# Path Integral approach to the 2D Ising model through the Worm Algorithm

Alberto Cavallar and Enric Simó
*Bachelor's Degree in Engineering Physics*
(Polytechnic University of Catalonia)

(Dated: June 3, 2021)

The aim of this article is to present a Path Integral Monte Carlo method known as the Worm Algorithm (WA) and analyze its promising performance by studying a particular problem. For this purpose, the well-known 2D Ising model on a square lattice with periodic boundary conditions and without external field is considered and its alternative path integral formulation is presented. The WA is implemented and analyzed by comparing it with the classical Metropolis algorithm, which is known to show slowing-down near the critical temperature.

## I. INTRODUCTION

Theoretical studies often involve mappings of the original system onto an equivalent (with regards to the final answer for some property) description in terms of abstract mathematical or graphical objects. Path integrals, high temperature expansions and Feynman diagrams are well-known examples [1].

Under mapping one has to deal with the infinite-dimensional configuration space having complex topology and non-local constraints, which severely reduce efficiency of Monte Carlo simulations based on standard local updates. This sometimes leads to ergodicity problems in large systems when the entire configuration space can not be sampled in a reasonable computation time. Also a somewhat related difficulty facing conventional Monte Carlo schemes is the computation of off-diagonal correlation functions since they have no direct relation to the configuration space of the partition function.

In what follows we present the path integral formulation of the 2D Ising model and the implementation of this approach through the WA dealing with the above mentioned constraints by going to the enlarged configuration space. A performance analysis of the WA is intended by comparing it with the classical Metropolis algorithm, whose convergence strongly slows down near the critical region (critical slow-down effect).

## II. THE 2D ISING MODEL

The 2D Ising model, aiming to study the behavior of ferromagnetic materials, considers an $L \times L$ square lattice with (toroidal) periodic boundary conditions, whose nodes represent atomic spins $\sigma$ that can be found in two states represented by the values ±1 and are allowed to interact only with their neighbours. Although we will work in the absence of external field $h$, the general hamiltonian for this model reads

$$\mathcal{H} = -J \sum_{\langle ij \rangle} \sigma_i \sigma_j - h \sum_i \sigma_i \tag{1}$$

where $J$ is the coupling constant or bond strength in the lattice and $\langle ij \rangle$ refers to a pairwise interaction between nearest neighbours. Parallel spins present a lower energy than antiparallel ones. While the system tends to the lowest energy, heat disturbs this tendency creating thus the possibility of different structural phases. The model allows the identification of a phase transition characterized by a critical temperature $T_c$.

According to this hamiltonian (from now on $h = 0$), the partition function $\mathcal{Z}$ can be written as:

$$\mathcal{Z} = \sum_{\{\sigma_i\}} e^{-\beta \mathcal{H}} = \sum_{\{\sigma_i\}} e^{K \sum_{\langle ij \rangle} \sigma_i \sigma_j} = \sum_{\{\sigma_i\}} \prod_{\langle ij \rangle} e^{K \sigma_i \sigma_j} \tag{2}$$

where $\{\sigma_i\}$ denotes all possible configurations and where $K := \beta J$ is defined, being $\beta = 1/k_B T$ [6].

According to Onsager's exact solution, there exists a phase transition at a given critical temperature

$$T_c = \frac{2}{\ln(1+\sqrt{2})} \tag{3}$$

At high temperatures spins are random and uncorrelated, but as the temperature is lowered the interactions between them encourage nearby spins to point in the same direction, giving rise to correlations. Groups of adjacent spins which are correlated (tend to point in the same direction) are called clusters. As we approach $T_c$, the typical size $\xi$ of these clusters, known as correlation length, diverges giving place to arbitrarily large areas in which the spins are pointing mostly up or mostly down.

## III. METROPOLIS ALGORITHM

The Ising model becomes unapproachable to evaluate numerically when $L \gg 1$ so that the number $2^L$ of possible states in the system becomes large. This fact motivates the reason for simulating the Ising model using Monte Carlo methods.

