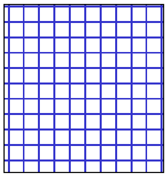
The Ising Model

Today we will switch topics and discuss one of the most studied models in statistical physics the **Ising Model**

- Some applications:
  - Magnetism (the original application)
  - Liquid-gas transition
  - Binary alloys (can be generalized to multiple components)
- Onsager solved the 2D square lattice (1D is easy!)
- Used to develop *renormalization group theory* of phase transitions in 1970's.
- Critical slowing down and "cluster methods".

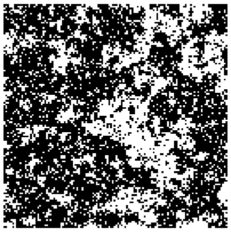
Figures from Landau and Binder (LB), MC Simulations in Statistical Physics, 2000.

The Model



- Consider a lattice with L^2 sites and their connectivity (e.g. a square lattice).
- Each lattice site has a single spin variable: $s_i = \pm 1$.
- With magnetic field **h**, the energy is:

$$H = -\sum_{(i,j)} J_{ij} s_i s_j - \sum_{i=1}^{N} h_i s_i \qquad and \qquad Z = \sum e^{-\beta H}$$



- J is the nearest neighbors (i,j) coupling:
  - J > 0 ferromagnetic.
  - J < 0 antiferromagnetic.
- Picture of spins at the critical temperature T_c. (Note that connected (percolated) clusters.)

Mapping liquid-gas to Ising

- For ***liquid-gas*** transition let n(r) be the density at lattice site r and have two values n(r)=(0,1).

$$E = \sum_{(i,j)} v_{ij} n_i n_j + \mu \sum_i n_i$$

- Let's map this into the Ising model spin variables:

$$s = 2n - 1 \quad \text{or} \quad n = \frac{1}{2}(s+1)$$

$$H = \frac{v}{4}\sum_{(i,j)} s_i s_j + \frac{(v+\mu)}{2}\sum_i s_i + c$$

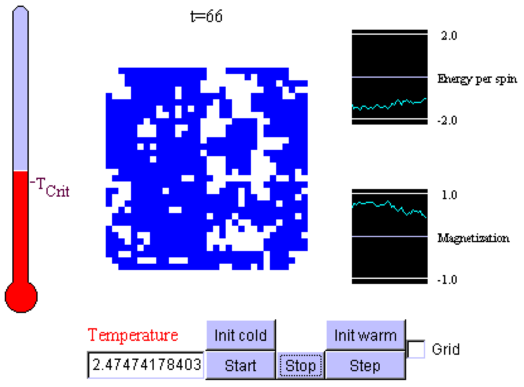
$$J = -v/4$$

$$h = -(v+\mu)/2$$

$$M = \frac{1}{N}\sum_i s_i \qquad \langle \mathrm{n} \rangle = \frac{1}{N}\sum_i n_i = \frac{1}{2}(M+1)$$

JAVA Ising applet

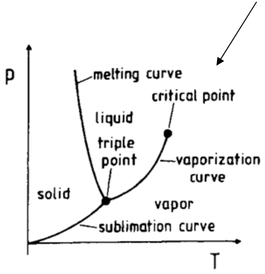
- Check under Course software for the Ising simulator.
- Dynamically runs using heat bath algorithm.

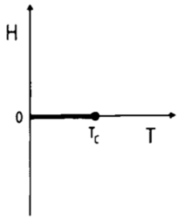


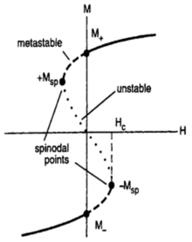
Temperature 2.47474178403

Phase Diagram

- **High-T phase:** spins are random (uncorrelated).
- **T > T_c phase near T_c:** spins are random but correlated: magnetic short-range (local) order.
- **Low-T (T~0) phase:** spins are aligned (fully correlated).
- A ***first-order transition*** (where there is a discontinuous jump in **M**) occurs as H passes through zero for $T<T_c$.
- Similar to **LJ phase diagram**. Magnetic field=pressure.







