Simulations in Statistical Physics

Course for MSc physics students

Janos Török

Department of Theoretical Physics

October 13, 2015

Metropolis algorithm

(Metropoli-Rosenbluth-Rosenbluth-Teller-Teller=MR^{2}T^2 algorithm)

- Sequence of configurations using a Markov chain
- Configuration is generated from the previous one
- Transition probability: equilibrium probability
- Detailed balance:

$$P(x)W(x \to x') = P(x')W(x' \to x)$$

- Rewritten:

$$\frac{W(x \to x')}{W(x' \to x)} = \frac{P(x')}{P(x)} = e^{-\beta \Delta E}$$

- Only the ration of transition probabilities are fixed

Metropolis algorithm

(Metropoli-Rosenbluth-Rosenbluth-Teller-Teller=MR^2T^2 algorithm)

$$\frac{W(x \to x')}{W(x' \to x)} = \frac{P(x')}{P(x)} = e^{-\beta \Delta E}$$

- Metropolis:

$$W(x \to x') = \begin{cases} e^{-\beta \Delta E} & \text{if} \, \Delta E > 0 \\ 1 & \text{otherwise} \end{cases}$$

- Symmetric:

$$W(x \to x') = \frac{e^{-\beta \Delta E}}{1 + e^{-\beta \Delta E}}$$

Metropolis algorithm

Recipes:

- Choose an elementary step $x \to x'$
- Calculate ΔE
- Calculate $W(x \to x')$
- Generate random number $r \in [0, 1]$
- If $r < W(x \to x')$ then new state is x'; otherwise it remains x
- Increase time
- Measure what you want
- Restart

Movie1 Movie2

Metropolis algorithm, proposal probability

Transition probability:

$$W(x \to x') = g(x \to x')A(x \to x')$$

- $g(x \to x')$: proposal probability
  - Generally uniform
  - If different interactions are present then it must be incorporated
- $A(x \to x')$: acceptance probability
  - Metropolis
  - Symmetric

Metropolis, *proof*

State flow
Let $E > E'$:

- $x \to x'$

$$P(x)g(x \to x')A(x \to x') = P(x)$$

- $x' \to x$

$$P(x')g(x' \to x)A(x' \to x) = P(x')e^{-\beta \Delta E}$$

- In equilibrium they are equal:

$$\frac{P(x)}{P(x')} = e^{\beta \Delta E}$$

- What we wanted.

Do we need optimization?

- Correlation lenth ξ
- Characteristic time $\tau_{\rm char}$
- Dynamical exponent z

$$\tau_{\rm char} \propto \xi^{z}$$

- For 2d Ising model $z \simeq 2.17$
- Simulation time:

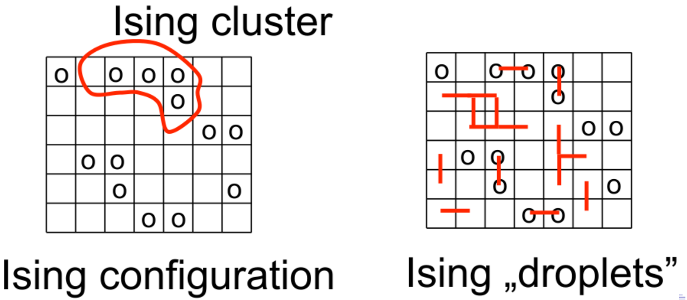
$$t_{\rm CPU} \sim L^{d+z}$$

We need more effective algorithms!

Movie1 Movie2

Cluster algorithm

- Flip more spins together. How?
- The solution – based on an old relationship between the percolation and the Potts model – is that we consider the spin configuration as a correlated site percolation problem
- Ising cluster: a percolating cluster of parallel spins
- Ising droplets: a percolating subset of an Ising cluster
  $p_B = 1 - \exp(-2\beta J)$



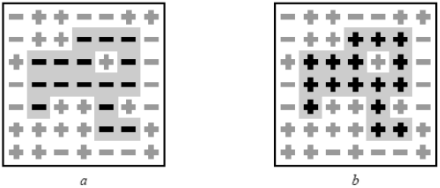
Swendsen-Wang algorithm

- Take an Ising configuration
- With probability $p_B = 1 - \exp(-2\beta J)$ make connection between *parallel* spins
- Identify the droplets by Hoshen-Kopelman algorithm
- Flip each droplet with probability: $1/2$ $(h = 0)$
- Repeat it over

