Monte Carlo Simulation of Spins

Aiichiro Nakano

Collaboratory for Advanced Computing & Simulations
Department of Computer Science
Department of Physics & Astronomy
Department of Chemical Engineering & Materials Science
Department of Quantitative & Computational Biology
University of Southern California

Email: anakano@usc.edu

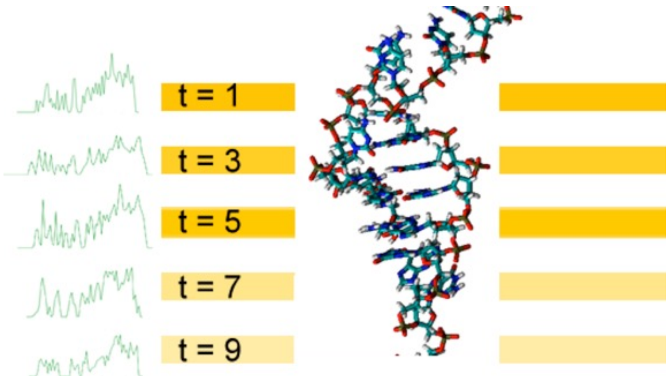
Markov Chains for Complex Dynamics

- **Hidden Markov model — Viterbi algorithm (*cf.* dynamic programming)**
- **Perron-Frobenius eigenvalue cluster analysis**

DNA Sequencing via Quantum Mechanics and Machine Learning

Int'l J. Comput. Sci. **4**, 352 ('10)

Henry Yuen[1], Fuyuki Shimojo[1,2], Kevin J. Zhang[1,3], Ken-ichi Nomura[1], Rajiv K. Kalia[1], Aiichiro Nakano[1*], Priya Vashishta[1]



cf. Henry's historical breakthrough

A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition

LAWRENCE R. RABINER, FELLOW, IEEE — *Proc. IEEE*, **77**, 257 ('89)

Implementing the Viterbi Algorithm

Fundamentals and real-time issues for processor designers

HUI-LING LOU

IEEE Signal Processing Mag., **12(5)**, 42 ('95)

ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Linear Algebra and its Applications 398 (2005) 161–184

LINEAR ALGEBRA AND ITS APPLICATIONS

www.elsevier.com/locate/laa

Robust Perron cluster analysis in conformation dynamics ☆

Peter Deuflhard, Marcus Weber *

USC Viterbi School of Engineering

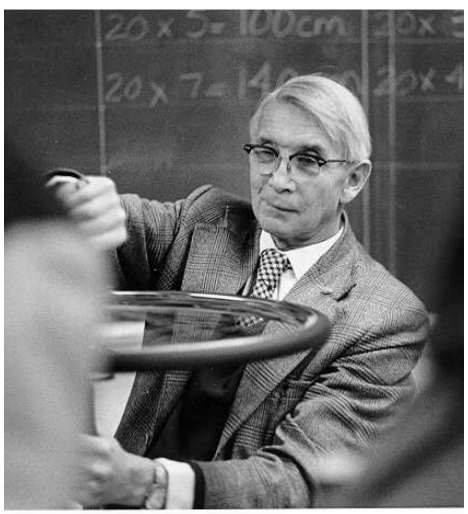
USC

Ising Model

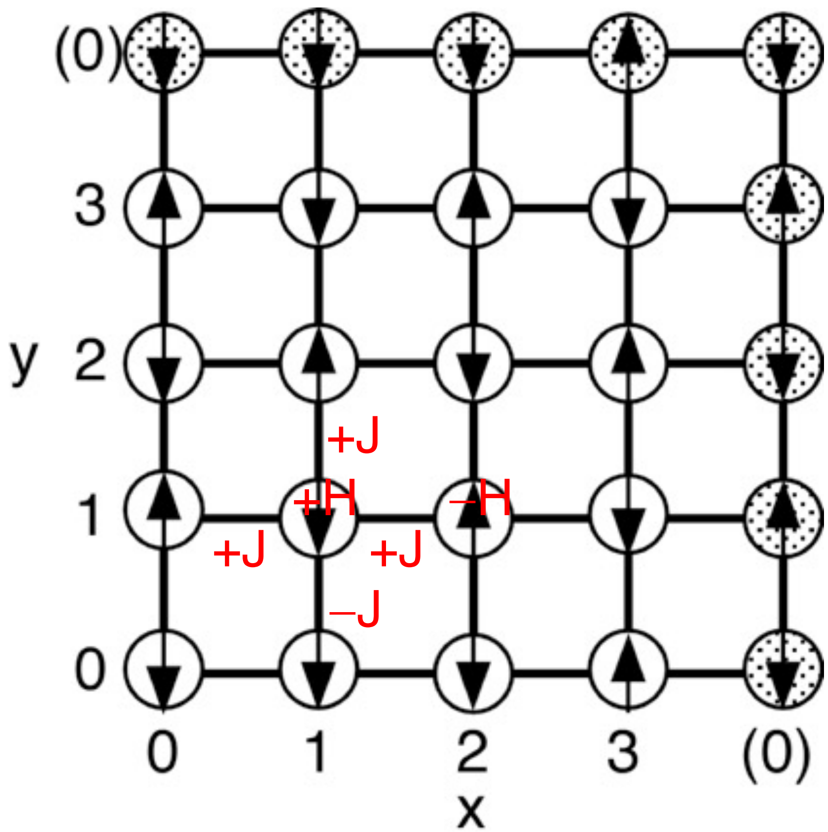
- **Ising model: A collection of spins on a lattice,** $\{s_k \mid s_k = \pm 1\}$
- **Potential energy**

$$V(s^N) = -J \sum_{(k,l)} s_k s_l - H \sum_k s_k, \quad s_k = \pm 1$$

where *J* is the exchange coupling, *H* is the magnetic field, & (*k*, *l*) are nearest-neighbor pairs of lattice sites



