Local updates are inefficient close to Tc. More efficient way:

Cluster algorithm for the Ising model

Define bond index corresponding to pair of interacting spins

$$\text{bond} \quad b = 1, 2, \ldots, N_b, \quad \text{interacting spins } \sigma_{i(b)}, \sigma_{j(b)}$$

Number of bonds N_b = dN for a d-dimensional cubic lattice

Write the energy of the Ising ferromagnet as

$$E = -|J| \sum_{b=1}^{N_b} [\sigma_{i(b)} \sigma_{j(b)} + 1] = -\sum_{b=1}^{N_b} E_b$$

Write the partition function as

$$Z = \sum_{\sigma} \mathrm{e}^{-E(\sigma)/T} = \sum_{\sigma} \prod_{b=1}^{N_b} \mathrm{e}^{E_b/T} = \sum_{\sigma} \prod_{b=1}^{N_b} [1 + (\mathrm{e}^{E_b/T} - 1)]$$

Define bond functions with arguments 0,1 (bond variable):

$$F_b(0) = 1$$
$$F_b(1) = \mathrm{e}^{E_b/T} - 1$$

$$Z = \sum_{\sigma} \prod_{b=1}^{N_b} [F_b(0) + F_b(1)]$$

Introduce bond variables

$$\tau_b = 0, 1, \quad \tau = \{\tau_1, \tau_2, \ldots, \tau_{N_b}\}$$

Partition function can be written as sum over spins and bonds

$$Z = \sum_{\sigma} \prod_{b=1}^{N_b} [F_b(0) + F_b(1)] = \sum_{\sigma} \sum_{\tau} \prod_{b=1}^{N_b} F_b(\tau_b)$$

The functions F_b depend on the spins:

$$F_b(0) = 1$$

$$F_b(1) = \mathrm{e}^{E_b/T} - 1 = \begin{cases} \mathrm{e}^{2|J|/T} - 1, & \text{if } \sigma_{i(b)} = \sigma_{j(b)} \\ 0, & \text{if } \sigma_{i(b)} \neq \sigma_{j(b)} \end{cases}$$

$\tau_b = 1$ allowed only between parallel spins

Probabilities: For everything else fixed, probability for a given b

$$P(\tau_b) = \frac{F(\tau_b)}{F(0) + F(1)} = \frac{F(\tau_b)}{\mathrm{e}^{2|J|/T}}$$

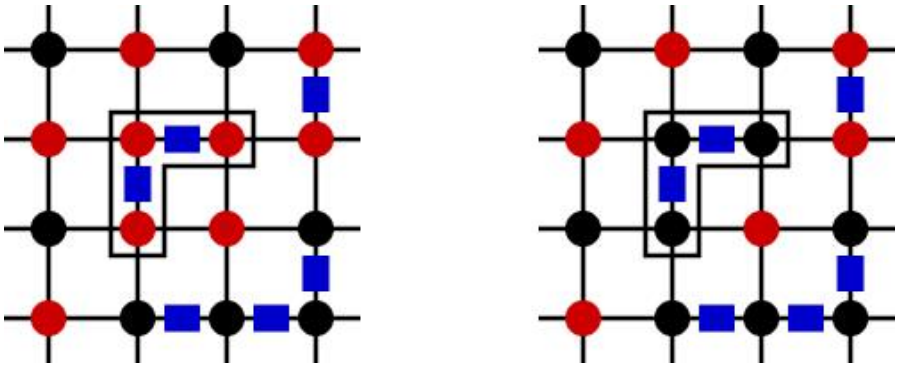
If parallel spins on bond b, probabilities for the bond variable

$$P(\tau_b = 0) = \mathrm{e}^{-2|J|/T}, \quad P(\tau_b = 1) = 1 - \mathrm{e}^{-2|J|/T}$$

If anti-parallel spins on bond b

$$P(\tau_b = 0) = 1, \quad P(\tau_b = 1) = 0$$

For a fixed bond configuration, spins forming clusters (spins connected by "filled" bonds) can be flipped and then give a configuration (term) with the same weight in Z (F_b=1 for all bonds between clusters, F_b unchanged inside cluster).