Wolff algorithm

1. Add a random spin to a list of active spins
2. Take a spin from the active list
3. Add each parallel neighboring (not yet visited) spin with probability $p_B = 1 - \exp(-2\beta J)$ to the list of active spins
4. If list of active spins is not empty go to 2.
5. Flip all active spins

A Wolff droplet (gray) before flipping



The new configuration The droplet contour is still shown, though the bonds are eliminated after flipping

Wolff algorithm *proof*

- Detailed balance:

$$P^{eq}(x)W(x \to x') = P^{eq}(x')W(x' \to x)$$

- Metropolis:

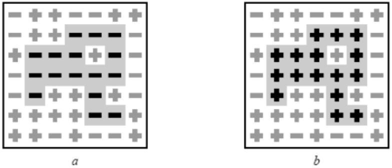
$$W(x \to x') = \min\left\{1, \frac{P^{eq}(x)}{P^{eq}(x')}\right\}$$

- Split W into acceptance A and proposal g probability

$$A(x \to x') = \min\left\{1, \frac{P^{eq}(x)g(x' \to x)}{P^{eq}(x')g(x \to x')}\right\}$$

Wolff algorithm *proof*

A Wolff droplet (gray) before flipping



The new configuration The droplet contour is still shown, though the bonds are eliminated after flipping

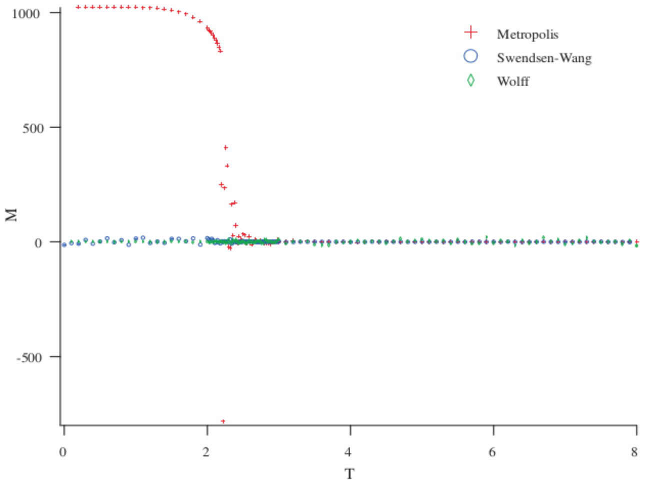
movie

- On the boundary: n_{same} spins parallel and n_{diff} antiparallel.

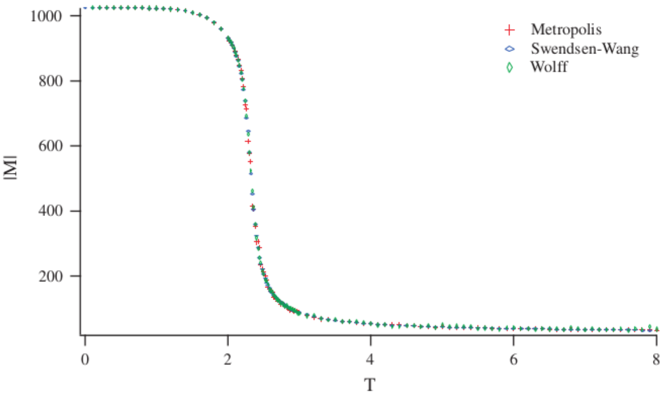
$$
\begin{aligned}
A(x \to x') &= \min\left\{1, \frac{e^{\beta J(n_{\text{diff}}-n_{\text{same}})}}{e^{\beta J(n_{\text{same}}-n_{\text{diff}})}} \frac{(1-p_B)^{n_{\text{diff}}}}{(1-p_B)^{n_{\text{same}}}}\right\} \\
&= \min\left\{1, \frac{e^{-2\beta J n_{\text{same}}}}{e^{-2\beta J n_{\text{diff}}}} \frac{(1-p_B)^{n_{\text{diff}}}}{(1-p_B)^{n_{\text{same}}}}\right\}
\end{aligned}
$$

- It gives: $p_B = 1 - \exp(-2\beta J)$.