Ernst Ising



Curie temperature

- **Periodic boundary condition: Wrapping around the lattice**

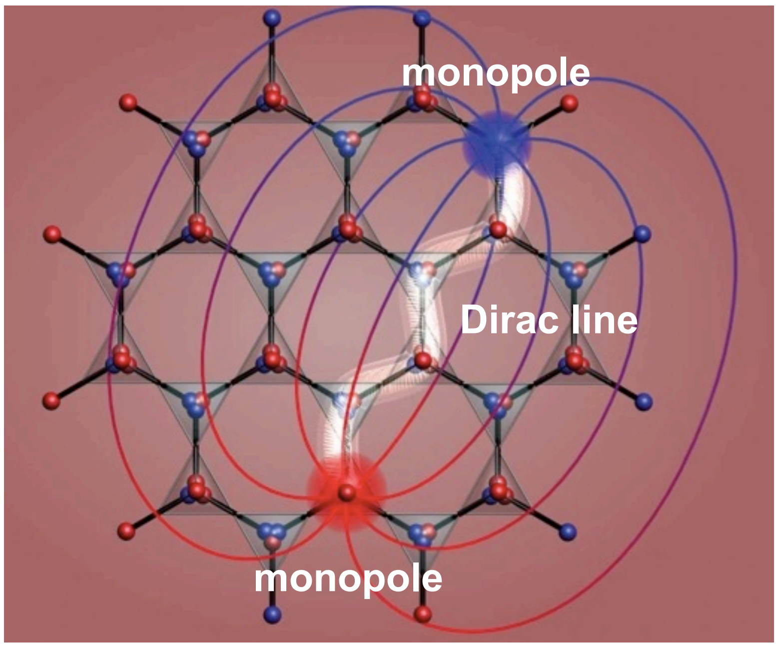
Exotic Magnets

nature Vol 451 | 3 January 2008 | doi:10.1038/nature06433

LETTERS

Magnetic monopoles in spin ice

C. Castelnovo[1], R. Moessner[1,2] & S. L. Sondhi[3]

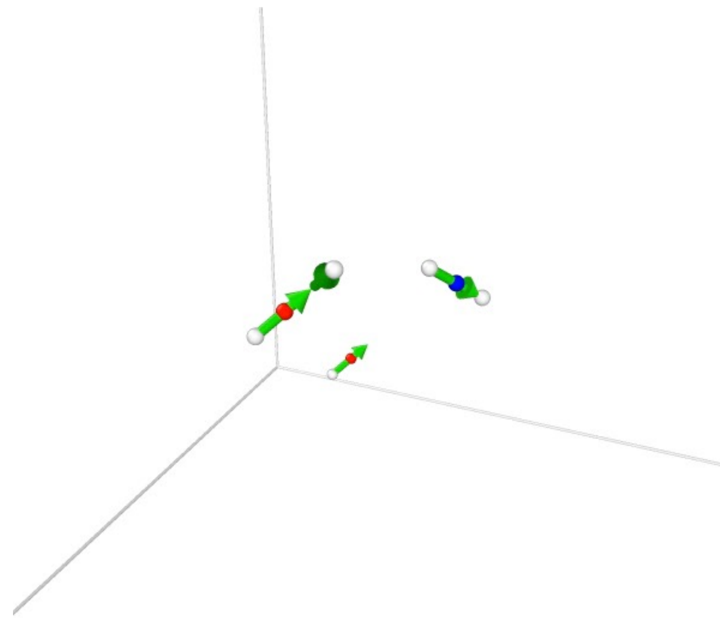


MAGNETISM

Qubit spin ice

Andrew D. King[1]*, Cristiano Nisoli[2]*, Edward D. Dahl[1,3], Gabriel Poulin-Lamarre[1], Alejandro Lopez-Bezanilla[2]

Science **373**, 576–580 (2021)



$$H=\frac{J}{3}\sum_{\langle ij\rangle} S_i S_j + D a^3 \sum_{(ij)} \left[\frac{\hat{e}_i \cdot \hat{e}_j}{\left|\mathbf{r}_{ij}\right|^3} - \frac{3\left(\hat{e}_i \cdot \mathbf{r}_{ij}\right)\left(\hat{e}_j \cdot \mathbf{r}_{ij}\right)}{\left|\mathbf{r}_{ij}\right|^5} \right] S_i S_j$$

Monte Carlo simulation

3D Ising Problem Is NP-Complete

PHYSICAL REVIEW VOLUME 65, NUMBERS 3 AND 4 FEBRUARY 1 AND 15, 1944

Crystal Statistics. I. A Two-Dimensional Model with an Order-Disorder Transition

LARS ONSAGER
Sterling Chemistry Laboratory, Yale University, New Haven, Connecticut

(Received October 4, 1943)

The partition function of a two-dimensional "ferromagnetic" with scalar "spins" (Ising model) is computed rigorously for the case of vanishing field. The eigenwert problem involved in the corresponding computation for a long strip crystal of finite width (n atoms), joined straight to itself around a cylinder, is solved by direct product decomposition; in the special case $n=\infty$ an integral replaces a sum. The choice of different interaction energies $(\pm J, \pm J')$ in the (0 1) and (1 0) directions does not complicate the problem. The two-way infinite crystal has an order-disorder transition at a temperature $T=T_c$ given by the condition

$$\sinh(2J/kT_c)\sinh(2J'/kT_c)=1.$$

The energy is a continuous function of T; but the specific heat becomes infinite as $-\log|T-T_c|$. For strips of finite width, the maximum of the specific heat increases linearly with $\log n$. The order-converting dual transformation invented by Kramers and Wannier effects a simple automorphism of the basis of the quaternion algebra which is natural to the problem in hand. In addition to the thermodynamic properties of the massive crystal, the free energy of a (0 1) boundary between areas of opposite order is computed; on this basis the mean ordered length of a strip crystal is

$$(\exp(2J/kT)\tanh(2J'/kT))^n.$$

Statistical Mechanics, Three-Dimensionality and NP-completeness *

I. Universality of Intractability for the Partition Function of the Ising Model Across Non-Planar Lattices

[Extended Abstract]

Sorin Istrail
Sandia National Laboratories
Applied Mathematics Department, MS 1110
Albuquerque, NM 87185-1110
scistra@cs.sandia.gov

STOC 2000 Portland Oregon USA
1-58113-184-4/00/5

Nobel laureate Richard Feynman wrote in 1972 of the three-dimensional Ising model that "the exact solution for three dimensions has not yet been found."

Other researchers who have tried read like a roll call of famous names in science and mathematics: Onsager, Kac, Feynman, Fisher, Kasteleyn, Temperley, Green, Hurst, and more recently Barahona.

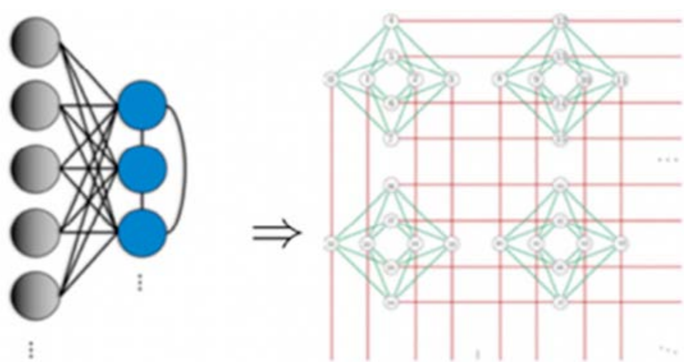
Says Istrail, "What these brilliant mathematicians and physicists failed to do, indeed cannot be done."

Ising Model in Machine Learning

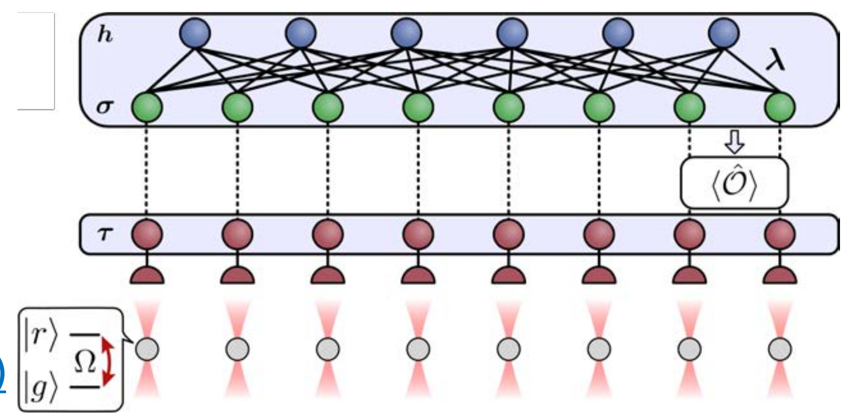
Physically very appealing methods for unsupervised learning are the so-called ***Boltzmann machines*** (BM). A BM is basically an ***inverse Ising model*** where the data samples are seen as samples from a Boltzmann distribution of a pairwise interacting Ising model. The goal is to learn the values of the interactions and magnetic fields so that the likelihood (probability in the Boltzmann measure) of the observed data is large.

Machine Learning and the Physical Sciences
G. Carleo *et al*., *Rev. Mod. Phys*. **91**, 045002 ('19)

Quantum-neural nexus?