Critical point

- Concepts and understanding are universal.
  Apply to all phase transitions of similar type.
- Order parameter is *average* magnetization: $<s(r)>=m(r)$
- Look at correlation function: $\chi(r-r')=<s(r)s(r')>-<s(r)><s(r')>$.
- Magnetic susceptibility is: $dm(r)/dh(r')|_{h\to 0} = \beta\chi(r-r')$
- In ordered phase, spin is correlated over long distance.
- At critical point, fluctuations of all scales.

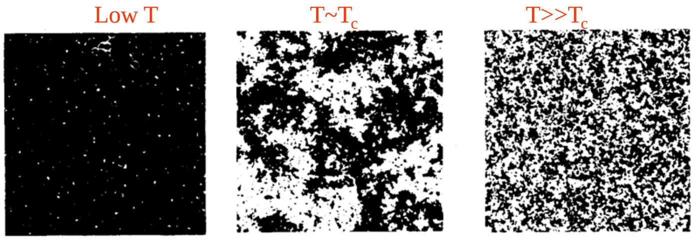
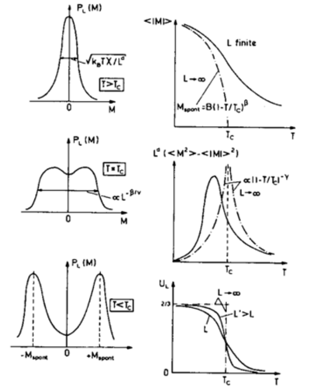

Low T
$T \sim T_c$
$T >> T_c$

Fig. 4.1 Typical spin configurations for the two-dimensional Ising square lattice: (left) $T \ll T_c$; (center) $T \sim T_c$; (right) $T \gg T_c$

Magnetization probability

- How does magnetization vary across transition?
- And with the system size?
- In ordered phase, broken symmetry and barrier to flipping.



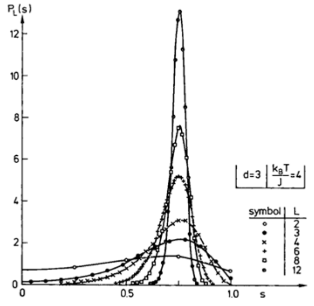


Figure 2. Probability distribution $P_L(s)$ of the magnetization s per spin of $L\times L\times L$ subsystems of a simple cubic Ising lattice with $N = 24^3$ spins and periodic boundary conditions for zero magnetic field and temperature $k_BT/J = 4.0$ (note that the critical temperature occurs at about $k_BT_c/J \approx 4.51$[26].

Figure 3. Schematic variation of the probability distribution $P_L(m)$ to magnetization m in a finite system of linear dimension L from $T > T_c$ to (left part) and the associated temperature variation of the average order pa $< |m| >$, "susceptibility" $k_BT\chi' = L^d(< m^2 > - < |m| >^2)$ and reduced order cumulant $U_L = 1- < m^4 > / \left[3 < m^2 >^2\right]$ (right part).

- If we quench too fast we will end in a two phase region.
- The larger the system the sharper the phase transition.

Phase Diagram: T vs. M

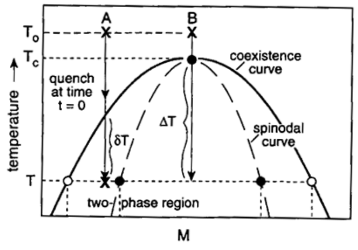
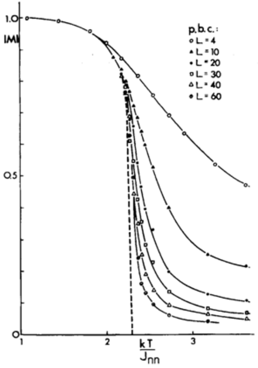


Fig. 2.11 Schematic phase coexistence diagram showing the 'spinodal' line. Paths (A) and (B) represent quenches into the nucleation regime and the spinodal decomposition regime, respectively.

|M| vs. 1/βJ for varying L



Magnetization Scaling depends on T:

$M\propto (T_c-T)^\beta$ for $T< T_c$

$\beta = 0.125$ for D=2.