Classically, the Metropolis algorithm is implemented, where at each step a randomly selected spin is proposed to be flipped with probability

$$W[\{\sigma_i\}_{old} \longrightarrow \{\sigma_i\}_{new}] = \begin{cases} 1 & \Delta E < 0 \\ e^{-\beta \Delta E} & \Delta E \geq 0 \end{cases}$$

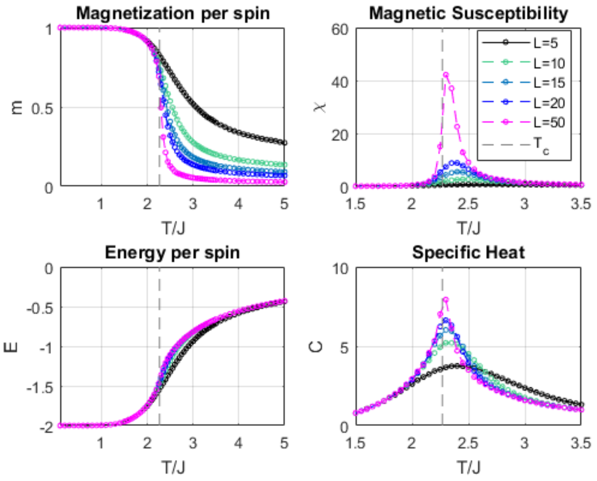

FIG. 1. Mean value of magnetization $m$ and energy $E$ per spin along with the magnetic susceptibility and specific heat as a function of $T/J$ for $L = 5, 10, 15, 20, 50$, by Metropolis. The dashed line corresponds to critical temperature $T_c$.

where $\Delta E = E_{new} - E_{old}$. This guarantees ergodicity and that a stable equilibrium will be reached [2].

Implementing this algorithm on the 2D Ising model yields the results shown in Fig. 1, where the magnetization per spin ($m$), magnetic susceptibility ($\chi$), energy per spin ($E$) and specific heat ($C$) are computed as

$$m = \frac{1}{L^2}\sum_i \sigma_i, \qquad \chi = \frac{d\langle m\rangle}{dh} = \beta(\langle m^2\rangle - \langle m\rangle^2)$$
$$\langle E\rangle = \frac{1}{L^2}\frac{1}{\mathcal{Z}}\sum_{\{\sigma_i\}} e^{-\beta\mathcal{H}}, \quad C = \frac{d\langle E\rangle}{dT} = \beta^2(\langle E^2\rangle - \langle E\rangle^2)$$

Fig. 1 shows a phase transition at a critical temperature [7] which coincides with the theoretical one $T_c$, where $m$ and $E$ present the expected sudden change and $\chi$ and $C$ diverge.

## IV. FINITE SIZE SCALING

To better estimate the value of the critical temperature, one proceeds with the finite size scaling. In this context, a useful dimensionless quantity for accurately locating $T_c$ is the binder ratio

$$Q = \frac{\langle m^2\rangle}{\langle |m|\rangle^2}$$

which is size independent at $T_c$. Hence, the curves for different values of the size $L$ should intersect at a common point given by $T_c$, as is indeed shown in Fig. 2.

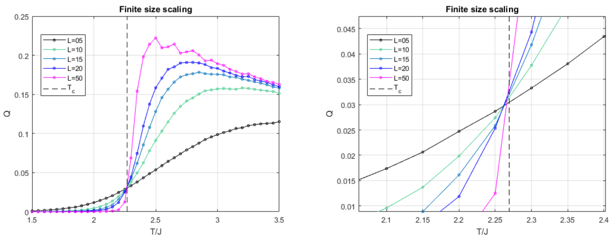

FIG. 2. Finite size scaling for Metropolis algorithm.

## V. PATH INTEGRAL FORMULATION

Metropolis scheme is usually the most universal and easy to program approach to Monte Carlo simulations. However, its advantages are virtually canceled out near phase transition points. In the following we present a method which essentially eliminates the critical slowing down problem and yet remains local. The corner stone of this approach is the possibility to introduce the configuration space of closed paths. Closed-path (CP) configurations may be then sampled very efficiently using Worm algorithm (presented in next section) for quantum statistical models in which closed trajectories naturally arise from imaginary-time evolution of world lines. In classical models the CP representation derives from high-temperature expansions for a broad class of lattice models.

Let us develop expression (2) by expanding the exponential in power series and rearranging terms [5]:

$$\begin{aligned}\mathcal{Z} &= \sum_{\{\sigma_i\}}\prod_{\langle ij\rangle} e^{K\sigma_i\sigma_j} = \sum_{\{\sigma_i\}}\prod_{b=\langle ij\rangle}\sum_{N_b=0}^{\infty}\frac{K^{N_b}}{N_b!}(\sigma_i\sigma_j)^{N_b} \\ &= \sum_{\{\sigma_i\}}\sum_{\{N_b\}}\prod_{b=\langle ij\rangle}\frac{K^{N_b}}{N_b!}(\sigma_i\sigma_j)^{N_b} \\ &= \sum_{\{N_b\}}\left(\prod_{b=\langle ij\rangle}\frac{K^{N_b}}{N_b!}\right)\left(\prod_{i=1}^{N}\sum_{\sigma_i=\pm1}\sigma_i^{\mathrm{L}_i}\right) \\ &= 2^N\sum_{\{N_b\}}^{loops}\prod_b\frac{K^{N_b}}{N_b!} = 2^N\sum_{\{N_b\}}^{loops}W[\{N_b\}].\end{aligned} \tag{4}$$