Hybrid BM-quantum circuit
G. Torlai *et al*., *Phys. Rev. Lett*. **123**, 230504 ('19)



Quantum-annealing Boltzmann machine
J. Liu *et al*., *Comput. Mater. Sci*. **173**, 109429 ('20)

MC Algorithm for 2D Ising Model

```
initialize the spins, s[i][j] (0 ≤ i,j ≤ L-1)
Sum_A = 0
for step = 1 to maximum_step
  randomly select a grid point, (i,j)
  compute the change in potential energy, dV, with a single spin
  flip, s_i,j → -s_i,j
  if dV < 0 accept the flip, s_i,j ← -s_i,j
  else if random() ≤ exp(-dV/kBT) then      // 0 < random() < 1
    accept the flip, s_i,j ← -s_i,j
  endif
  Sum_A = Sum_A + A(s^N)      // Sample physical quantity A(s^N)
endfor
Average_A = Sum_A/maximum_step
```

Sample

Mean

$$\pi_{mn} = \overbrace{\min(\rho_m/\rho_n, 1)}^{\text{accept/reject}}\ \overbrace{\alpha_{mn}}^{\text{attempt}}$$

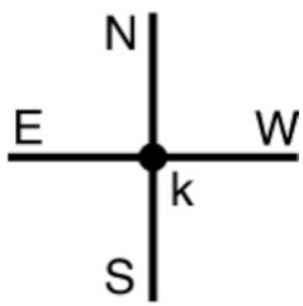
$$\delta V = V(\ldots, s_k', \ldots) - V(\ldots, s_k, \ldots)$$

$$= -J \sum_{l \in n.n.(k)} (s_k' - s_k) s_l - H(s_k' - s_k)$$

$$= -2 s_k' \left(J \sum_{l \in n.n.(k)} s_l + H \right)$$

$$\frac{P(\ldots, s_k', \ldots)}{P(\ldots, s_k, \ldots)} = \frac{e^{-V(\ldots, s_k', \ldots)/k_B T}}{e^{-V(\ldots, s_k, \ldots)/k_B T}} = \exp\left(- \frac{\overbrace{V(\ldots, s_k', \ldots) - V(\ldots, s_k, \ldots)}^{\delta V}}{k_B T} \right)$$

```
~~i = L*(rand()/(double)RAND_MAX);~~
~~j = L*(rand()/(double)RAND_MAX);~~
i = rand()%L;
j = rand()%L;
```



$$V(s^N) = -J \sum_{(k,l)} s_k s_l - H \sum_k s_k, \quad s_k = \pm 1$$

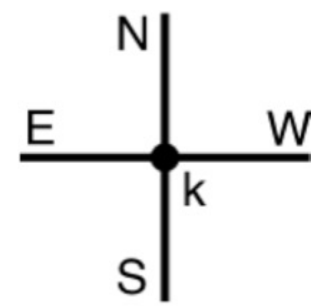
State Transition

Spin flip: $s_k \rightarrow s'_k = -s_k$

- `double exp_dV[2][5]:` $\exp(-\delta V / k_B T) = \exp\left(2\boxed{s'_k}\left(\frac{J}{k_B T}\boxed{\sum_{l \in n.n.(k)} s_l} + \frac{H}{k_B T}\right)\right)$

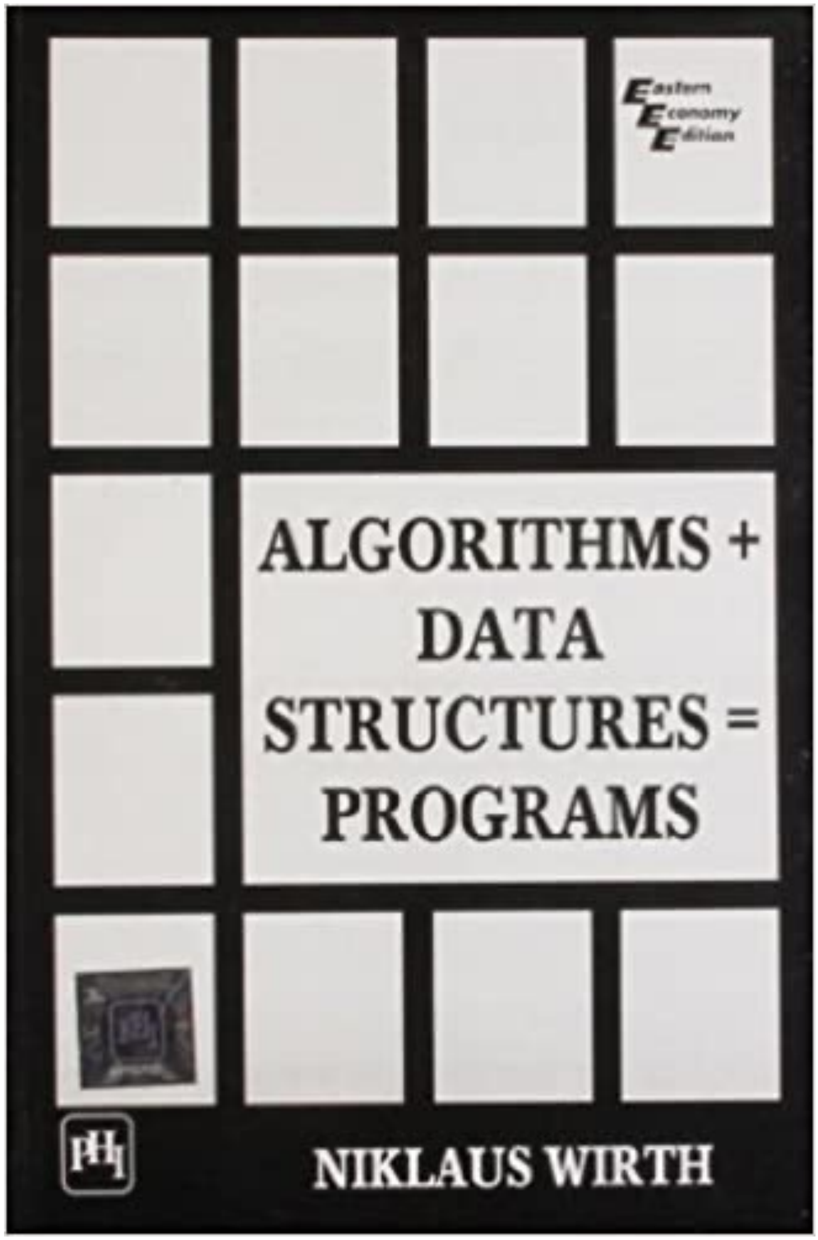
Central spin, s_k'
-1
1

Neighbor spins				Sum
-1	-1	-1	-1	-4
-1	-1	-1	1	-2
-1	-1	1	1	0
-1	1	1	1	2
1	1	1	1	4