$N(\tau_b = 1) =$ No. of filled bonds

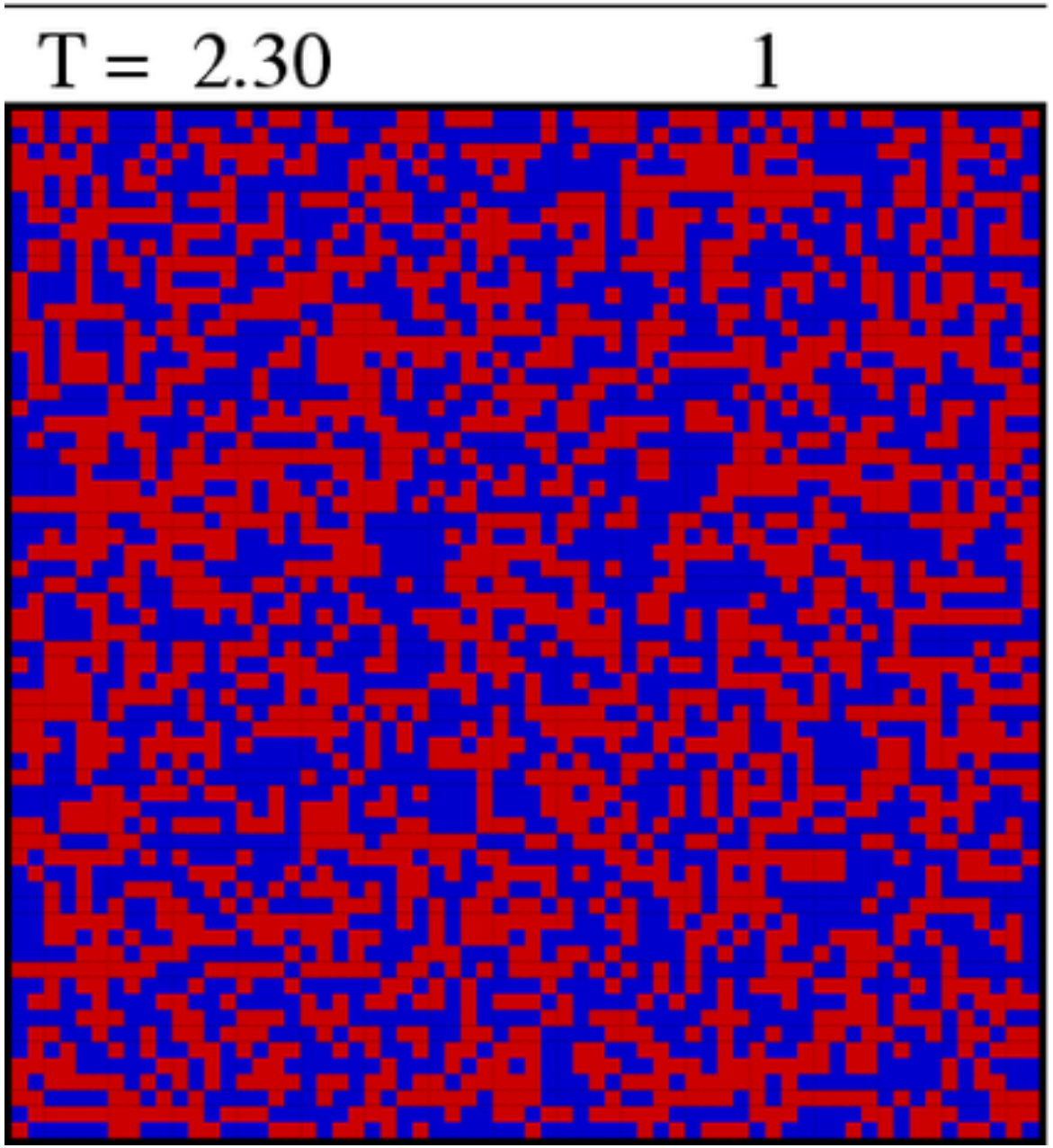
$$W = (e^{2|J|/T} - 1)^{N(\tau_b=1)}$$

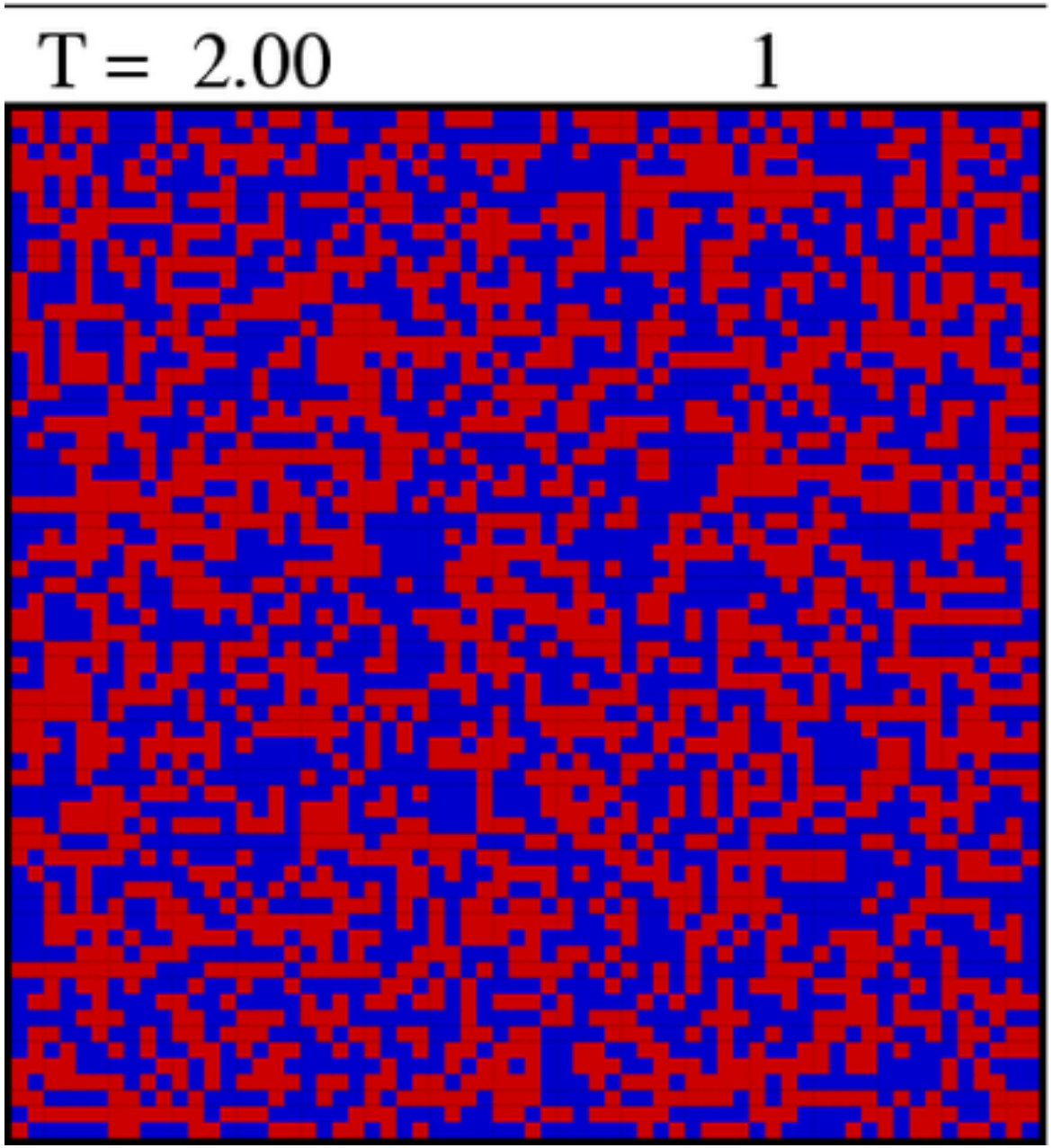
(unchanged after flip)

Spins not connected to any filled bonds are single-spin clusters

Swendsen-Wang algorithm

- Start from spin configuration
- Generate bond configuration
- Identify clusters of spins connected by bonds
- Flip each cluster with probability 1/2
- Generate new bonds with the current spins, etc


T = 2.30
1


T = 2.00
1

Quantum Monte Carlo simulations (spin-1/2 models)

Euclidean path integrals, continuous time limit

Stochastic series expansion
- discrete representation of the continuum limit

Ground state projection with valence bond states

Path integrals on the lattice, imaginary time

We want to compute a thermal expectation value

$$\langle A \rangle = \frac{1}{Z} \mathrm{Tr}\{A \mathrm{e}^{-\beta H}\}$$

where β=1/T (and possibly T→0). How to deal with the exponential operator?

"Time slicing" of the partition function

$$Z = \mathrm{Tr}\{\mathrm{e}^{-\beta H}\} = \mathrm{Tr}\left\{\prod_{l=1}^{L} \mathrm{e}^{-\Delta_\tau H}\right\} \qquad \Delta_\tau = \beta/L$$

Choose a basis and insert complete sets of states;

$$Z = \sum_{\alpha_0} \sum_{\alpha_1} \cdots \sum_{\alpha_L - 1} \langle \alpha_0 | \mathrm{e}^{-\Delta_\tau H} | \alpha_{L-1} \rangle \cdots \langle \alpha_2 | \mathrm{e}^{-\Delta_\tau H} | \alpha_1 \rangle \langle \alpha_1 | \mathrm{e}^{-\Delta_\tau H} | \alpha_0 \rangle$$

Use approximation for imaginary time evolution operator. Simplest way

$$Z \approx \sum_{\{\alpha\}} \langle \alpha_0 | 1 - \Delta_\tau H | \alpha_{L-1} \rangle \cdots \langle \alpha_2 | 1 - \Delta_\tau H | \alpha_1 \rangle \langle \alpha_1 | 1 - \Delta_\tau H | \alpha_0 \rangle$$

Leads to error $\propto \Delta_\tau$. Limit $\Delta_\tau \longrightarrow 0$ can be taken