The variable $N_b$ in the power series has a deeper meaning and is called *bond number*. The *loops* label on the sum represents the constraint that the sum of all bond numbers "incident on every lattice site" $i$, $\mathrm{L}_i = \sum_{b=\langle ij\rangle} N_b$, has to be even; otherwise $\sum_{\sigma_i=\pm1}\sigma_i^{\mathrm{L}_i}$ is zero. These non-vanishing terms explain the leading factor $2^N$.

It is at this point where we notice that the possible combinations of $\{N_b\}$ can be mapped to a graphical representation which considers combinations of bonds of width $N_b$ between sites in the lattice (see Fig. 3 left), where the constraint that the sum $\mathrm{L}_i$ of all bond numbers incident on every site be even is translated into demanding that the allowed configurations of lines are that

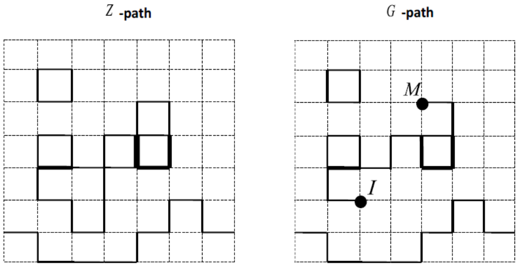

FIG. 3. Loop representations for the partition function and the correlation function for the 2D Ising model, where line thickness is proportional to $N_b$.

of closed un-oriented loops, since loops always contribute an even number to $\mathrm{L}_i = \sum_{b=\langle ij\rangle} N_b$.

This formulation justifies the loop graphical representation and the used terminology. However, in order to implement the WA, an alternative and more compact formulation can be obtained by using the following identity (recalling $\sigma_i\sigma_j = \pm 1$), since in this case only one line can be drawn on each bond [3]:

$$e^{K\sigma_i\sigma_j} = \frac{e^K + e^{-K}}{2} + \sigma_i\sigma_j\frac{e^K - e^{-K}}{2} = \cosh(K)(1+\mathcal{T}\sigma_i\sigma_j)$$

where we have denoted $\mathcal{T} := \tanh(K)$. Since in the 2D lattice with $N$ spins and periodic boundary conditions there are $2N$ bonds, the partition function becomes

$$\mathcal{Z} = \cosh^{2N}(K)\cdot 2^N\left(2^{-N}\sum_{\{\sigma_i\}}\prod_{\langle ij\rangle}(1+\mathcal{T}\sigma_i\sigma_j)\right) \quad (5)$$

where a $2^N$ factor has been taken out for convenience. Calling $\mathcal{Z}'$ the last term in brackets and expanding it (omitting the $2^{-N}$ factor for a moment) yields

$$\sum_{\sigma_1=\pm 1}\dots\sum_{\sigma_N=\pm 1}\left[1+\mathcal{T}\sum_{\ell=1}\sigma_i\sigma_j + \mathcal{T}^2\sum_{\ell=2}(\sigma_i\sigma_j)(\sigma_{i'}\sigma_{j'})+\dots\right]$$

where the sums $\sum_{\ell=\mathrm{L}}$ should be understood as the sum over all sets of bonds $\{\langle ij\rangle\}$ such that the link length in the set is L, being the link length the number of coupling terms $\sigma_i\sigma_j$.

Notice here that, again since $\sum_{\sigma_i=\pm 1}\sigma_i = 0$, only terms with an even link length number contribute to $\mathcal{Z}'$ (actually, only $\mathrm{L} = 0, 4, 6, \dots$). Call these terms closed, indicating that they represent a closed loop. The sum over all contributing terms gives a factor $2^N$, which cancels out the $2^{-N}$ one in $\mathcal{Z}'$. Rewriting $\mathcal{Z}'$ in terms of loop lengths gives

$$\mathcal{Z}' = \sum_{\mathrm{L}} g(\mathrm{L})\mathcal{T}^{\mathrm{L}} \quad (6)$$

where $g(\mathrm{L})$ is the number of loops with length L and corresponds to the correlation function over the square lattice. Thus, the partition function can finally be written as

$$\mathcal{Z} = 2^N\cosh^{2N}(K)\sum_{L} g(\mathrm{L})\mathcal{T}^{\mathrm{L}} \quad (7)$$

As mentioned, the improvement of (7) with respect to (4) is that we only consider $N_b \in \{0,1\}$ instead of $N_b = 0, 1, 2, \dots$, so only one line can be drawn on each bond.