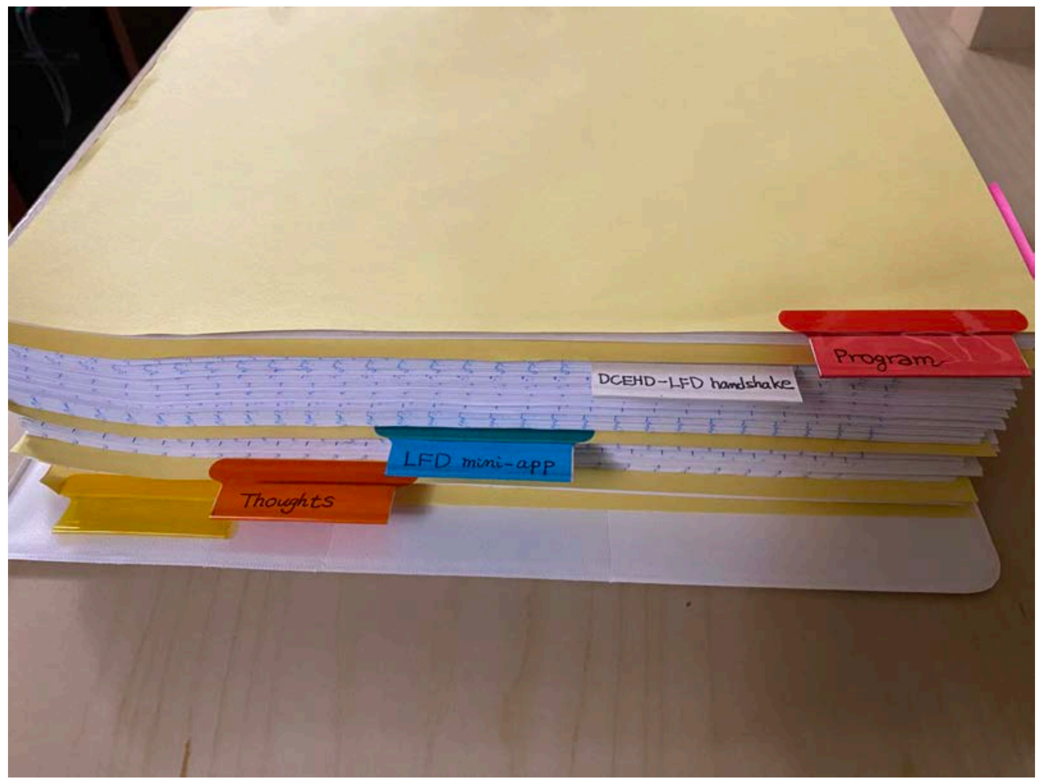


Pre-compute & store transition probability in a table
Note exp() function is compute-intensive

Now, How to Code?



Life of a computational physicist: Concept to algebra to program (= data structure—define & algorithm—pseudocode)



https://github.com/USCCACS/DCMESH

Data Structures

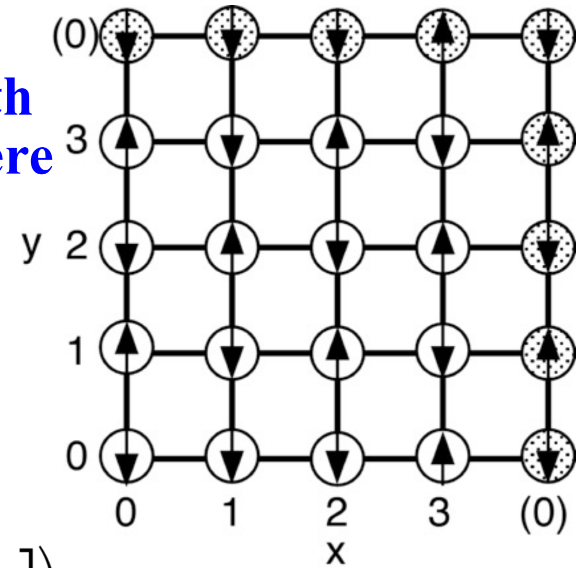
- ```
  #define L 20 //Lattice size
  int s[L][L]; //Spins s[i][j] = ±1
  ```
- **Periodic boundary condition: The west, east, south, north neighbors of site (*i*, *j*) are (*im*, *j*), (*ip*, *j*), (*i*, *jm*), (*i*, *jp*), where**

  $im = (i + L - 1)\ \%\ L$

  $ip = (i + 1)\ \%\ L$

  $jm = (j + L - 1)\ \%\ L$

  $jp = (j + 1)\ \%\ L$



- **Transition probability:** `double exp_dV[2][5]`

$$\exp\_dV[k][l] = \exp\left(-\frac{\delta V}{k_B T}\right) = \exp\left(2s\left[\frac{J}{k_B T}\underbrace{\sum_{s' \in n.n.(s)} s'}_{S} + \frac{H}{k_B T}\right]\right)$$

where

  $k = (1 + s)/2\ (k = 0,1;\ s = -1,1)$

  $l = (4 + S)/2\ (l = 0,1,2,3,4;\ S = \Sigma_{\text{neighbor}}\ s' = -4,-2,0,2,4)$

Spin→array-index transformation

k=(1+s)/2	s
0	−1
1	1

l=(4+S)/2	S
0	−4
1	−2
2	0
3	2
4	4

- **Ising model parameters**

  `double JdivT` $= J/k_B T$

  `double HdivT` $= H/k_B T$
  ```

Physical Quantities

- `double runM;`
  > **Running value of magnetization,** $M = \sum_k s_k$
  > **To update,** $M \mathrel{+}= 2s_k'$

- `double sumM = 0.0, sumM2 = 0.0;` // double type to avoid overflow
  > **To calculate the mean & variance of the magnetization at the end**
  > **After each MC step**

