_l}}{-2\left(\sum_{l=1}^{m}\lambda_l\right)\mathrm{e}^{-2\epsilon\lambda_l}} \\ &= \frac{\lambda_l}{\lambda_1 + ... + \lambda_m}\end{aligned}$$

□