Trotter decomposition: error $\propto \Delta_\tau^2$

Example of linear approximation and $\Delta\tau \to 0$: hard-core bosons

$$H = K = -\sum_{\langle i,j \rangle} K_{ij} = -\sum_{\langle i,j \rangle} (a_j^\dagger a_i + a_i^\dagger a_j) \qquad n_i = a_i^\dagger a_i \in \{0, 1\}$$

Equivalent to S=1/2 XY model

$$H = -2\sum_{\langle i,j \rangle} (S_i^x S_j^x + S_i^y S_j^y) = -\sum_{\langle i,j \rangle} (S_i^+ S_j^- + S_i^- S_j^+), \quad S^z = \pm\frac{1}{2} \sim n_i = 0, 1$$

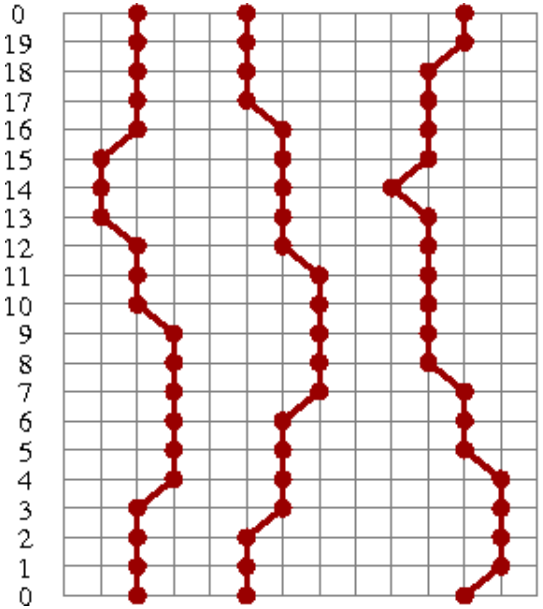
World line representation of

$$Z \approx \sum_{\{\alpha\}} \langle \alpha_0 | 1 - \Delta_\tau H | \alpha_{L-1} \rangle \cdots \langle \alpha_2 | 1 - \Delta_\tau H | \alpha_1 \rangle \langle \alpha_1 | 1 - \Delta_\tau H | \alpha_0 \rangle$$

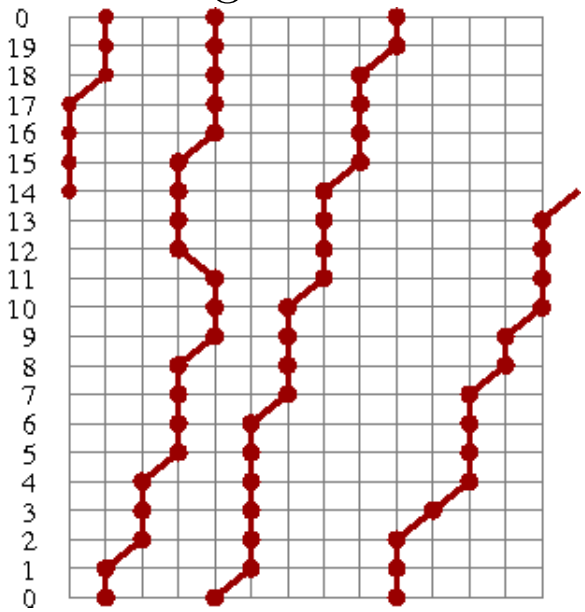
Path: One term of

$$1 + \Delta_\tau \sum_{\langle i,j \rangle} K_{ij}$$

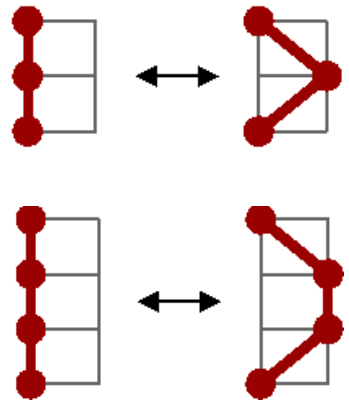
acts on each slice



winding number $= 1$



simplest world line moves for Monte Carlo sampling



$$Z = \sum_{\{\alpha\}} W(\{\alpha\}), \quad W(\{\alpha\}) = \Delta_\tau^{n_K}$$

n_K = number of "jumps"

Expectation values

$$\langle A\rangle = \frac{1}{Z}\sum_{\{\alpha\}}\langle\alpha_0|\mathrm{e}^{-\Delta_\tau}|\alpha_{L-1}\rangle\cdots\langle\alpha_2|\mathrm{e}^{-\Delta_\tau H}|\alpha_1\rangle\langle\alpha_1|\mathrm{e}^{-\Delta_\tau H}A|\alpha_0\rangle$$

We want to write this in a form suitable for MC importance sampling

$$\langle A\rangle = \frac{\sum_{\{\alpha\}} A(\{\alpha\})W(\{\alpha\})}{\sum_{\{\alpha\}} W(\{\alpha\})} \quad\longrightarrow\quad \langle A\rangle = \langle A(\{\alpha\})\rangle_W$$

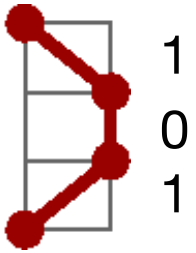
$$W(\{\alpha\}) = \text{weight}$$
$$A(\{\alpha\}) = \text{estimator}$$

For any quantity diagonal in the occupation numbers (spin z):

$$A(\{\alpha\}) = A(\alpha_n) \quad \text{or} \quad A(\{\alpha\}) = \frac{1}{L}\sum_{l=0}^{L-1} A(\alpha_l)$$

Kinetic energy (here full energy). Multiply and divide by W,

$$K\mathrm{e}^{-\Delta_\tau K} \approx K \qquad K_{ij}(\{\alpha\}) = \frac{\langle\alpha_1|K_{ij}|\alpha_0\rangle}{\langle\alpha_1|1-\Delta_\tau K|\alpha_0\rangle} \in \{0, \frac{1}{\Delta_\tau}\}$$



Average over all slices → count number of kinetic jumps

$$\langle K_{ij}\rangle = \frac{\langle n_{ij}\rangle}{\beta}, \quad \langle K\rangle = -\frac{\langle n_K\rangle}{\beta} \qquad \langle K\rangle \propto N \longrightarrow \langle n_K\rangle \propto \beta N$$

There should be of the order βN kinetic jumps

- independently of $\Delta\tau$ (when small enough)

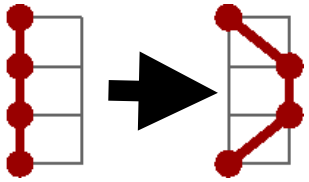
Including interactions

For any diagonal interaction V (Trotter, or split-operator, approximation)

$$e^{-\Delta_\tau H} = e^{-\Delta_\tau K} e^{-\Delta_\tau V} + \mathcal{O}(\Delta_\tau^2) \rightarrow \langle \alpha_{l+1}|e^{-\Delta_\tau H}|\alpha_l\rangle \approx e^{-\Delta_\tau V_l}\langle \alpha_{l+1}|e^{-\Delta_\tau K}|\alpha_l\rangle$$