```
sumM += runM;
sumM2 += runM*runM;
```

- `int Sta_step;`
  > **# of MC steps to be performed**
  > **Mean & standard deviation of the magnetization**

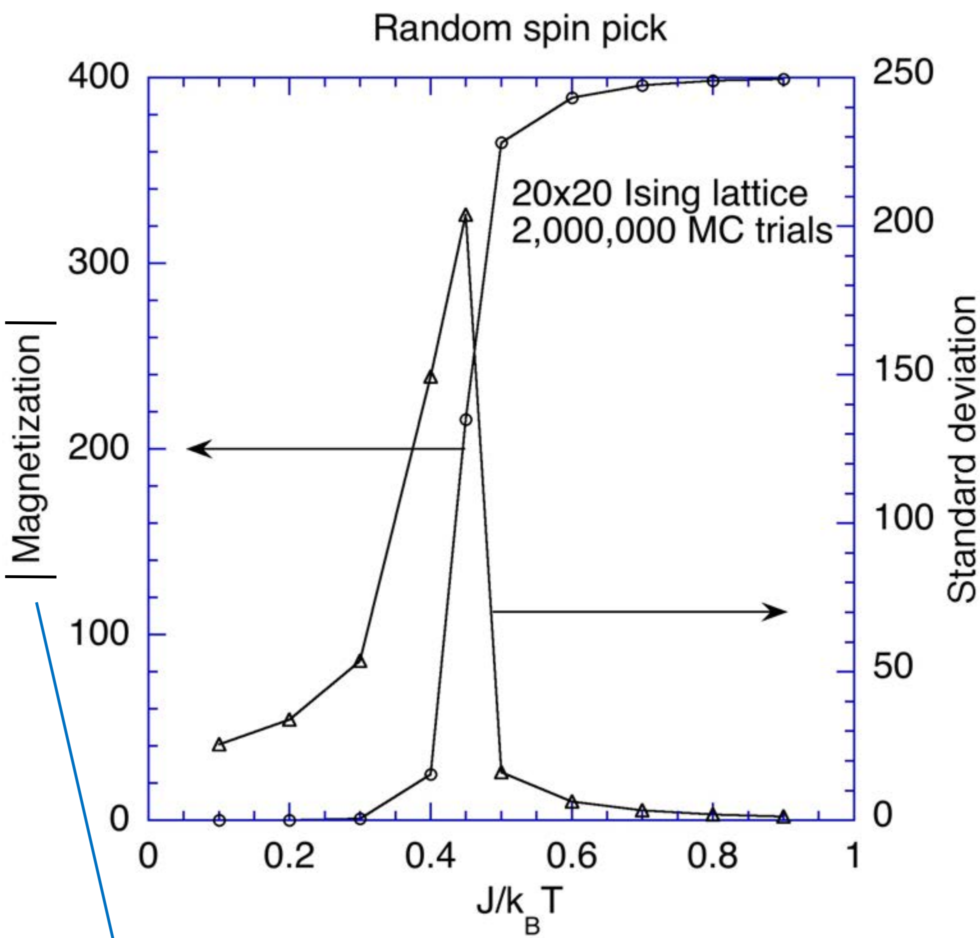
$$avgM = sumM/Sta\_step$$

$$sigM = \sqrt{\frac{sumM2}{Sta\_step} - avgM^2}$$

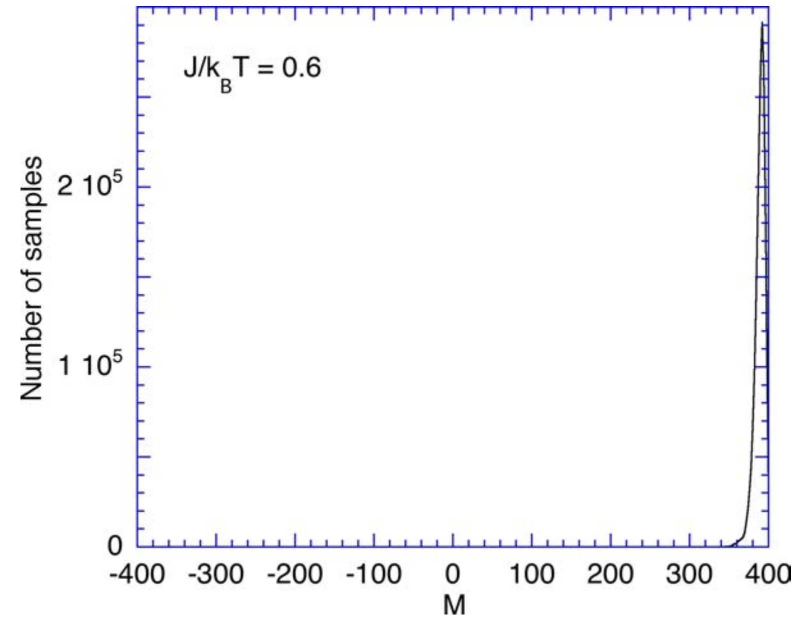
```
double avgM, sigM;
```

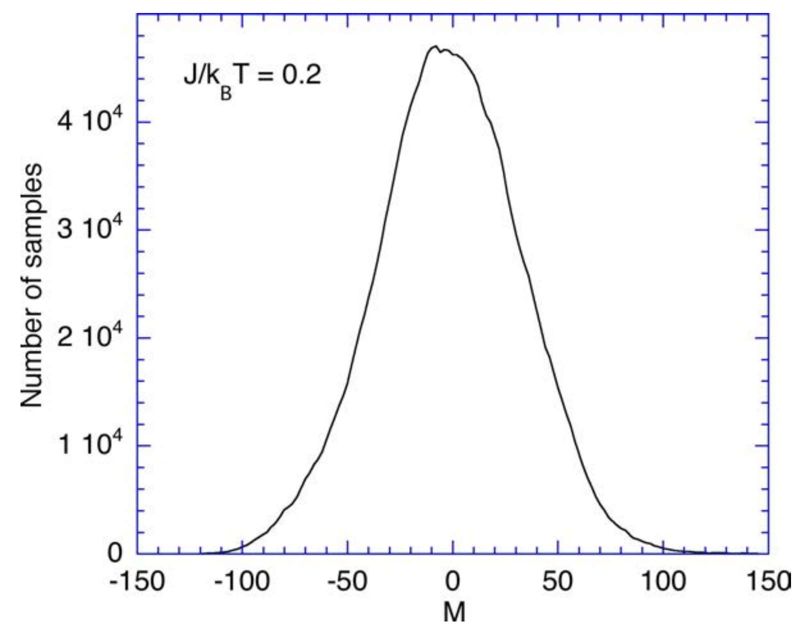
Magnetization & Its Fluctuation

L = 20



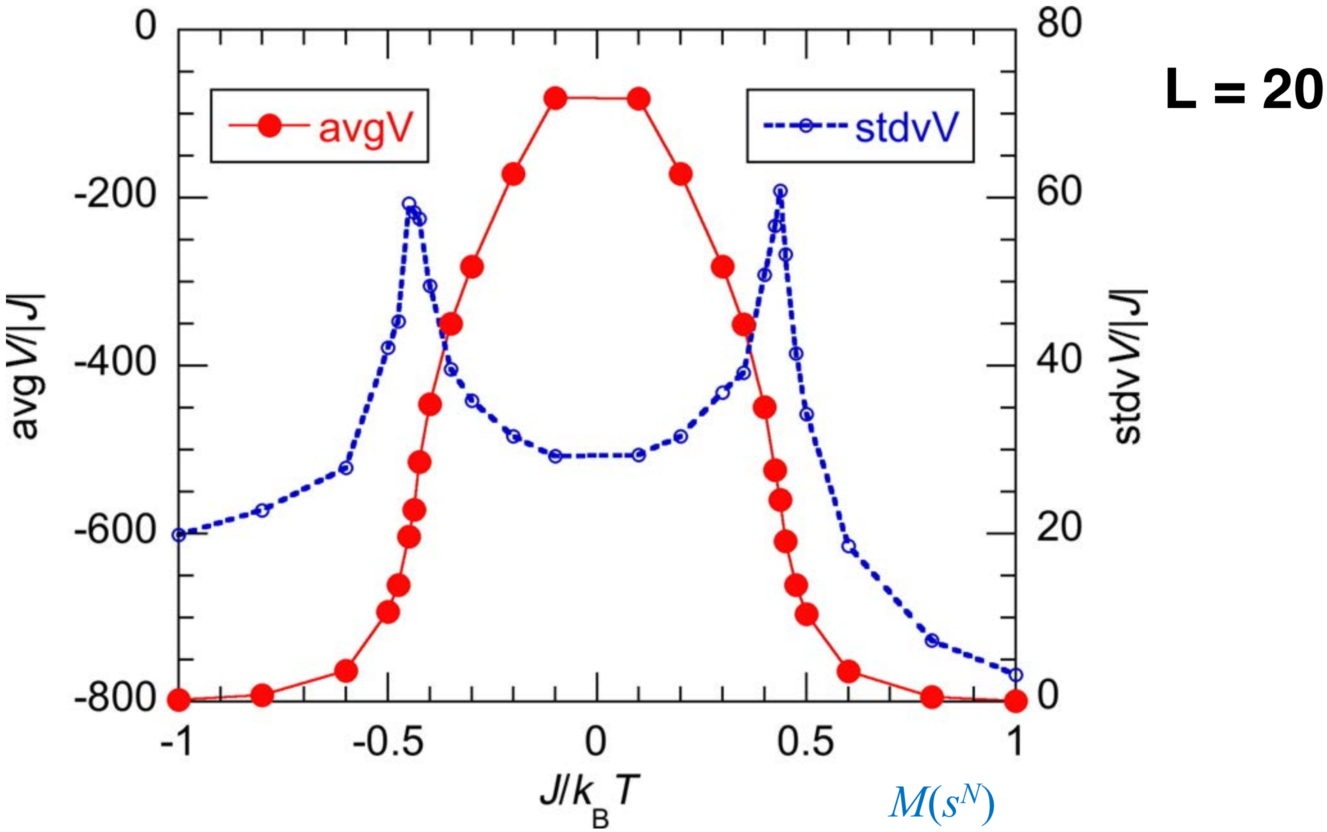
Note: it's an absolute value





Histogram for 2 million MC steps

Energy & Its Fluctuation



Potential energy: $V(s^N) = -J \sum_{(k,l)} s_k s_l - H \overbrace{\sum_k s_k}^{M(s^N)}, \quad s_k = \pm 1$

Rich physics: (1) ferro-to-para-to-antiferromagnetic phase transitions; (2) nucleation growth; (3) fluctuation-dissipation theorem

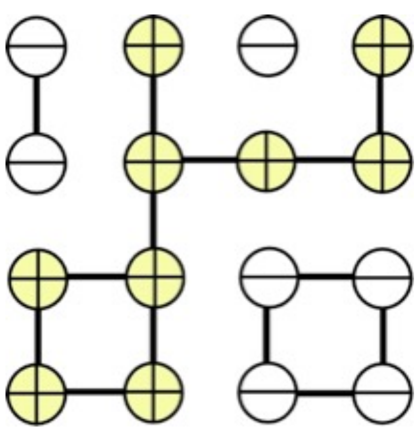
See notes on (1) fluctuation-dissipation theorem, (2) unitary time propagation & (3) linear response

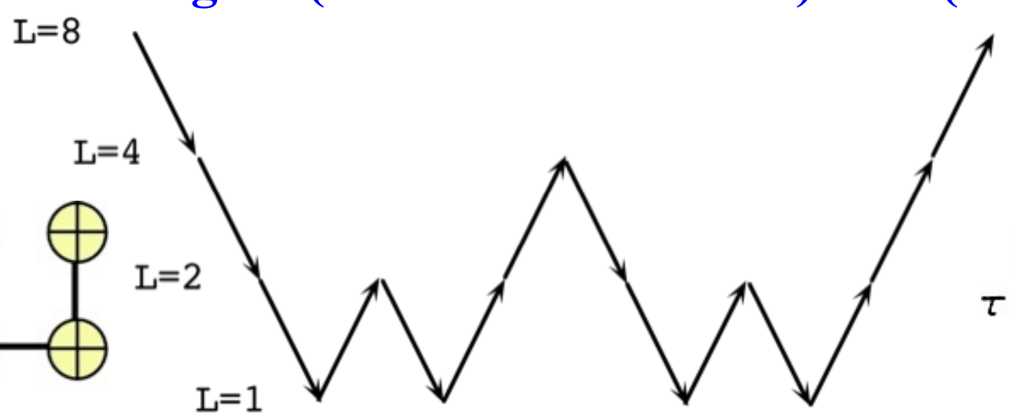
Cluster MC Algorithms

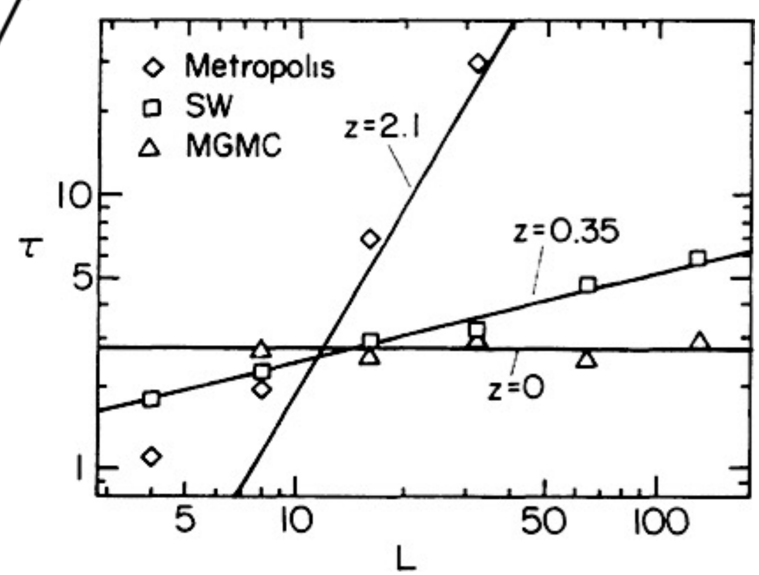
- **Cluster MC: Speed up the convergence of MC simulations by introducing collective motions of the degrees-of-freedom, *e.g.*, flipping a cluster of spins at a time**
- **Correlation time, τ: The number of MC steps before two states become uncorrelated,** $\langle \delta M(t+t_0)\delta M(t_0)\rangle_{t_0} = \langle (\delta M)^2\rangle \exp(-t/\tau)$
- **Dynamic critical exponent, *z*: Near the critical temperature for magnetic-to-nonmagnetic phase transition,** $\tau \sim L^z$ **(*L*: system size)**
  > $z = 2.125$ **for 2D Ising model**

