Monte Carlo simulation of the 2D Ising model

- tutorial -

Zoltán Néda

Babeş-Bolyai University

Department of Theoretical and
Computational Physics

١. A basic Metropolis Algorithm for simulating the 2D and 3D Ising model on square lattice → free boundary condition

٢. Implementing the periodic boundary condition

٣. Calculating averages as a function of temperature. Finite-size effects. Calculation of: $m(T)$ $C_V(T)$ $\chi(T)$

The Ising model

$$H = -J \sum_{<i,j>} S_i S_j - \mu h \sum_{\{i\}} S_i$$

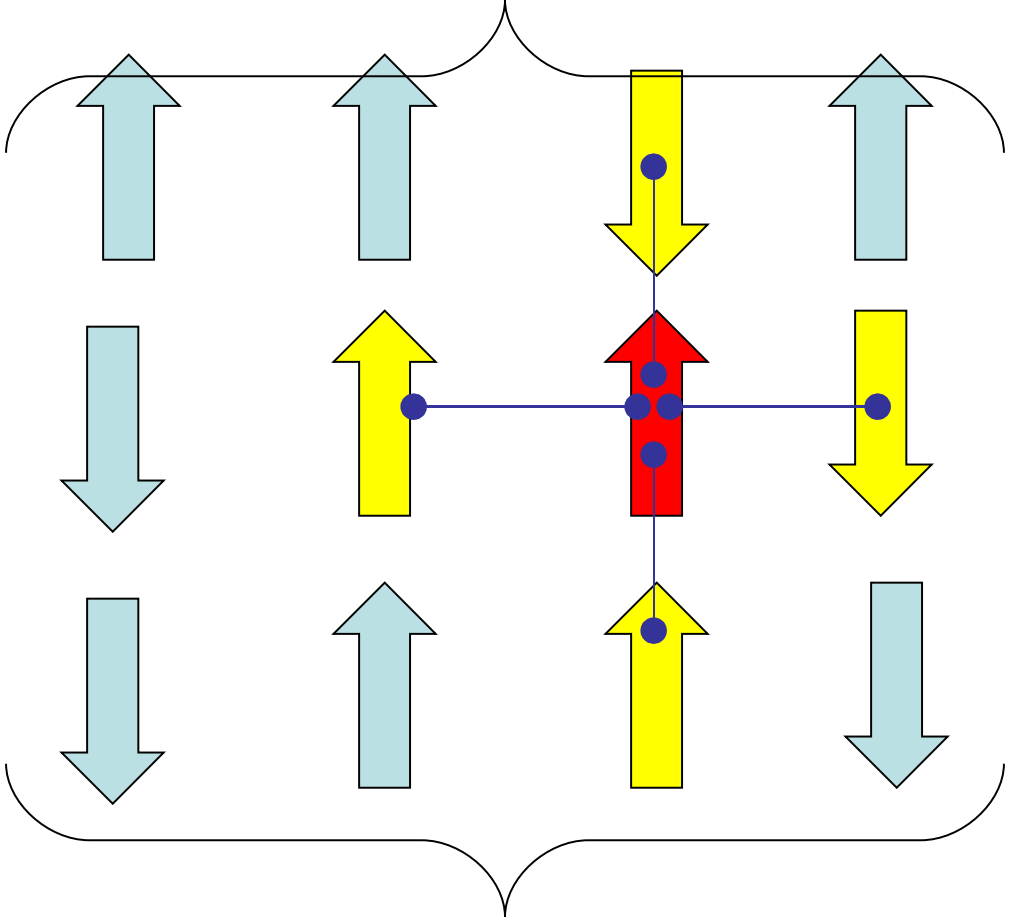
$$S_i = \pm 1$$

we fix J=1

k=1 → fixing the units for T

h=0 → no external magnetic field

- spontaneous magnetization is possible (M≠0 for h=0)
- first model for understanding ferro- and anti-ferromagnetism for localized spins
- for J>0 --> **ferromagnetic** order
- for J<0 --> **anti-ferromagnetic** order
- no phase transition in 1D
- ferro-paramagnetic phase transition for D>1
- second order phase transition (order-disorder)



encoded in a Matrix M[X][Y] (*DimX* x *DimY*)

$$M[X][Y] = S_i = \pm 1$$

$$S_i = 1 \leftrightarrow \Uparrow$$

$$S_i = -1 \leftrightarrow \Downarrow$$

The Metropolis algorithm:

Single spin-flip algorithm→ one spin with coordinates (X=I, Y=j) is attempted to flip

$$P = P(x \to x') = \begin{cases} \exp[-\beta \Delta E(x,x')] \cdots for \cdots \Delta E(x,x') > ۰ \\ ۱ \cdots for \cdots \Delta E(x,x') \le ۰ \end{cases}$$

$$S_{initial} = M[i][j]$$

$$S_{final} = -M[i][j]$$

$$E^{i,j}_{initial} = -M[i][j] \cdot \{M[i+۱][j] + M[i-۱][j] + M[i][j+۱] + M[i][j-۱]\}$$

$$E^{i,j}_{final} = M[i][j] \cdot \{M[i+۱][j] + M[i-۱][j] + M[i][j+۱] + M[i][j-۱]\}$$

$$\Delta E = E^{i,j}_{final} - E^{i,j}_{initial} = ۲ * M[i][j] * \{M[i+۱][j] + M[i-۱][j] + M[i][j+۱] + M[i][j-۱]\}$$

Spins are randomly selected and flipped with a P probability

after the transient ("heat-up") steps the magnetization (Mag) and total energy (E) is followed in time (MC steps)

$$Mag = \sum_{\substack{x=۱,DimX \\ y=۱,DimY}} M[x][y]$$

$$m = |Mag| / (DimX \cdot DimY)$$

$$E = -\frac{۱}{۲} \sum_{\substack{x=۱,DimX \\ y=۱,DimY}} M[x][y] \cdot \{M[x+۱][y] + M[x-۱][y] + M[x][y-۱] + M[x][y+۱]\}$$

The Metropolis MC algorithm for the problem:

1. Fix a temperature (T)

2. Consider an initial spin configuration ($\{\sigma_i\}$). For example for all $i = 1, N \rightarrow \sigma_i = \pm 1$

3. Calculate the initial value of E and M

4. Consider a new spin configuration by virtually "flipping" one randomly selected spin

5. Calculate the energy E' of the new configuration, and the energy change ΔE due to this spin-flip

6. Calculate the Metropolis P=P(x-->x') probabilities for this change

7. Generate a random number "r" between 0 and 1

8. If $r \leq P$ accept the flip and update the value of the energy to E' and magnetization to M'

If $r > P$ reject the spin flip and take again the initial E and M values in the needed averages