Product over all times slices →

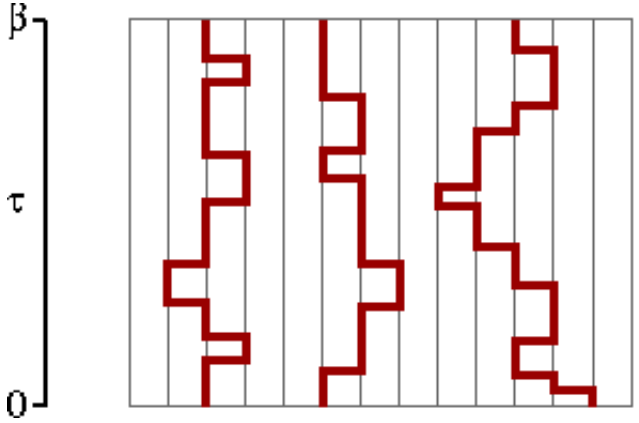
$$W(\{\alpha\}) = \Delta_\tau^{n_K} \exp\left(-\Delta_\tau \sum_{l=0}^{L-1} V_l\right)$$



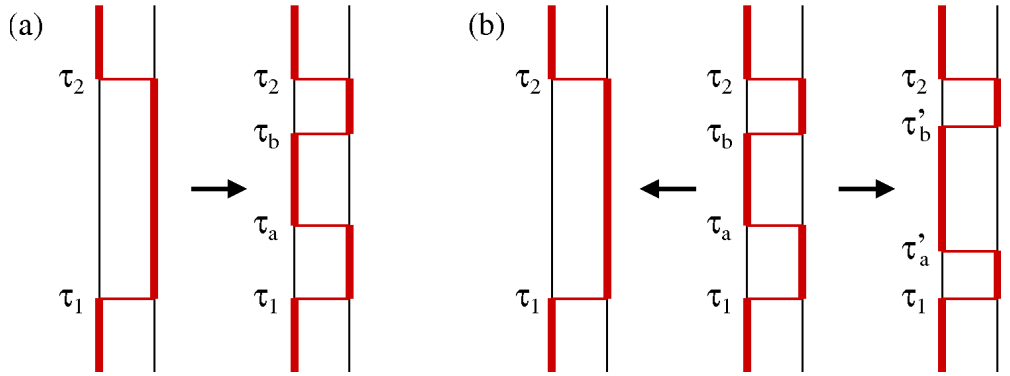
$$P_{\text{acc}} = \min\left[\Delta_\tau^2 \exp\left(-\frac{V_{\text{new}}}{V_{\text{old}}}\right), 1\right]$$

The continuous time limit

Limit $\Delta_\tau \rightarrow 0$: number of kinetic jumps remains finite, store events only



Special methods (**loop and worm updates**) developed for efficient sampling of the paths in the continuum



local updates (problem when $\Delta_\tau \rightarrow 0$?)

- consider probability of inserting/removing events within a time window
- non-zero integrated probabilitis for insertion at all times, choose random time.

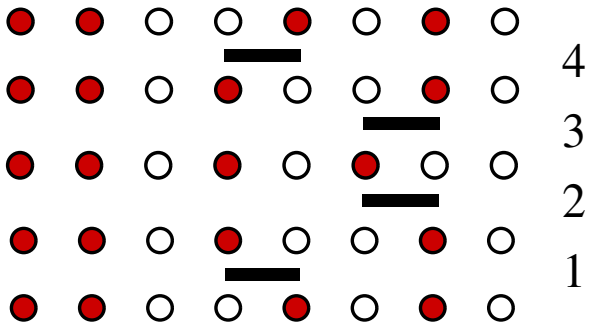
Series expansion representation

Start from the **Taylor expansion** (no approximation)

$$Z = \mathrm{Tr}\{\mathrm{e}^{-\beta H}\} = \sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} \sum_{\alpha_0} \langle \alpha_0 | H^n | \alpha_0 \rangle$$

Index sequence (string) referring to terms of H

$$H = \sum_{i=1}^{m} H_i \qquad S_n = (a_1, a_2, \ldots, a_n) \qquad a_i \in \{1, \ldots, m\}$$



Break up H^n into strings:

$$Z = \sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} \sum_{\alpha_0} \sum_{S_n} \langle \alpha_0 | H_{a_n} \cdots H_{a_2} H_{a_1} | \alpha_0 \rangle$$

We should have (always possible): $H_i |\alpha_j\rangle \propto |\alpha_k\rangle$

- **no branching** during propagation with operator string
- some strings not allowed (illegal operations)

For hard-core bosons the (allowed) path weight is: $W(S_n, \alpha_0) = \dfrac{\beta^n}{n!}$

We can make this look more similar to a path integral by introducing partially propagated states: $|\alpha_p\rangle = H_{a_p} \cdots H_{a_2} H_{a_1} |\alpha_0\rangle$

$$Z = \sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} \sum_{\alpha_0} \sum_{S_n} \langle \alpha_0 | H_{\alpha_n} | \alpha_{n-1} \rangle \langle \alpha_{n-1} | \cdots | \alpha_1 \rangle \langle \alpha_1 | H_{a_1} | \alpha_0 \rangle$$

$$|\alpha_n\rangle = |\alpha_0\rangle$$

Same-looking paths, different-looking weights
- but become equivalent with time continuum in path integral

Energy: $$\langle H \rangle = \frac{1}{Z} \sum_{n=0}^{\infty} \frac{(-\beta)^n}{n!} \sum_{\alpha_0} \langle \alpha_0 | H^n H | \alpha_0 \rangle$$

Relabel terms of n-sum: replace n+1 by n

$$\langle H \rangle = -\frac{1}{Z} \sum_{n=1}^{\infty} \frac{(-\beta)^n}{n!} \frac{n}{\beta} \sum_{\alpha_0} \langle \alpha_0 | H^n | \alpha_0 \rangle$$

we can extend the sum to include n=0, because that term vanishes

Therefore the energy is: $E = -\langle n \rangle / \beta$

Can also derive specific heat: $C = \langle n^2 \rangle - \langle n \rangle^2 - \langle n \rangle$

Follows: $\langle n \rangle \propto \beta N, \quad \sigma_n \propto \sqrt{\beta N}$

Fixed string-length scheme

- n fluctuating → varying size of the sampled configurations
- the expansion can be truncated at some n_{max}=L
  (exponentially small error if large enough)
- cutt-off at n=L, fill in operator string with unit operators **H_0=I**

n=10 | H_4 | H_7 | H_1 | H_6 | H_2 | H_1 | H_8 | H_3 | H_3 | H_5 | $\Longrightarrow$

L=14 | H_4 | I | H_7 | I | H_1 | H_6 | I | H_2 | H_1 | H_8 | H_3 | H_3 | I | H_5 |

- conisider all possible locations in the sequence $\binom{L}{n}^{-1} = \frac{n!(L-n)!}{L!}$
- overcounting of original strings, correct by

$$Z = \sum_{\alpha_0} \sum_{S_L} \frac{(-\beta)^n (L-n)!}{L!} \langle \alpha_0 | H_{a_m} \cdots H_{a_2} H_{a_1} | \alpha_0 \rangle$$