  > $z = 0.35$ **for Swendsen-Wang cluster MC ('87)**

  > $z \sim 0$ **for multigrid (hierarchical cluster) MC (Kandel *et al.*, '88)**







Cluster MC Algorithm—Wolff

Algorithm: Single step of Wolff's cluster flip

1. Choose a seed spin at random from the lattice.
2. Look in turn at each of the neighbors of that spin. If they are pointing in the same direction as the seed spin, add them to the cluster with probability $P_{\mathrm{add}} = 1 - \exp(-2J/k_{\mathrm{B}}T)$.
3. For each spin that was added in step 2, examine each of its neighbors to find the ones pointing in the same direction & add each of them to the cluster with probability P_{add}. Repeat this step as many times as necessary until there are no spins left in the cluster whose neighbors have not been considered for inclusion in the cluster.
4. Flip the cluster

U. Wolff, *Phys. Rev. Lett*. **62**, 361 ('89)

https://aiichironakano.github.io/phys516/Phys516-MCred.pdf

Detailed Balance in Wolff Algorithm

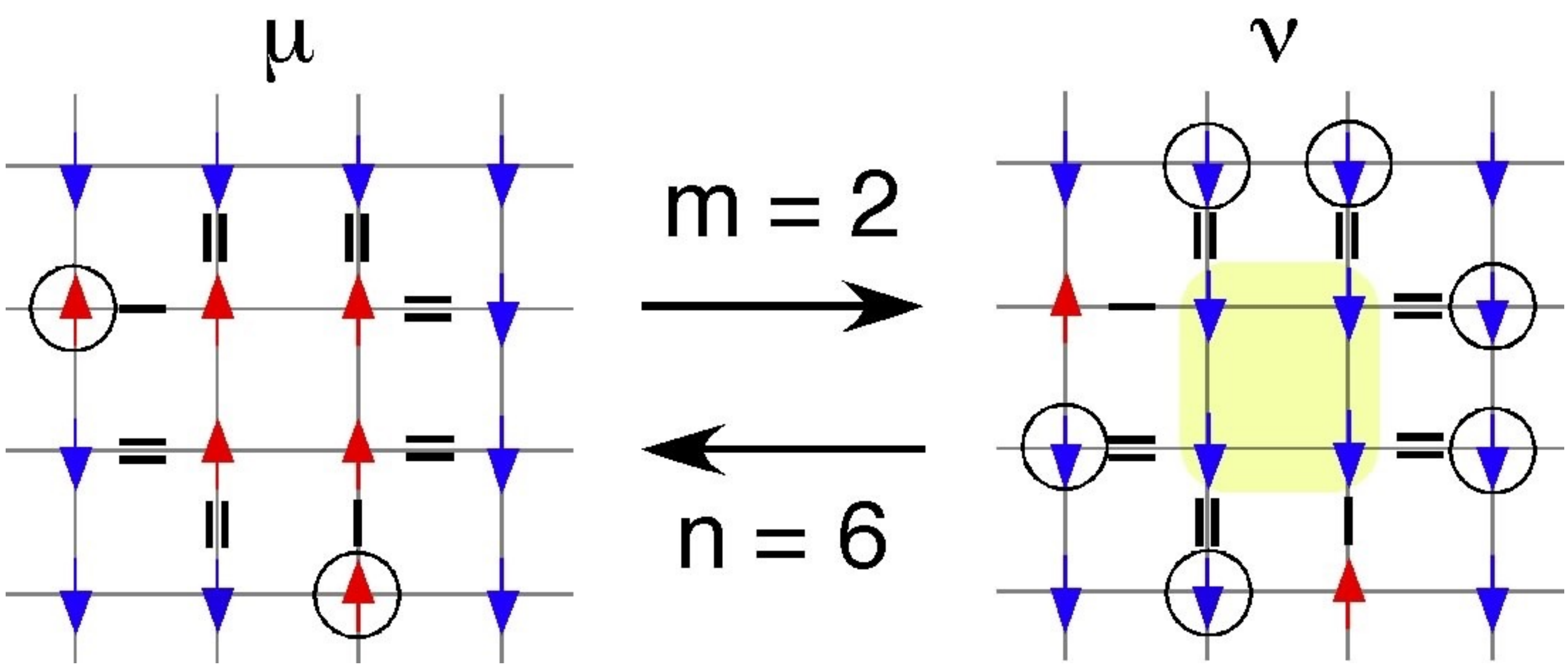
- **Key to detailed balance,** $\pi_{mn}\rho_n = \pi_{nm}\rho_m$ **:** $P_{\text{add}} = 1 - \exp(-2J/k_B T)$

- **Detailed balance for a cluster flip:**
  μ → ν (*m* bonds broken); ν → μ (*n* bonds broken)

$$(1 - P_{\text{add}})^m P(\mu) = (1 - P_{\text{add}})^n P(\nu)$$

or

$$(1 - P_{\text{add}})^{m-n} = P(\nu)/P(\mu)$$



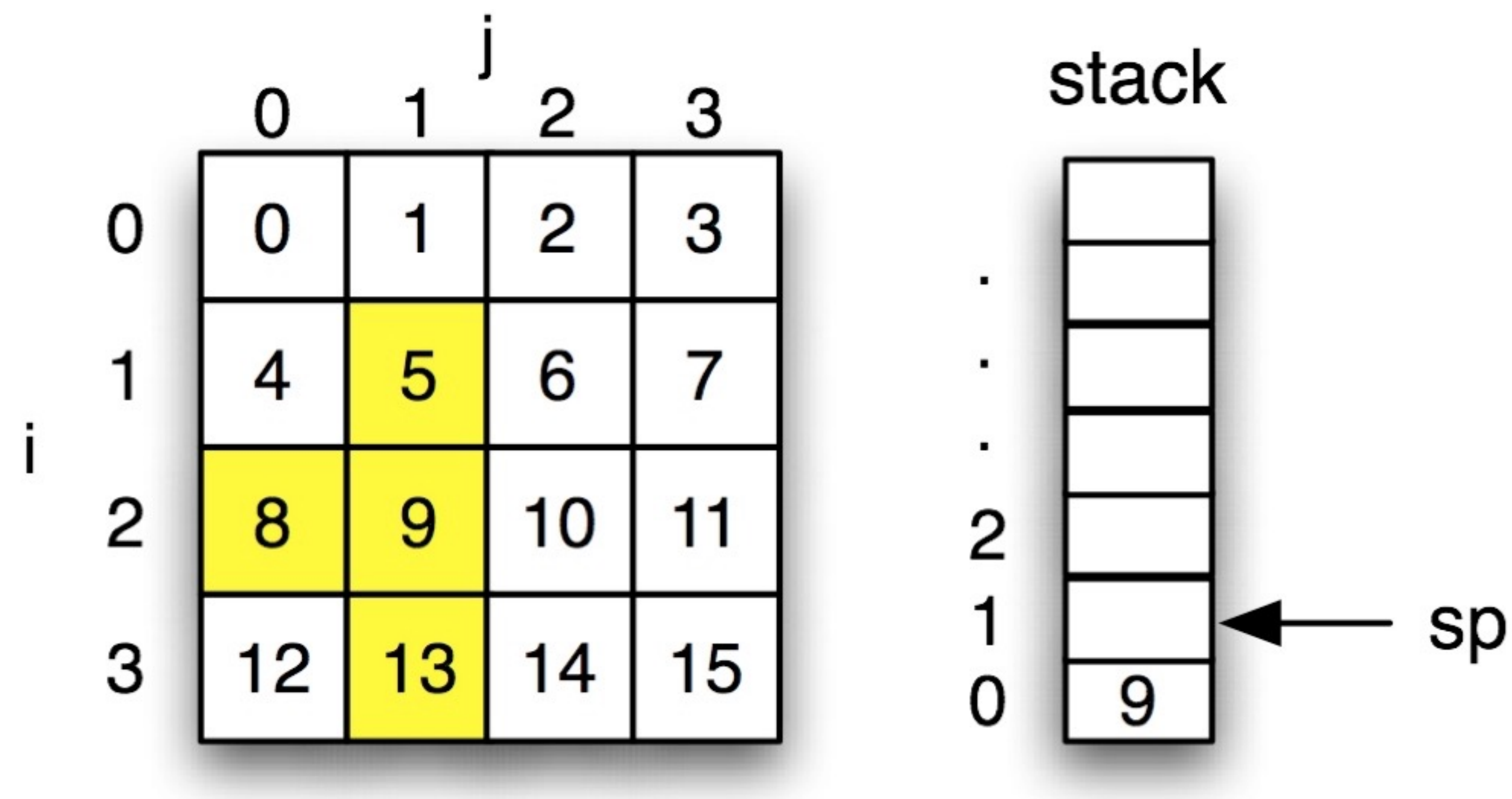
$$(1 - P_{\text{add}})^{m-n} = \exp(-2J[m-n]/k_B T) = \exp(-[E_\nu - E_\mu]/k_B T) = P(\nu)/P(\mu)$$

Wolff Cluster MC Program

Initiate a cluster-flip step using a stack