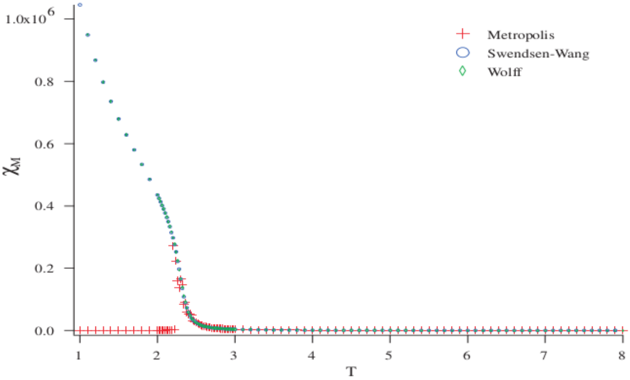
Comparison magnetization



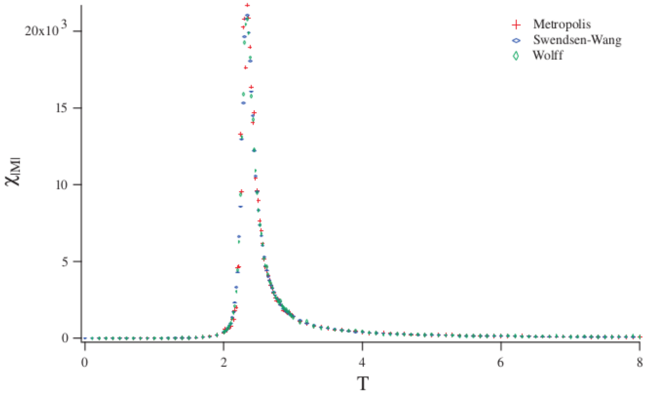
Comparison magnetization



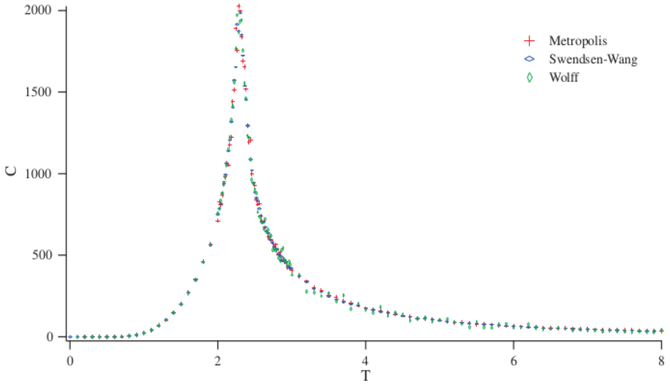
Comparison magnetization



Comparison magnetization



Comparison magnetization



Other ensembles

Microcanonical ensemble

- Daemon with bag with tolerance (both directions)
  - Pick a move, and calculate energy change
  - If energy change does not fit into bag reject it
  - Otherwise add energy change to bag
- In case of conservation the dynamic exponent z is larger!

Other ensembles

Conserved order parameter: Kawasaki dynamics

- Elementary step:
  - Exchange up-down spin pairs (can be anywhere) simultaneously
  - Apply Metropolis to net energy change!
  - Diffusive dynamics is more physical: pick neighboring spins
- In case of conservation the dynamic exponent z is larger!

Calculation of the entropy, free energy, etc.

- Equilibrium statistical physics: From F we can calculate everything
- In simulations F and S cannot be measured directly
- $F = E - TS$ so one of them is enough (E and T are known)
- Solution:
  Calculate the specific heat!

$$C = k_B T^2 \langle (\Delta E)^2 \rangle$$

- The energy fluctuations are measurable
- Since

$$C = T \frac{\partial S}{\partial T}$$

  We have

$$S(T) = S(T_0) + \int_{T_0}^{T} \frac{C(T')}{T'} dT'$$

Calculation of the entropy, free energy, etc.

- In many cases derivate of the entropy is needed so $S(T_0)$ is not important in

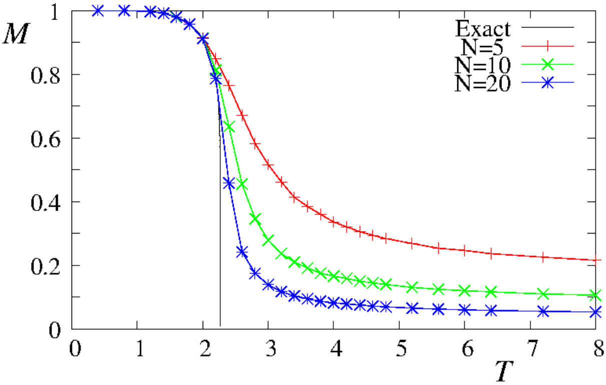
$$S(T) = S(T_0) + \int_{T_0}^{T} \frac{C(T')}{T'} dT'$$

- From third law of thermodynamics: $S(T = 0) = 0$.