Here n is the number of H_i, i>0 instances in the sequence of L ops
- the summation over n is now implicit

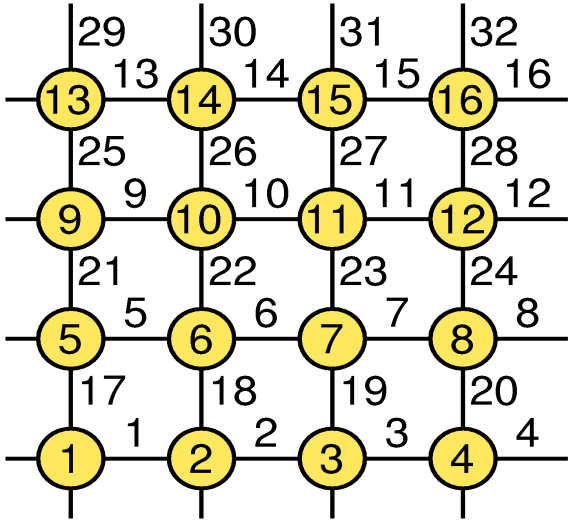
L can be chosen automatically by the simulation (shown later)

Stochastic Series expansion (SSE): S=1/2 Heisenberg model

Write H as a bond sum for arbitrary lattice

$$H = J \sum_{b=1}^{N_b} \mathbf{S}_{i(b)} \cdot \mathbf{S}_{j(b)},$$

2D square lattice
bond and site labels



Diagonal (1) and off-diagonal (2) bond operators

$$H_{1,b} = \frac{1}{4} - S^z_{i(b)} S^z_{j(b)},$$

$$H_{2,b} = \frac{1}{2}(S^+_{i(b)} S^-_{j(b)} + S^-_{i(b)} S^+_{j(b)}).$$

$$H = -J \sum_{b=1}^{N_b} (H_{1,b} - H_{2,b}) + \frac{J N_b}{4}$$

Four non-zero matrix elements

$$\langle \uparrow_{i(b)} \downarrow_{j(b)} | H_{1,b} | \uparrow_{i(b)} \downarrow_{j(b)} \rangle = \frac{1}{2} \qquad \langle \downarrow_{i(b)} \uparrow_{j(b)} | H_{2,b} | \uparrow_{i(b)} \downarrow_{j(b)} \rangle = \frac{1}{2}$$

$$\langle \downarrow_{i(b)} \uparrow_{j(b)} | H_{1,b} | \downarrow_{i(b)} \uparrow_{j(b)} \rangle = \frac{1}{2} \qquad \langle \uparrow_{i(b)} \downarrow_{j(b)} | H_{2,b} | \downarrow_{i(b)} \uparrow_{j(b)} \rangle = \frac{1}{2}$$

Partition function

$$Z = \sum_{\alpha} \sum_{n=0}^{\infty} (-1)^{n_2} \frac{\beta^n}{n!} \sum_{S_n} \left\langle \alpha \left| \prod_{p=0}^{n-1} H_{a(p),b(p)} \right| \alpha \right\rangle$$

n_2 = number of a(i)=2
(off-diagonal operators)
in the sequence

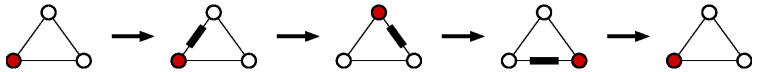
Index sequence: $S_n = [a(0), b(0)], [a(1), b(1)], \ldots, [a(n-1), b(n-1)]$

For fixed-length scheme

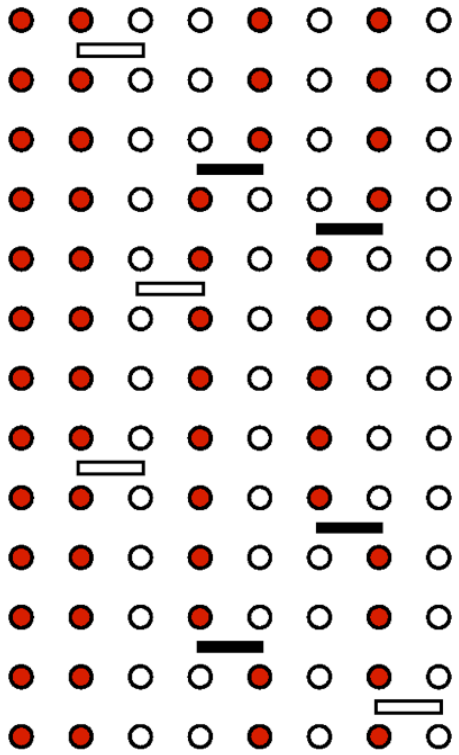
$$Z = \sum_{\alpha} \sum_{S_L} (-1)^{n_2} \frac{\beta^n (L-n)!}{L!} \left\langle \alpha \left| \prod_{p=0}^{L-1} H_{a(p),b(p)} \right| \alpha \right\rangle \qquad W(\alpha, S_L) = \left(\frac{\beta}{2}\right)^n \frac{(L-n)!}{L!}$$

Propagated states: $|\alpha(p)\rangle \propto \prod_{i=0}^{p-1} H_{a(i),b(i)} |\alpha\rangle$

W>0 (n_2 even) for bipartite lattice
Frustration leads to **sign problem**



i =	1	2	3	4	5	6	7	8
$\sigma(i)$ =	-1	+1	-1	-1	+1	-1	+1	+1



p	$a(p)$	$b(p)$	$s(p)$
11	1	2	4
10	0	0	0
9	2	4	9
8	2	6	13
7	1	3	6
6	0	0	0
5	0	0	0
4	1	2	4
3	2	6	13
2	0	0	0
1	2	4	9
0	1	7	14

In a program:

s(p) = operator-index string
- **s(p) = 2*b(p) + a(p)-1**
- diagonal; s(p) = even
- off-diagonal; s(p) = off

σ(i) = spin state, i=1,...,N
- only one has to be stored

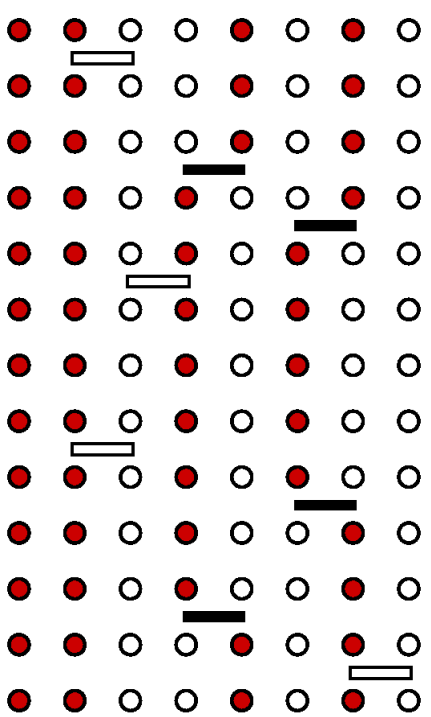
SSE effectively provides a discrete representation of the time continuum!
- computational advantage; only integer operations in sampling