```
/* Put a random seed spin site onto a stack */
i = rand()%L; j = rand()%L;
stack[0] = i*L + j; // 1D site index put in a stack
sp = 1;  // Stack pointer

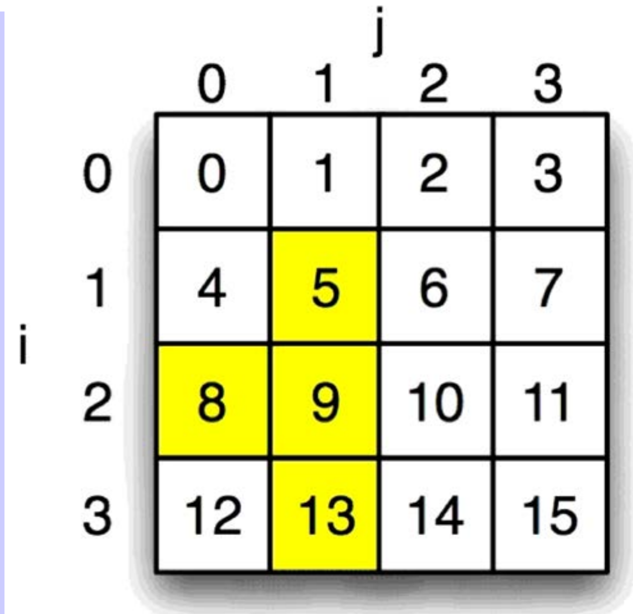
/* Flip the seed and remember the old & new spins */
oldspin = s[i][j]; newspin = -s[i][j];
s[i][j] = newspin;
```

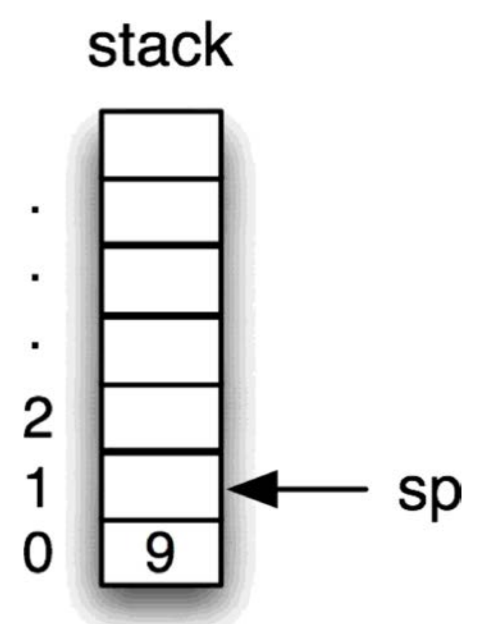


Cluster Flip

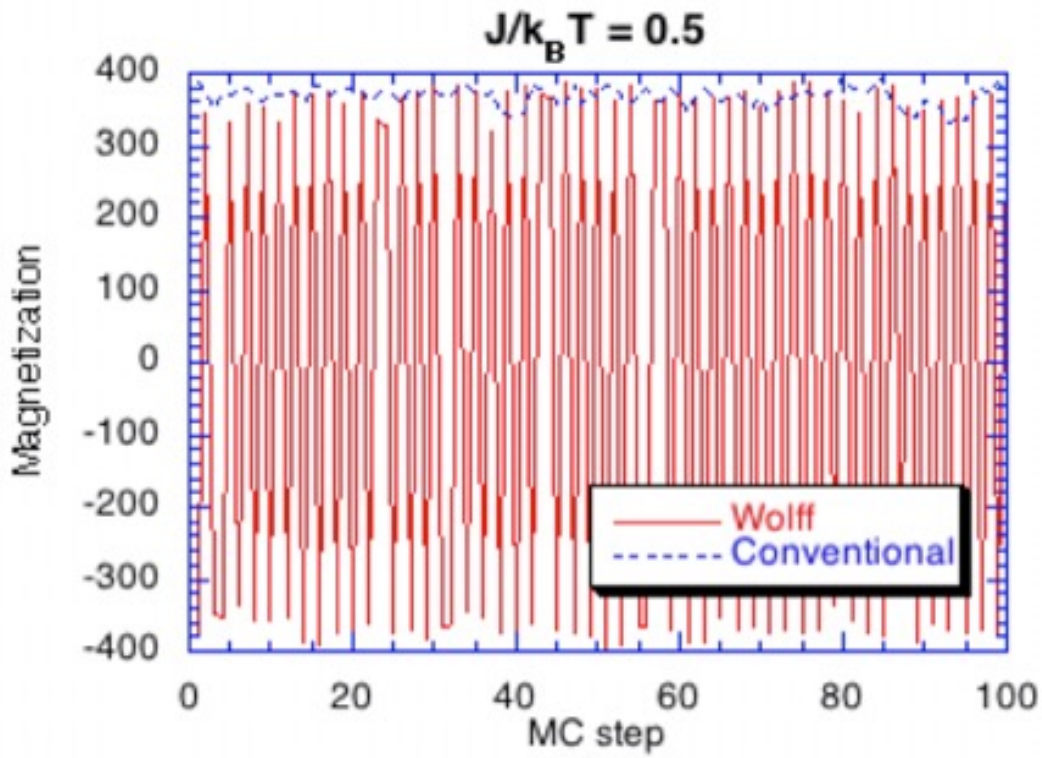
```c
while (sp) {
  /* Pop a site off the stack */
  current = stack[--sp]; i = current/L; j = current%L;