Finite size effects

Magnetization 2d lattice Ising model

- Determine critical temperature
- Determine critical exponents
- System size dependence???



Finite size scaling

- Correlation length

$$\xi \propto |T - T_c|^{-\nu}$$

- Cannot be infinite!
- There will be a critical point for the finite system
- If L is finite ξ cannot be larger than L

$$L \propto |T(L) - T_c|^{-\nu}$$

- The position and the width of the transition

$$|T(L) - T_c| \propto L^{-1/\nu}$$

- 3 parameters to fit ν, T_c, and a constant

Finite size scaling

- Binder Cumulant method (find something which does not scale with L)
- Find something which scales with ν
  - The standard deviation of the order parameter:

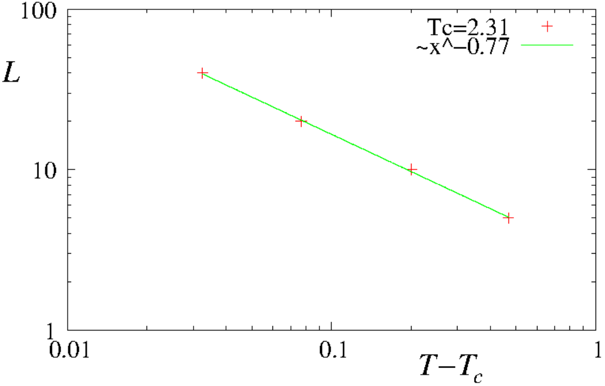
$$\sigma(L) \propto L^{-1/\nu}$$

  - Two steps, both with two parameter fits:

$$\begin{aligned} \sigma(L) &\propto L^{-1/\nu} \\ |T(L) - T_c| &\propto L^{-1/\nu} \end{aligned}$$

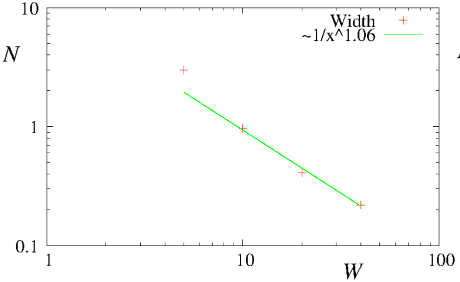
Three parameter fit: Ising model

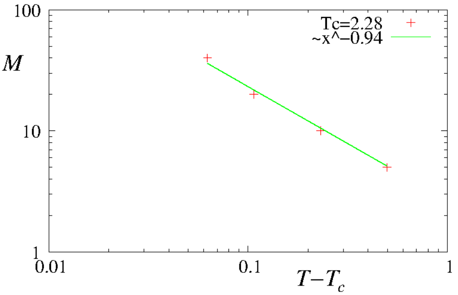
- Theory: $\nu = 1$, $T_c \simeq 2.27$



Finite size scaling: Ising model

- Theory: $\nu = 1$, $T_c \simeq 2.27$





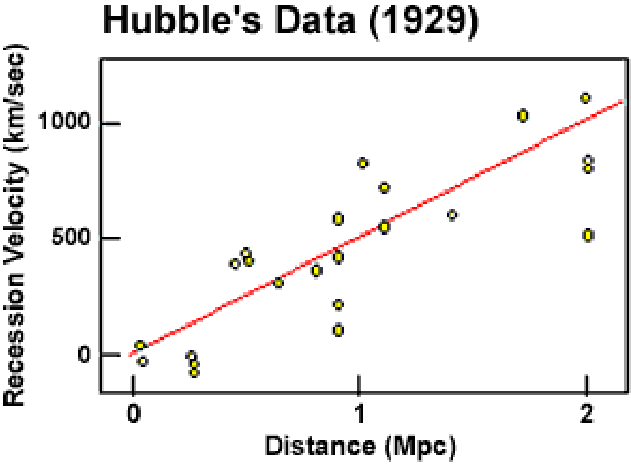
Fitting

Linear regression

$$
\begin{aligned}
y &= \alpha + \beta x \\
\hat{\beta} &= \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} = \frac{\overline{xy} - \bar{x}\bar{y}}{\overline{x^2} - \bar{x}^2} \\
\hat{\alpha} &= \bar{y} - \hat{\beta}\bar{x} \\
\rho &= \frac{\overline{xy}}{\sqrt{\bar{x}\bar{y}}} \qquad (1)
\end{aligned}
$$