Monte Carlo sampling scheme

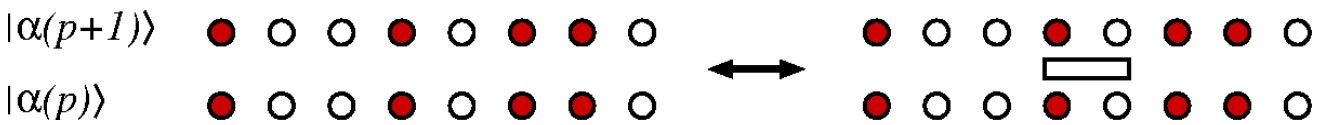
Change the configuration; $(\alpha, S_L) \to (\alpha', S'_L)$

$$W(\alpha, S_L) = \left(\frac{\beta}{2}\right)^n \frac{(L-n)!}{L!}$$

$$P_{\text{accept}} = \min\left[\frac{W(\alpha', S_L)}{W(\alpha, S_L)} \frac{P_{\text{select}}(\alpha', S'_L \to \alpha, S_L)}{P_{\text{select}}(\alpha, S_L \to \alpha', S'_L)}, 1\right]$$



Diagonal update: $[0,0]_p \leftrightarrow [1,b]_p$



Attempt at p=0,...,L-1. Need to know |α(p)>

- generate by flipping spins when off-diagonal operator

$$P_{\text{select}}(a = 0 \to a = 1) = 1/N_b, \quad (b \in \{1, \ldots, N_b\})$$

$$P_{\text{select}}(a = 1 \to a = 0) = 1$$

$$\frac{W(a=1)}{W(a=0)} = \frac{\beta/2}{L-n} \qquad \frac{W(a=0)}{W(a=1)} = \frac{L-n+1}{\beta/2}$$

n is the current power

- n → n+1 (a=0 → a=1)
- n → n-1 (a=1 → a=0)

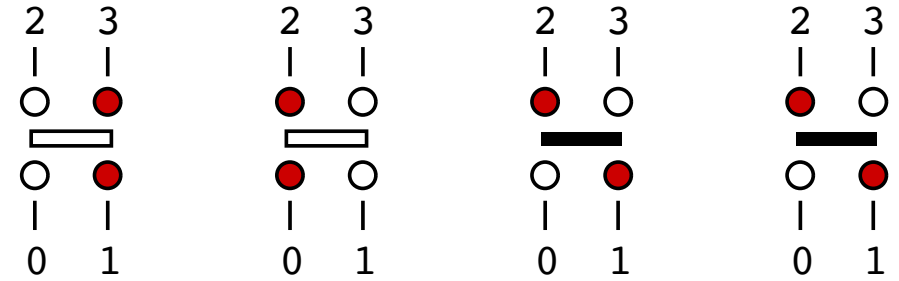
Acceptance probabilities

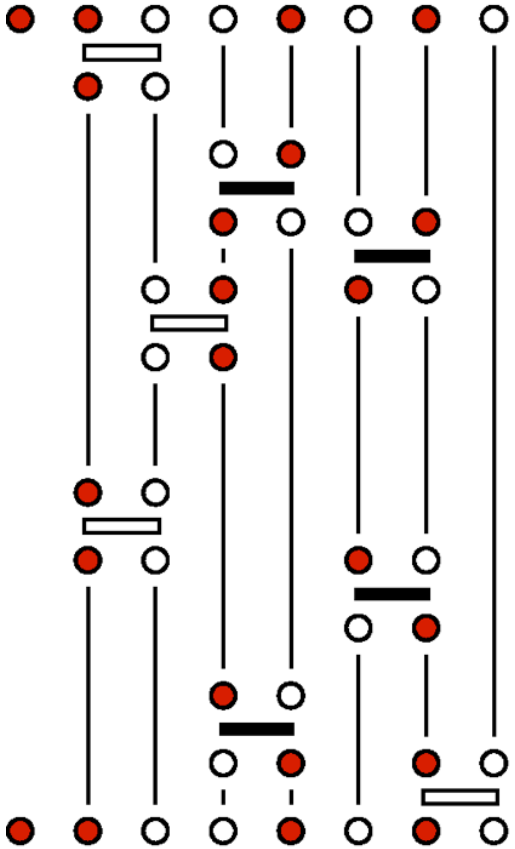
$$P_{\text{accept}}([0,0] \to [1,b]) = \min\left[\frac{\beta N_b}{2(L-n)}, 1\right]$$

$$P_{\text{accept}}([1,b] \to [0,0]) = \min\left[\frac{2(L-n+1)}{\beta N_b}, 1\right]$$

Linked vertex storage

The “legs” of a vertex represents the spin states before (below) and after (above) an operator has acted





p	v	X(v)	v	X(v)	v	X(v)	v	X(v)
11	44	18	45	30	46	16	47	17
10	40	-	41	-	42	-	43	-
9	36	31	37	7	38	4	39	5
8	32	14	33	15	34	12	35	0
7	28	19	29	6	30	45	31	36
6	24	-	25	-	26	-	27	-
5	20	-	21	-	22	-	23	-
4	16	46	17	47	18	44	19	28
3	12	34	13	2	14	32	15	33
2	8	-	9	-	10	-	11	-
1	4	38	5	39	6	29	7	37
0	0	35	1	3	2	13	3	1
	l=0		l=1		l=2		l=3	