9. Repeat the steps 4 - 8 many times (drive the system to the desired canonical distribution of the states)

10. Repeat the steps 4 -8 by collecting the values of E, E^2, M, M^2, for the needed averages

11. Compute this average for a large number of microstates

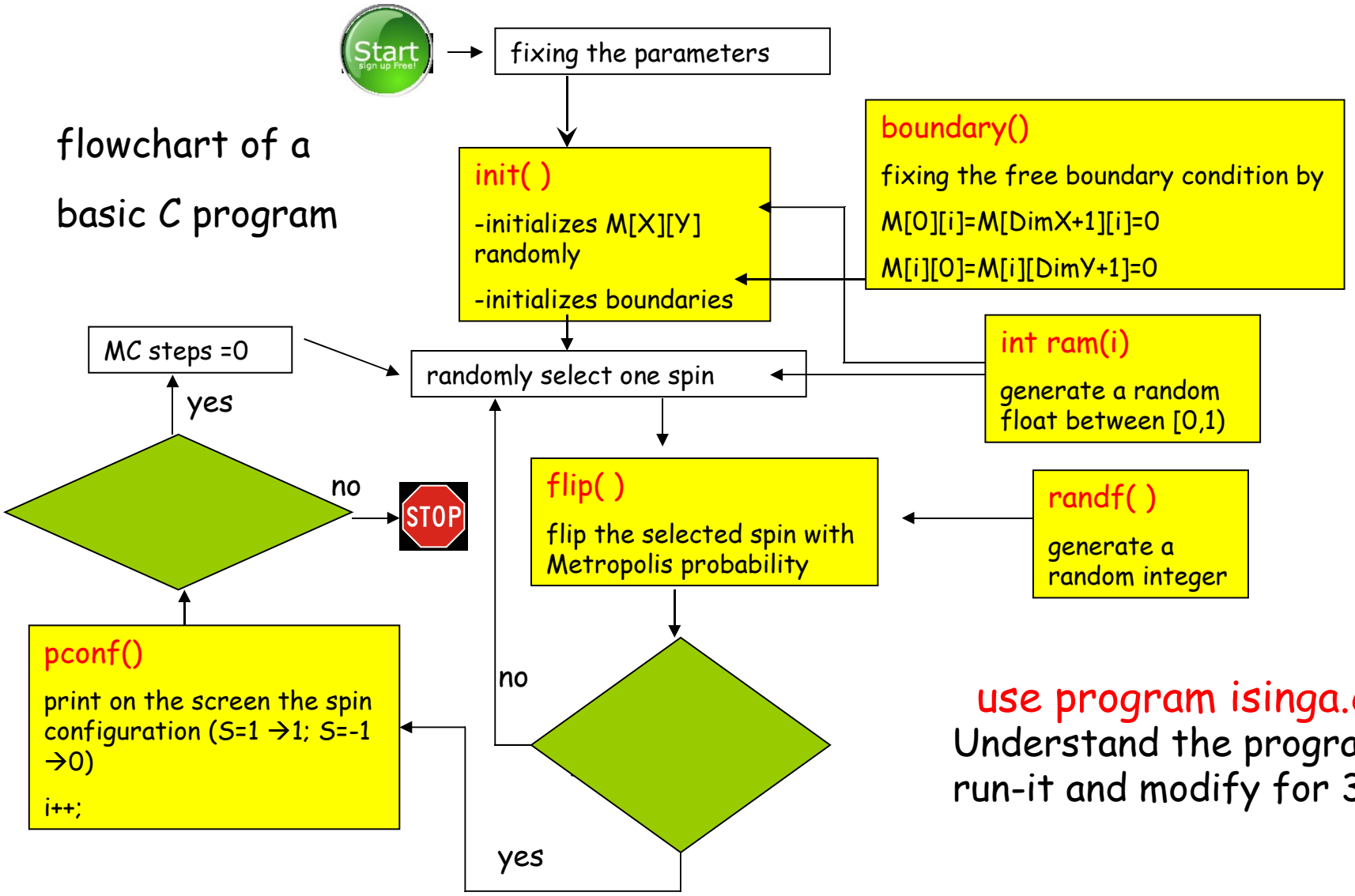
12. Calculate the value of m(T), <E(T)>, $C_v(T)$ and $\chi(T)$ using the given formulas

13. Change the temperature and repeat the algorithm for the new temperatures as well.

14. Construct the desired m(T), <E(T)>, $C_v(T)$, $\chi(T)$ curves

1. Basic program for free boundary condition

Free boundary condition realized by adding a extra rows and columns to the boundaries and considering there spins with S=0 spins! → this gives no contribution to the energies



some Linux commands

- to go in the working directory

cd zneda

- to see what is there

ls -l

- editing the code:

gedit isinga.c or kedit isinga.c

-compiling the code

cc isinga.c –lm –o isinga

-running the code

./isinga

-plotting the results

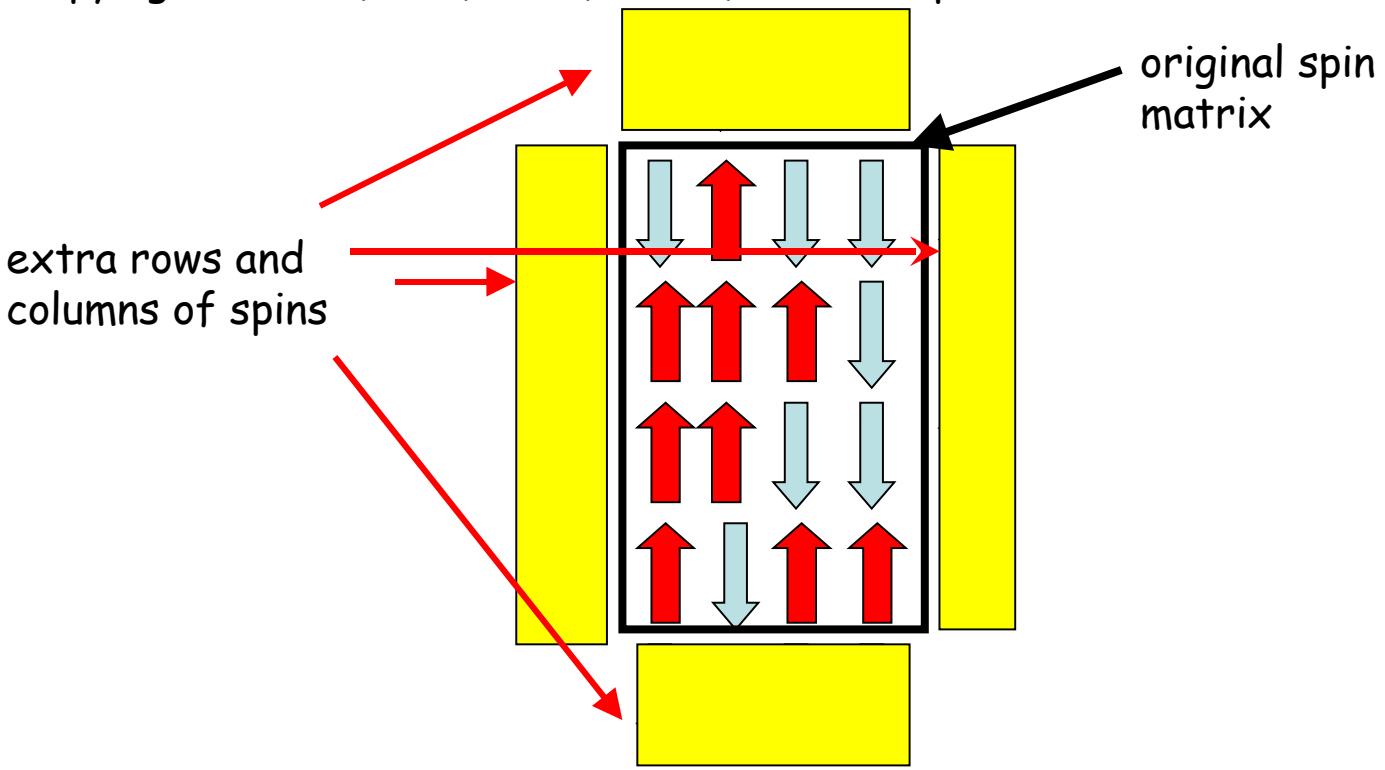
xmgrace magn.dat or gnuplot

>p 'magn.dat' u 1:3 w lp

(this gnuplot instruction plots from file "magn.dat" the third column as a function of the first, with line-connected points)

2. Basic program for periodic boundary condition

The periodic boundary conditions are realized by adding a extra rows and columns to the boundaries, copying the last (first) row (column). See the picture:



Important! → when a spin at the boundaries is flipped, it's image in the added rows (columns) have to be also flipped!

use program isingb.c

Understand the code, run-it and modify it for 3D

3. Calculating averages as a function of temperature. Finite-size effects. Calculation of: $m(T)$ $C_V(T)$ $\chi(T)$

$$m(T) = \left\langle \frac{|M|}{N} \right\rangle = \left\langle \frac{\left|\sum_i \sigma_i\right|}{N} \right\rangle = \frac{\left\langle \left|\sum_i \sigma_i\right| \right\rangle}{N}$$

$$\chi(T) = \frac{1}{NT}\left[< M^2(T) > - < M(T) >^2 \right]$$

$$C_v(T) = \frac{1}{NT^2}\left[< E^2(T) > - < E(T) >^2 \right]$$

-The system is "heated-up" and no averages are calculated for Step_trans MC steps.

-The <m>,<M>, <M^2>, <E> and <E^2> averages are computed for Step_max MC steps

- Results are written out in the "magn.dat" file!

use program isingc.c

- Understand it, run it, modify it for 3D.

- Change the system size, and study finite-size effects!

-Plot the desired $m(T)$ $C_V(T)$ $\chi(T)$ curves

-Study finite-size effects