Introduction to the Stochastic Series Expansion method

Anders Sandvik, Boston University

- **Illustration of concept; classical Monte Carlo example**
- **Detailed account of SSE for the S=1/2 Heisenberg model**

This presentation is based on material available at
http://physics.bu.edu/~sandvik/programs/

A simple SSE program (Fortran90) for the
2D Heisenberg model can be downloaded from this site

Warm-up: SSE for a classical problem

Classical thermal expectation value

$$\langle f \rangle = \frac{1}{Z} \sum_{\{\sigma\}} f(\sigma) \mathrm{e}^{-\beta E(\sigma)}, \qquad Z = \sum_{\{\sigma\}} \mathrm{e}^{-\beta E(\sigma)}$$

Classical (e.g., Ising) spins: $\sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_N\}$

Classical Monte Carlo: Importance sampling of spin configurations

Probability of generating a configuration

$$P(\sigma) = \frac{1}{Z} W(\sigma), \qquad W(\sigma) = \mathrm{e}^{-\beta E(\sigma)}$$

Estimate of expectation value based on sampled configurations

$$\langle f \rangle = \langle f \rangle_W \approx \frac{1}{N_{\text{samples}}} \sum_i f(\sigma[i])$$

Imagine that we are not able to evaluate the exponential function
How could we proceed then?

Use Taylor expansion of the exponential function

$$\langle f \rangle = \frac{1}{Z} \sum_{\{\sigma\}} \sum_{n=0}^{\infty} f(\sigma) \frac{(-\beta E)^n}{n!}, \qquad Z = \sum_{\{\sigma\}} \sum_{n=0}^{\infty} \frac{(-\beta E)^n}{n!}$$

Expansion power n is a new "dimension" of the configuration space

To ensure positive-definitness we may have to shift E (must be < 0)

$$E(\sigma) \rightarrow E(\sigma) - \epsilon$$

The sampling weight for the **configurations (□,n)** is

$$W(\sigma, n) = \frac{\beta^n [\epsilon - E(\sigma)]^n}{n!}$$

The function to be averaged (estimator) **f(□) is the same as before**; it does not depend on n

$$\langle f \rangle = \langle f \rangle_W \approx \frac{1}{N_{\text{samples}}} \sum_i f(\sigma[i])$$

However, **if f(□) is a function of the energy it can be rewritten as a function of n only!**

Define: $H(\sigma) = \epsilon - E(\sigma)$

$$\langle H \rangle = \frac{1}{Z} \sum_{\sigma,n} H(\sigma) W(\sigma,n), \quad Z = \sum_{\sigma,n} W(\sigma,n), \qquad W(\sigma,n) = \frac{\beta^n H(\sigma)^n}{n!}$$

Shift summation index: **m=n+1**

$$\sum_{\sigma,n} H(\sigma) W(\sigma,n) = \sum_{\sigma,m} \frac{m}{\beta} W(\sigma,m)$$

Therefore the energy expectation value is

$$\langle H \rangle = \frac{1}{\beta} \langle n \rangle_W \;\Rightarrow\; E = \epsilon - \frac{1}{\beta} \langle n \rangle_W$$

We can also easily obtain

$$\langle H^2 \rangle = \frac{1}{\beta^2} \langle n(n-1) \rangle_W$$

And thus the specific heat $C = \beta^{-1}(\langle E^2 \rangle - \langle E \rangle^2)$ is

$$C = \frac{1}{\beta} (\langle n^2 \rangle - \langle n \rangle^2 - \langle n \rangle)$$

What range of expansion orders n is sampled?

From the preceding results we obtain

$$\langle n \rangle = \beta(\epsilon - E)$$

$$\langle n^2 \rangle - \langle n \rangle^2 = \beta(C + \epsilon - E)$$

Consider low T; C □ 0

$$\langle n^2 \rangle - \langle n \rangle^2 = \langle n \rangle$$

Thus, for a system with N spins:

Average expansion order $\propto \beta N$

Width of distribution $\propto \sqrt{\beta N}$

These results hold true for quantum systems as well

In the quantum case H consists of non-commuting operators:
H^n **requires more complicated treatment**

Quantum-mechanical SSE

Thermal expectation value

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

Choose a basis and Taylor expand the exponential operator

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

Write the hamiltonian as a sum of local operators

$$H = -\sum_{a,b} H_{a,b}$$

a = operator type (e.g., 1=diagonal, 2=off-diagonal)
b = lattice unit (e.g., bond connecting sites i,j)

such that for every a, b: $H_{a,b}|\alpha\rangle = h_{a,b}(\alpha)|\alpha'\rangle$ (no branching)

Write the powers of H in terms of “strings” of these operators

$$(-H)^n = \sum_{\{H_{ab}\}}\prod_{p=1}^{n} H_{a(p),b(p)}$$

Operator strings of varying length n

• as in the classical case $\langle n \rangle = -\beta \langle H \rangle$

Fixed-length operator strings: introduce unit operator: $H_{0,0} = 1$

Expansion cut-off M: add M-n unit operators to each string

• there are M!/n!(M-n)! ways of doing this

$$(-H)^n = \sum_{\{H_{ab}\}} \frac{(M-n)!n!}{M!} \prod_{p=1}^{M} H_{a(p),b(p)}$$

n = number of non-[0,0] operators

The truncation should not be considered an approximation

• M can be chosen such that the **truncation error is negligible**

$$Z = \sum_{\alpha} \sum_{\{H_{ab}\}} \frac{\beta^n (M-n)!}{M!} \left\langle \alpha \left| \prod_{i=1}^{M} H_{a(i),b(i)} \right| \alpha \right\rangle$$