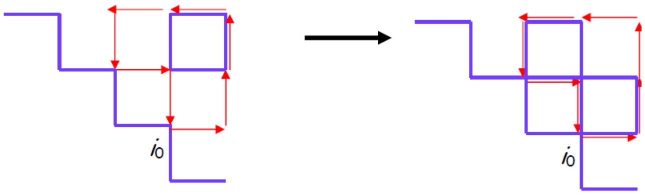

FIG. 4. Illustration of the worm steps forming a closed loop.

## VI. WORM ALGORITHM IMPLEMENTATION

We now proceed to describe the WA. As seen in (7), the partition function is a weighted sum of all closed loops, being $\mathcal{T}^{\mathrm{L}}$ the weight for a loop of length L. This is specially interesting since $\mathcal{T} = \tanh(K) \in [0,1]$ on $\mathbb{R}^+$. Thus, a probabilistic interpretation can be given to them (in contrast to weights in the first formulation (4), which require a proper fitting leading to a tougher algorithm).

Notice now that random closed loops can be sampled by *opening* a path from one fixed node and enlarging it with new bonds, each with a probability weight $\mathcal{T}$, till reaching the fixed node again thus closing the *worm*–loop of length L with total weight $\mathcal{T}^{\mathrm{L}}$. The fixed node is called *Ira* ($\mathcal{I}$) while the moving one *Masha* ($\mathcal{M}$). See Fig. 3.

In order to keep track of included bonds in the generation of a loop, we implement a $2N$–size array `bonds` being updated with 0/1 values denoting the presence/absence of a drawn bond. Further, an array structure `nbr` specifies the location of each of the four neighbours of each of the $N$ sites in the lattice as well as the location of the associated bond in `bonds`, being computed at the very beginning respecting the periodic boundary conditions.

The Worm Algorithm is implemented in bond– (and not spin–) space as follows, starting at $\mathcal{I} = \mathcal{M}$ [4]:

1. If $\mathcal{I} = \mathcal{M}$, add one loop to statistics. Erase the whole space by setting `bonds` to 0 and randomly select a site $\mathcal{I}_0$ to start a new loop setting $\mathcal{I} := \mathcal{I}_0$.

2. Select at random a direction to a neighbour $\mathcal{J}$ with equal probability.

3. Move $\mathcal{M}$ in that direction with probability $\tanh(K)^{1-N_b}$. If accepted, flip the bond variable $N_b := (N_b+1)\%2$ (i.e. $0 \leftrightarrow 1$) and set $\mathcal{M} := \mathcal{J}$. Increment statistics: $++G(\mathcal{M}-\mathcal{I})$.

4. If $\mathcal{I} = \mathcal{M}$ do step 1; otherwise go to step 2.

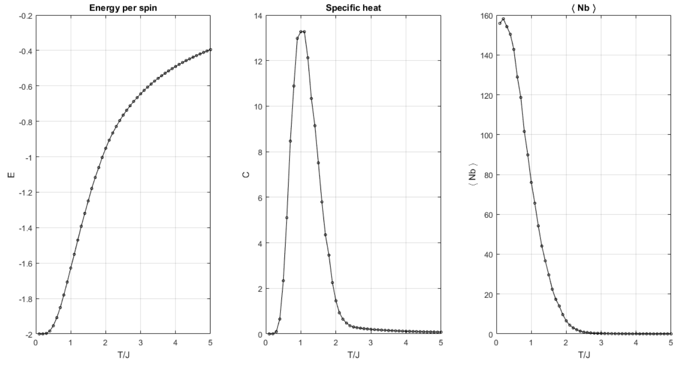

FIG. 5. Mean energy, heat capacity and mean loop length, respectively, as a function of $T/J$ for $L = 20$.

See illustrated procedure in Fig. 4. Notice that open loops correspond to the $G$–space configuration, while closed loops correspond to the $\mathcal{Z}$–space configuration and thus $G(0) \equiv \mathcal{Z}$. Further, the ratio $G(i-j)/G(0)$ gives the two–point correlation function $g(i-j)$.

After introducing the algorithm, we now deduce on our own the WA formulae for computing the magnitudes of interest, since in the whole found bibliography such results were presented without explanation and could not be clearly understood.