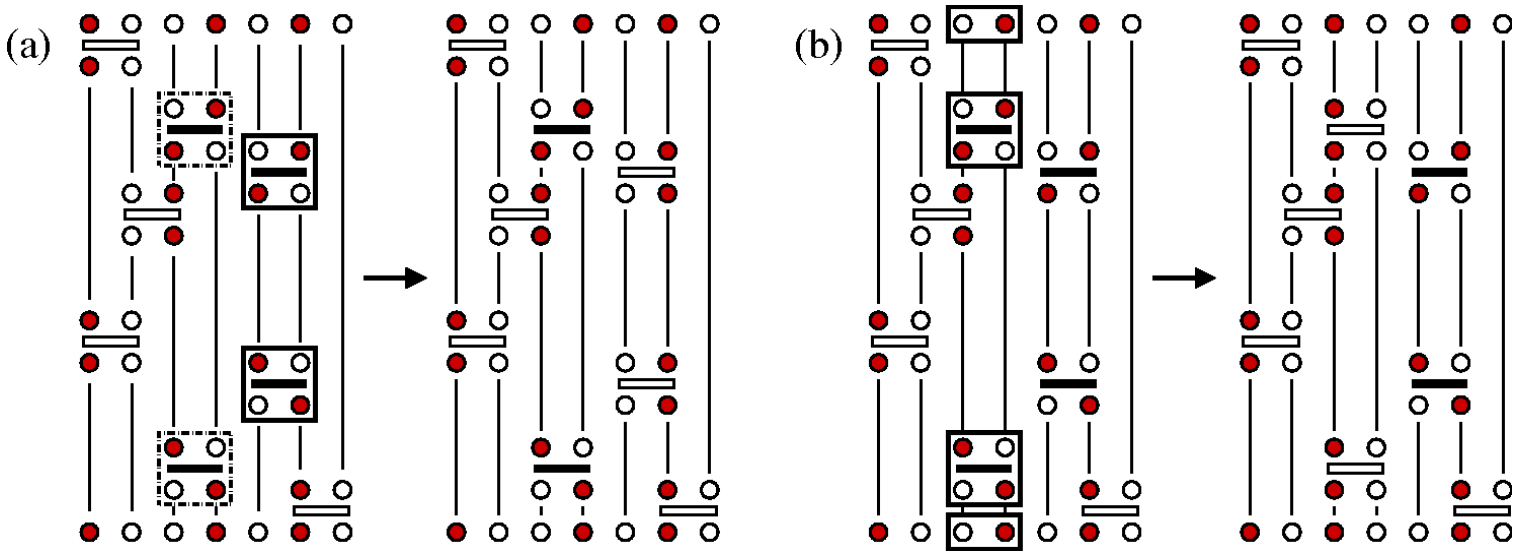
X( ) = vertex list

- operator at p→X(v)
  v=4p+l, l=0,1,2,3
- links to next and previous leg

Spin states between operations are redundant; represented by links

- network of linked vertices will be used for loop updates of vertices/operators

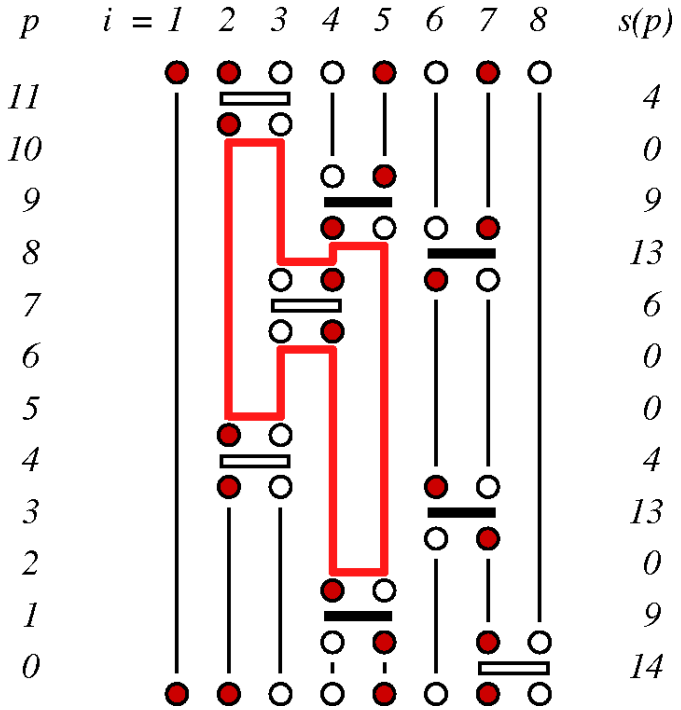
Off-diagonal updates

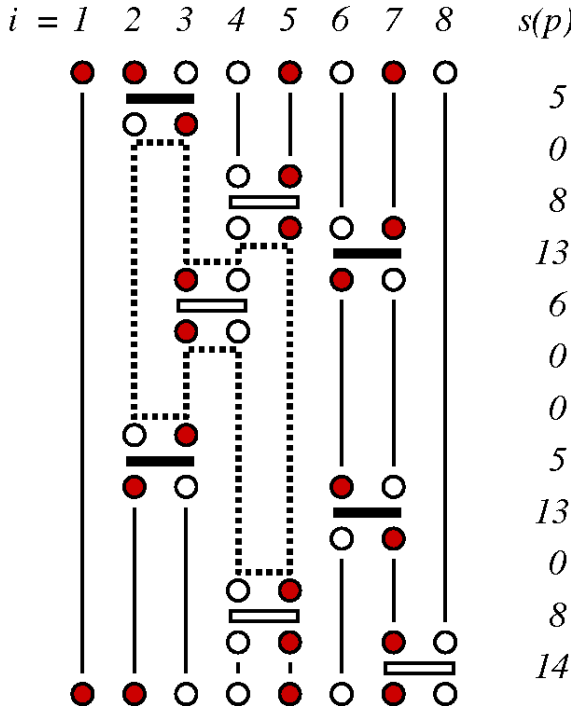


Local update

Change the type of two operators

- constraints
- inefficient
- cannot change winding numbers





Operator-loop update

- Many spins and operators can be changed simultaneously
- can change winding numbers

Determination of the cut-off L

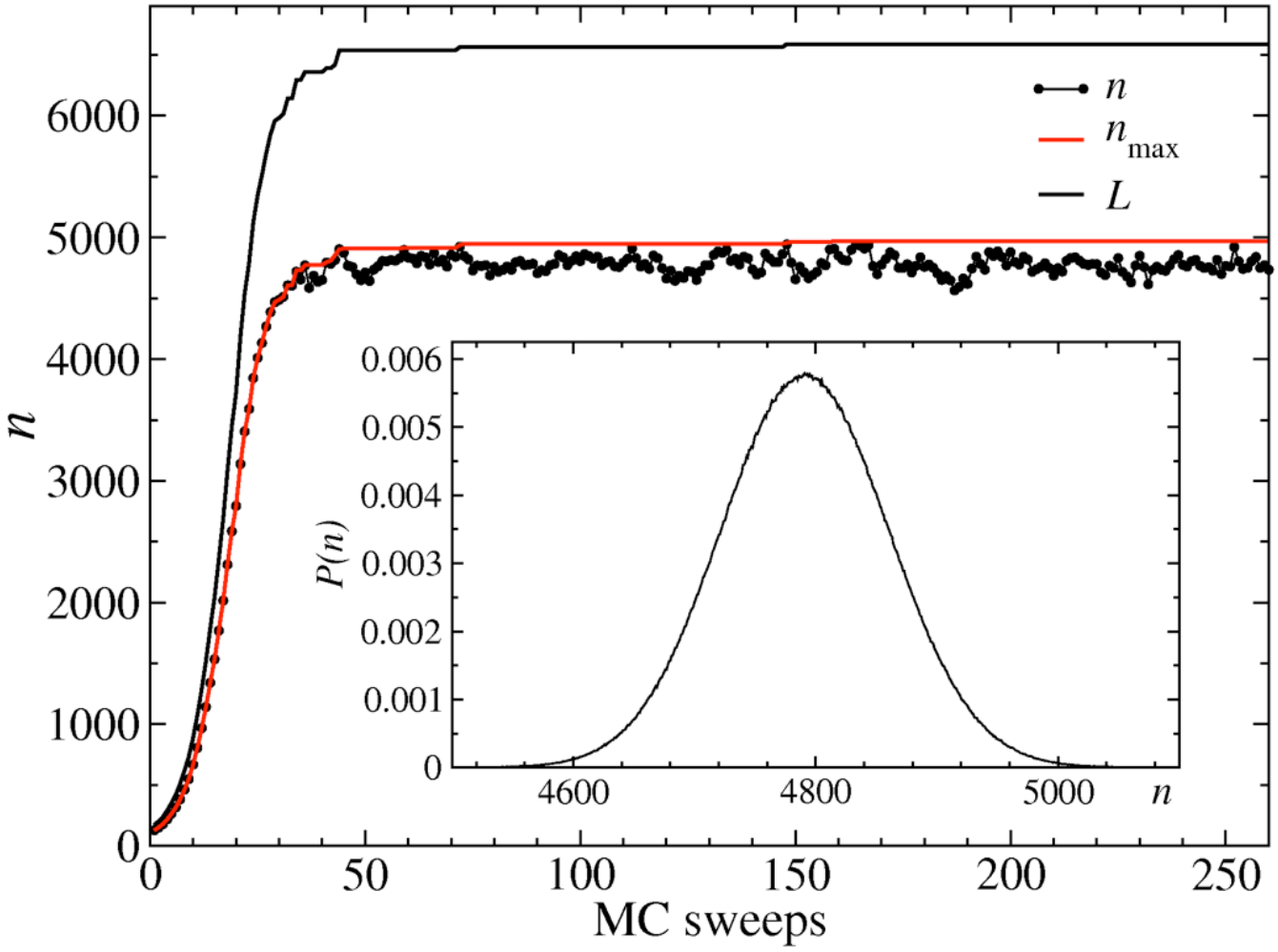
- adjust during equilibration
- start with arbitrary (small) n