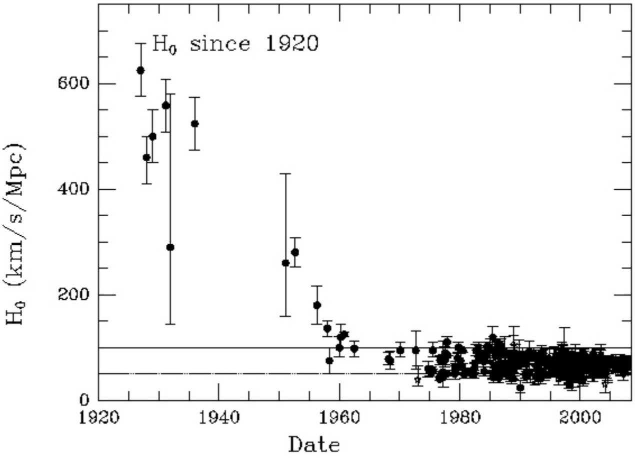
Fitting

Houbble original fit:

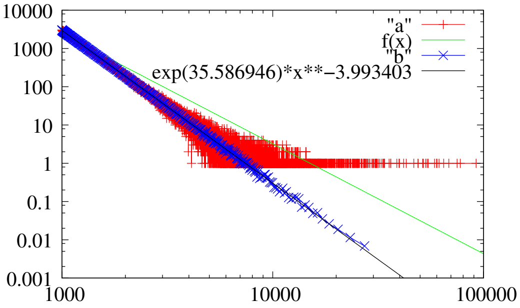


Fitting

Houbble change in time:



Binning and fitting



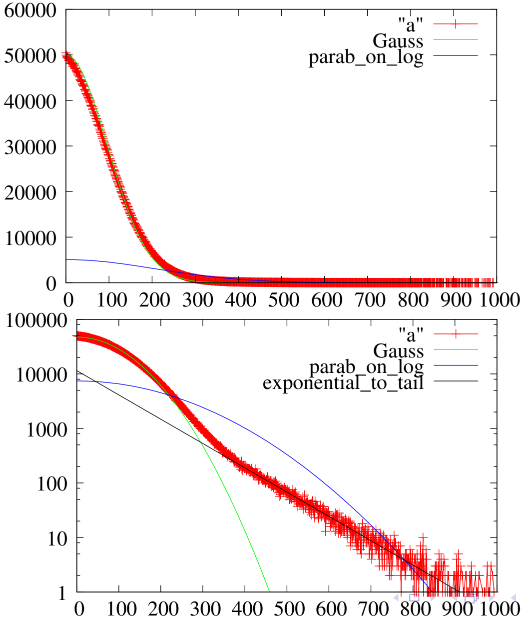
- Linear binning (Δc):

$$c_i = x_{\mathrm{min}} + i\Delta c$$

- Logarithmic binning (Δc):

$$c_i = c_0 \exp(i\Delta c)$$

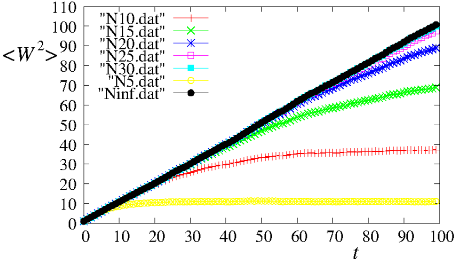
Fitting



"a"
Gauss
parab_on_log
60000
50000
40000
30000
20000
10000
0
0 100 200 300 400 500 600 700 800 900 1000
"a"
Gauss
parab_on_log
exponential_to_tail
100000
10000
1000
100
10
1
0 100 200 300 400 500 600 700 800 900 1000

Diffusion

- On normal lattice exactly sovable
- Otherwise e.g. Monte Carlo kinetics. E.g. 1D
  - With probability 1/2 $\rightarrow$ go right
  - With probability 1/2 $\rightarrow$ go left
  - Be careful with boundary conditions



- Can easily be biased
- Can be simulated on spurious lattices, e.g. Parcolation clusters

Diffusion

- Solution for diffusion on finite lattice:
- Count steps in both directions
- The net move is $W = n_+ - n_0$
- Use ensemble average
- Plot $\langle W^2 \rangle$ vs. t