  /* Check the neighbors */
  if ((nn=i+1) >= L) nn -= L; /* South neighbor */
  if (s[nn][j] == oldspin)
    if (rand()/(double)RAND_MAX < padd) {
      stack[sp++] = nn*L + j; s[nn][j] = newspin;
    }

  if ((nn=i-1) < 0) nn += L; /* North neighbor */
  if (s[nn][j] == oldspin)
    if (rand()/(double)RAND_MAX < padd) {
      stack[sp++] = nn*L + j; s[nn][j] = newspin;
    }

  if ((nn=j+1) >= L) nn -= L; /* East neighbor */
  if (s[i][nn] == oldspin)
    if (rand()/(double)RAND_MAX < padd) {
      stack[sp++] = i*L + nn; s[i][nn] = newspin;
    }

  if ((nn=j-1) < 0) nn += L; /* West neighbor */
  if (s[i][nn] == oldspin)
    if (rand()/(double)RAND_MAX < padd) {
      stack[sp++] = i*L + nn; s[i][nn] = newspin;
    }
} /* End while stack is not empty */
```

$$padd = 1 - \exp\left(-2J / k_B T\right)$$

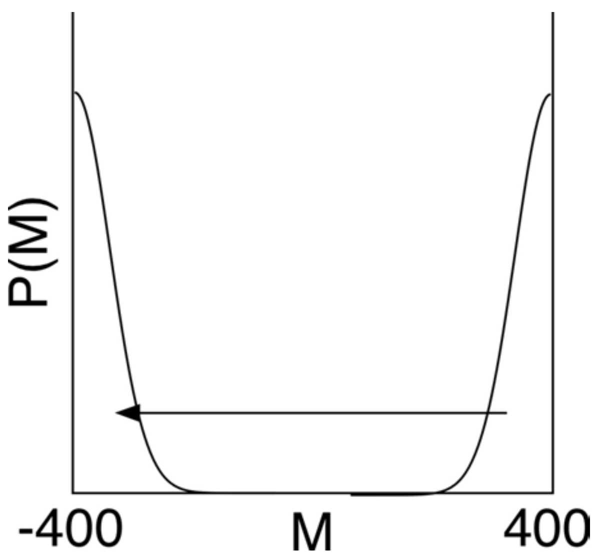




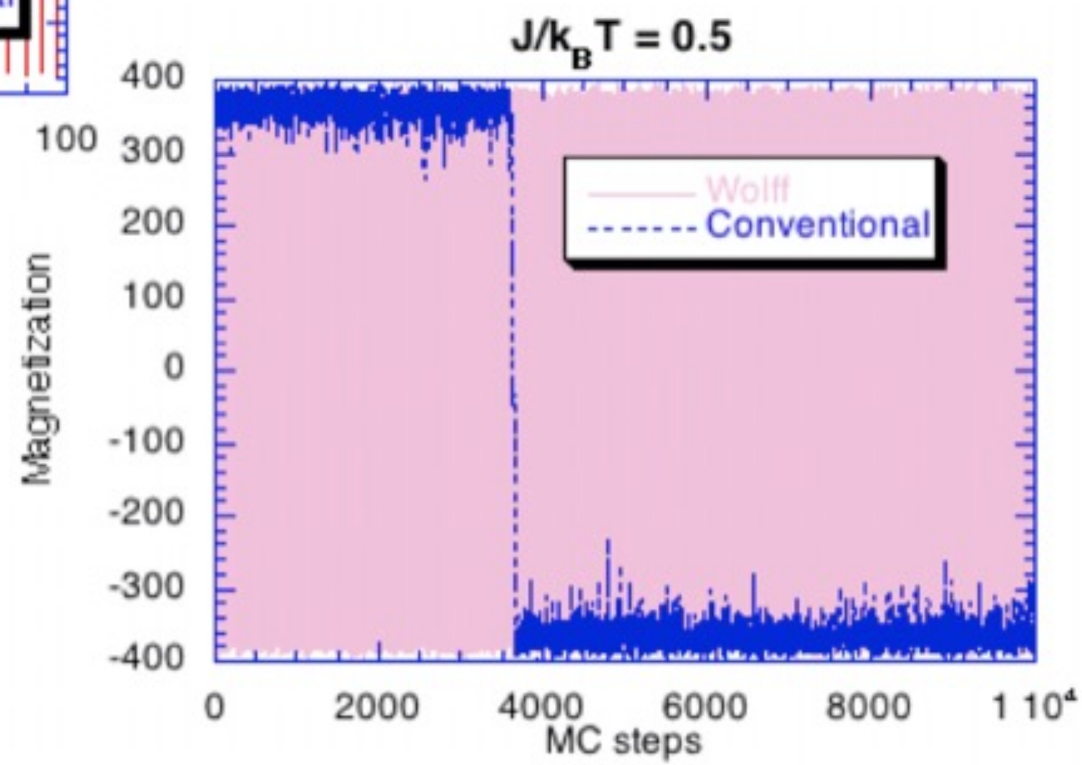
Sample Run



J/k_BT = 0.5
Magnetization
400
300
200
100
0
-100
-200
-300
-400
0
20
40
60
80
100
MC step
Wolff
Conventional



P(M)
-400
M
400



J/k_BT = 0.5
Magnetization
400
300
200
100
0
-100
-200
-300
-400
0
2000
4000
6000
8000
1 10⁴
MC steps
Wolff
Conventional