Keep track of number of operators n

- increase L if n is close to current L
- e.g., ***L=n+n/3***

Example

- 16×16 system, β=16 ⇒
- evolution of L
- n distribution after equilibration
- truncation is no approximation



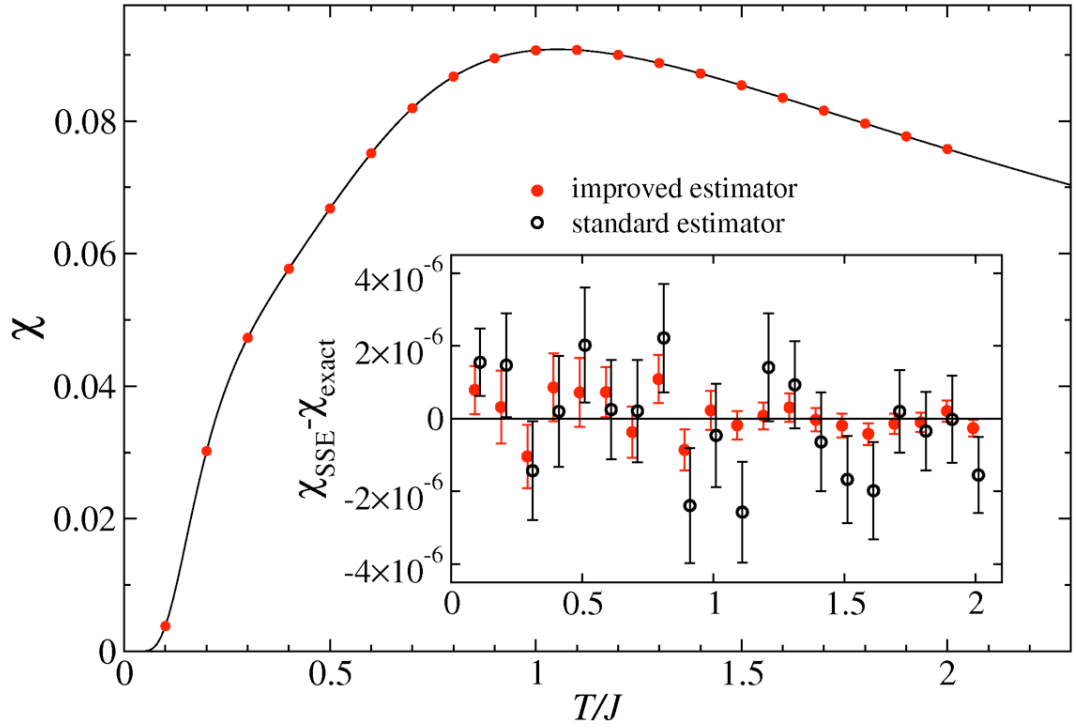
Does it work?

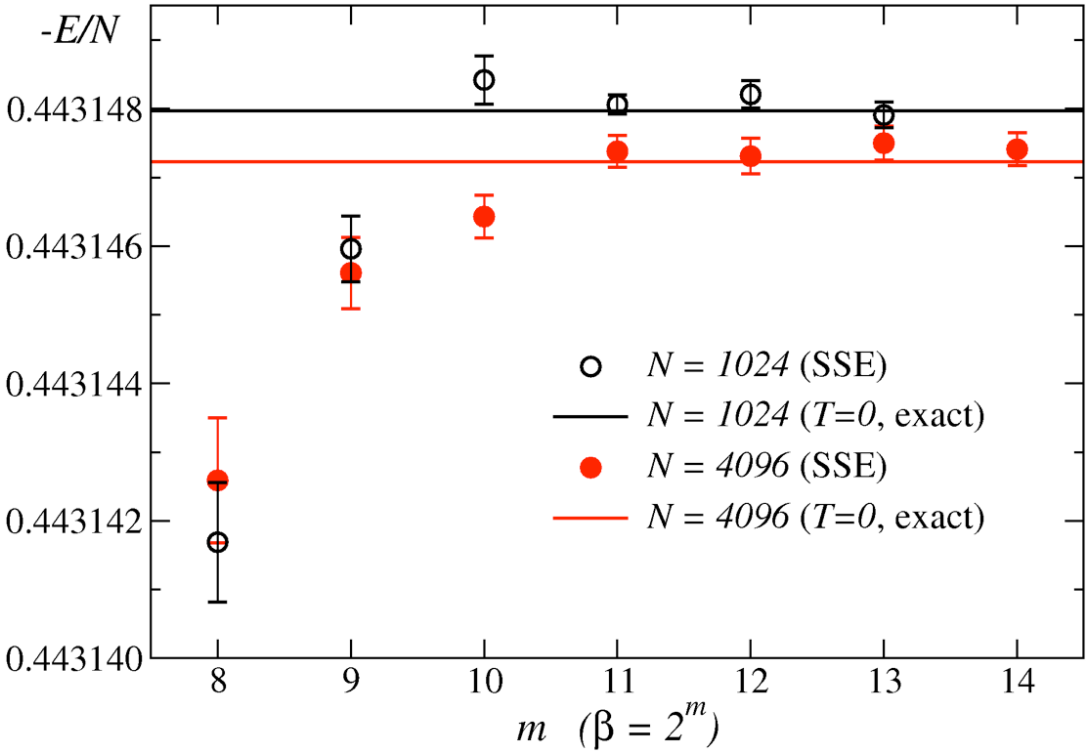
Compare with exact results

- 4×4 exact diagonalization
- Bethe Ansatz; long chains

Susceptibility of the 4×4 lattice ⇒

- SSE results from 10^{10} sweeps
- improved estimator gives smaller error bars at high T (where the number of loops is larger)





⇐ Energy for long 1D chains

- SSE results for 10^6 sweeps
- Bethe Ansatz ground state E/N
- SSE can achieve the ground state limit (T→0)