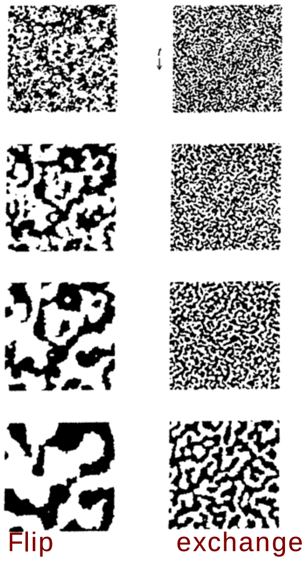
$\beta = 0.325$ for D=3.

Spinoidal decomposition

Suppose spin flips only locally.

- Model for phase separation such as a binary "alloy" (oil and vinegar).

- Dynamics depends on whether the spin is conserved
  - Spin flip (left)
  - Spin exchange (right). conserves particle number
- Transition appears through a coarsening of the separation.
- Becomes slower and slower as the transition proceeds.
  - Critical Slowing down.



$T=0.6T_c$

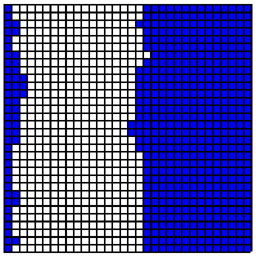
Surfaces/Boundary Conditions

- By quenching quickly we may catch a "trapped" surface.
- Topological excitation.
- You can see steps, etc.

- Can use *twisted boundary conditions* to study a liquid-gas surface without worrying about it disappearing.
- Just put -J along one plane (side). Antiferromagnetic interaction along one plane.

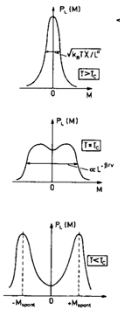
$$H = -\sum_{(i,j)} J_{ij} s_i s_j$$

$$J_{ij} = \begin{cases} J & i \neq 0 \\ -J & i = 0 \end{cases}$$

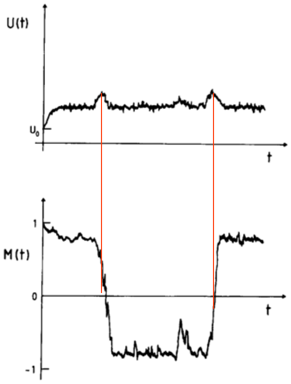


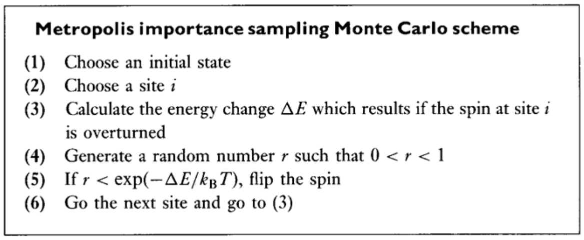
Critical slowing down

- Near the transition dynamics gets very slow if you use any local update method.
- The larger the system the less likely it is that the system can flip over.



Monte Carlo of a zero-field Ising Lattice
U vs. time and M vs. time.



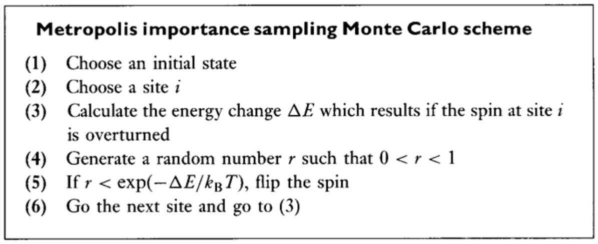


Metropolis importance sampling Monte Carlo scheme

(1) Choose an initial state
(2) Choose a site i
(3) Calculate the energy change ΔE which results if the spin at site i is overturned
(4) Generate a random number r such that $0 < r < 1$
(5) If $r < \exp(-\Delta E/k_B T)$, flip the spin
(6) Go the next site and go to (3)

Local versus cluster algorithms

- Simplest Metropolis:
  - Lots of tricks to make it run faster.
  - Tabulate exp(-E/kT)
  - Do several flips each cycle by packing bits into a word
  - But critical slowing down near Tc.
  - At low T accepted flips are rare--can speed up by sampling ***acceptance time***.
  - At high T all flips are accepted--ergodic problem.