First, using the statistical result based on the partition function and cleverly manipulating the derived terms, we obtain an expression for the mean energy:

$$\langle E \rangle = \frac{-1}{\mathcal{Z}} \frac{d\mathcal{Z}}{d\beta} = ... = -J\mathcal{T} \left( 2N + \frac{\langle \ell \rangle}{\sinh(K)^2} \right) \tag{8}$$

being $\langle \ell \rangle$ the average loop length. From here follows after some work an expression for the heat capacity:

$$C = \frac{d\langle E \rangle}{dT} = ... = \frac{KJ}{T} \frac{\mathcal{T}^2}{\sinh^4(K)} \left[ \left( 2N \sinh^2(K) + \langle \ell \rangle \right) + \right. \\ \left. + \left( \langle \ell^2 \rangle - \langle \ell \rangle^2 \right) - 2\langle \ell \rangle \cosh^2(K) \right] \tag{9}$$

Since we are working in the absence of external magnetic field, the average magnetization $\langle m \rangle$ is zero (for each state there exists another one with opposite magnetization, namely the one with all spins flipped). Thus, the last quantity we need is the magnetic susceptibility, which can be obtained as

$$\chi = \frac{\partial \langle m \rangle}{\partial h} = \beta \left( \langle m^2 \rangle - \langle m \rangle^2 \right) = \beta \langle m^2 \rangle$$

which using the partition function and some smart manipulations yields:

$$\chi = \beta \sum_{i,j} g(i-j) \tag{10}$$

In the implementation, the two-point correlation function $g(i-j)$ is obtained by computing the number of times Masha has been at distance $i-j$ from Ira (being updated each time a bond is drawn/erased by ±1, respectively), divided by the total number of loops (tracked by the quantity $G(0)$).

## VII. CONCLUSIONS

Despite the much effort devoted to the project, we still have not obtained the desired results. However, we present the obtained ones and discuss several features we have been commenting on. Since the obtained magnetic susceptibility curve isn't of much interest, we present instead the average loop length (this time denoted as $\langle N_b \rangle$) along with the mean energy and heat capacity, as seen in Fig. 5. Observe that the energy curve ranges in $[-2, 0]$ and has a similar shape as in Fig. 1, although a second inspection shows the sudden slope change is appearing too early. Also the heat capacity presents a curve with a well defined peak, which however doesn't lie at the expected critical temperature $T_c \simeq 2.26$. Last to mention is the mean loop length curve, were we see that large loops are formed for low temperatures while hardly any for high ones, as was to be expected. Observe that this curve does indeed show what seems to be a critical behaviour at the desired temperature, suddenly showing that almost no loops are formed for larger temperatures than $T_c \simeq 2.26$. This seems to show that the algorithm is correctly implemented and that what is going wrong is something related to the energy expression, perhaps a numerical problem since the expression we found coincides with other bibliographical sources. Although unfortunately not in time, this problem may soon be corrected and thus the improvement of this algorithm with respect to the Metropolis one will be analyzed showing an expected error decrease, not to mention the already observed improvement in computation velocity as also the elimination of the critical slowing down near $T_c$ which Metropolis algorithm does present. Regarding computation analysis, it is worth to mention that Worm Algorithm's time has been estimated to run around ten times faster than Metropolis algorithm, an improvement which is mainly due to the fact that (opposed to Metropolis) no thermalization between states in WA is needed since averages are computed over independent loops and not over spin configurations.

## VIII. ACKNOWLEDGEMENTS

We would like to thank Jordi Boronat Medico for the both interesting and demanding proposed problem, as well as for his guidance along the project and proofreading of this manuscript.

[1] Nikolay Prokof'ev, Boris Svistunov, *Worm Algorithm for Problems of Quantum and Classical Statistics*. DOI: 0910.1393 (2010).
`https://arxiv.org/abs/0910.1393`

[2] N. Metropolis et al., *Equation of State Calculations by Fast Computing Machines*, J. Chemical Physics, Vol. 21, 1953, pp. 1087–1092.

[3] Simon Rydell, *Worm simulation of Hausdorff dimension of critical loop fluctuations* (2018).
`http://kth.diva-portal.org/smash/get/diva2:1266521/FULLTEXT01.pdf`

[4] Luo Jie, *Worm Alogorithm and its application* (2007).
`https://core.ac.uk/download/pdf/48625213.pdf`

[5] Marcin Szyniszewski, *Simulating graphene impurities using the worm algorithm.*
`https://www.researchgate.net/publication/235765980_Simulating_graphene_impurities_using_the_worm_algorithm`

[6] From now on and concerning the implementation of the Worm Algorithm, we will work in units of $T^* = k_B T/J$, although we will continue writing $T$ instead of $T^*$.

[7] Where a critical slow-down has indeed been observed when studying the convergence in the implemented Metropolis algorithm.