Metropolis importance sampling Monte Carlo scheme

(1) Choose an initial state
(2) Choose a site i
(3) Calculate the energy change ΔE which results if the spin at site i is overturned
(4) Generate a random number r such that $0 < r < 1$
(5) If $r < \exp(-\Delta E/k_B T)$, flip the spin
(6) Go the next site and go to (3)

Glauber and Kawasaki dynamics

- **Heat bath or Glauber:**
  - Pick a spin and flip with probability
  - Will have lower flipping rate but no high T problem.
- **N-fold way:**
  - Look at all the sites, choose the site "i" according to:
  - The normalization determines how time advances.
  - Discuss this later with kinetic MC
- **Kawasaki dynamics**
  - Exchange spins and accept or reject
  - Spin is constant as in spinoidal decomposition.
- ALL THESE ARE LOCAL hence suffer from slowdown.

$$p^i = \frac{\pi_i}{\pi_i + \pi_j} = \frac{1}{1+e^{\text{Š}\beta\Delta E}}$$

$$T^i = \frac{\pi_i}{\sum_j \pi_j}$$

Swendsen-Wang cluster algorithm

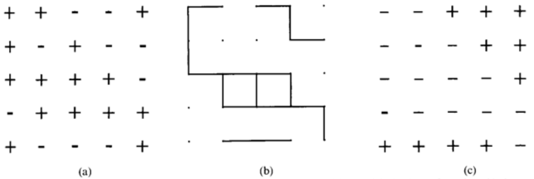
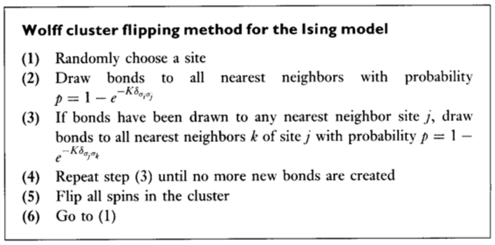
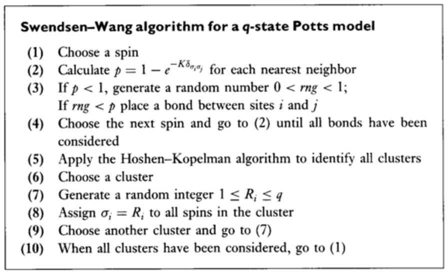


Fig. 5.1 Schematic view of the Swendsen–Wang algorithm for an Ising model: (a) original spin configuration; (b) clusters formed: (c) 'decorated' clusters.



Wolff cluster flipping method for the Ising model

(1) Randomly choose a site
(2) Draw bonds to all nearest neighbors with probability $p = 1 - e^{-K\delta_{\sigma_i\sigma_j}}$
(3) If bonds have been drawn to any nearest neighbor site j, draw bonds to all nearest neighbors k of site j with probability $p = 1 - e^{-K\delta_{\sigma_j\sigma_k}}$
(4) Repeat step (3) until no more new bonds are created
(5) Flip all spins in the cluster
(6) Go to (1)



Swendsen–Wang algorithm for a *q*-state Potts model

(1) Choose a spin
(2) Calculate $p = 1 - e^{-K\delta_{\sigma_i\sigma_j}}$ for each nearest neighbor
(3) If $p < 1$, generate a random number $0 < rng < 1$;
If $rng < p$ place a bond between sites i and j
(4) Choose the next spin and go to (2) until all bonds have been considered
(5) Apply the Hoshen–Kopelman algorithm to identify all clusters
(6) Choose a cluster
(7) Generate a random integer $1 \leq R_i \leq q$
(8) Assign $\sigma_i = R_i$ to all spins in the cluster
(9) Choose another cluster and go to (7)
(10) When all clusters have been considered, go to (1)

No critical slowing down at the critical point.

Non-local algorithm. Prove detailed balance! **See FS 399-408**