The terms $(\alpha, \{H_{ab}\})$ are sampled according to weight in this sum

• requires positive-definiteness

• to this end, a constant may have to be added to diagonal H_{ab}

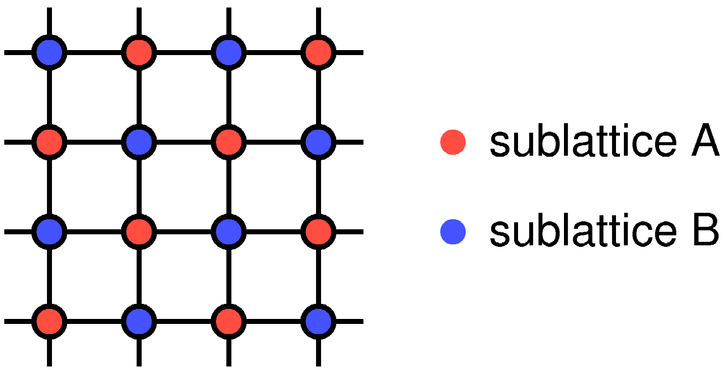
• there can still be a “sign problem” arising from off-diagonal H_{ab}

SSE algorithm for the S=1/2 Heisenberg model

- The algorithm for this model is particularly simple and efficient

$$H = J \sum_{\langle i,j \rangle} \mathbf{S}_i \cdot \mathbf{S}_j$$

Consider **bipartite lattice** (sign problem for frustrated systems)



Standard z-component basis:

$$|\alpha\rangle = |S_1^z, S_2^z, \ldots, S_N^z\rangle, \qquad S_i^z = \pm\frac{1}{2}$$

Bond operators: bond b connects sites i(b),j(b)

$$H = \sum_{b=1}^{B} \left[S_{i(b)}^z S_{j(b)}^z + \frac{1}{2}\left(S_{i(b)}^+ S_{j(b)}^- + S_{i(b)}^- S_{j(b)}^+ \right) \right]$$

$$H = -\sum_{b=1}^{B}\sum_{a=1}^{2} H_{a,b}$$

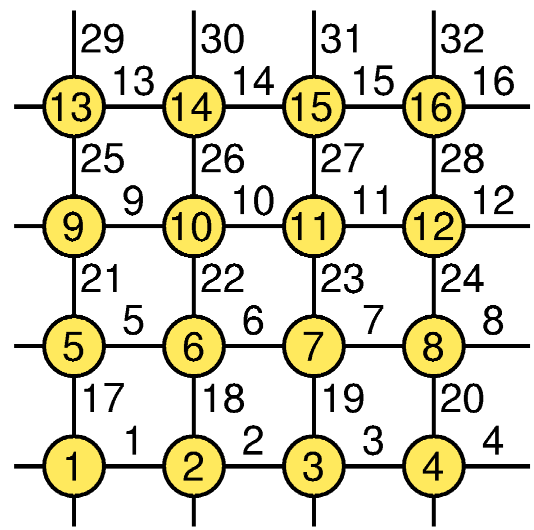
Diagonal and **off-diagonal** bond operators

$$H_{1,b} = \frac{1}{4} - S^z_{i(b)} S^z_{j(b)}, \qquad H_{2,b} = \frac{1}{2}\left(S^+_{i(b)} S^-_{j(b)} + S^-_{i(b)} S^+_{j(b)}\right)$$

A minus sign in front of the off-diagonal H_{2b} is neglected

- this corresponds to a **sublattice rotation**; 180 degree rotation in the xy-plane of the spin operators on sublattice B
- The **sign is irrelevant for a bipartite lattice** (will be shown later)

SSE operator string $\prod_{p=1}^{M} H_{a(p),b(p)}$



Represented in the computer program by

$$\texttt{opstring[p]} = 2b(p) + a(p) - 1$$

Spin state |□> represented by

$$\texttt{spin[i]} = 2S^z_i$$

SSE partition function

$$Z = \sum_{\alpha} \sum_{\{H_{ab}\}} \frac{\beta^n (M-n)!}{M!} \left\langle \alpha \left| \prod_{i=1}^{M} H_{a(i),b(i)} \right| \alpha \right\rangle$$

Both H_{1b} and H_{2b} give 0 when acting on parallel spins

• non-zero matrix element = 1/2 in both cases

Define **propagated states**

$$|\alpha(p)\rangle = \prod_{j=1}^{p} H_{a(j),b(j)} |\alpha\rangle \qquad |\alpha\rangle = |\alpha(0)\rangle$$

For a contributing configuration: $|\alpha(M)\rangle = |\alpha(0)\rangle$ (periodic)

The **configuration weight** is then

$$W(\alpha, \{H_{ab}\}) = \left(\frac{\beta}{2}\right)^n \frac{(M-n)!}{M!}$$

Periodicity requires an **even number of spin flips**

• This is why the sign of H_{2b} is irrelevant for a bipartite lattice

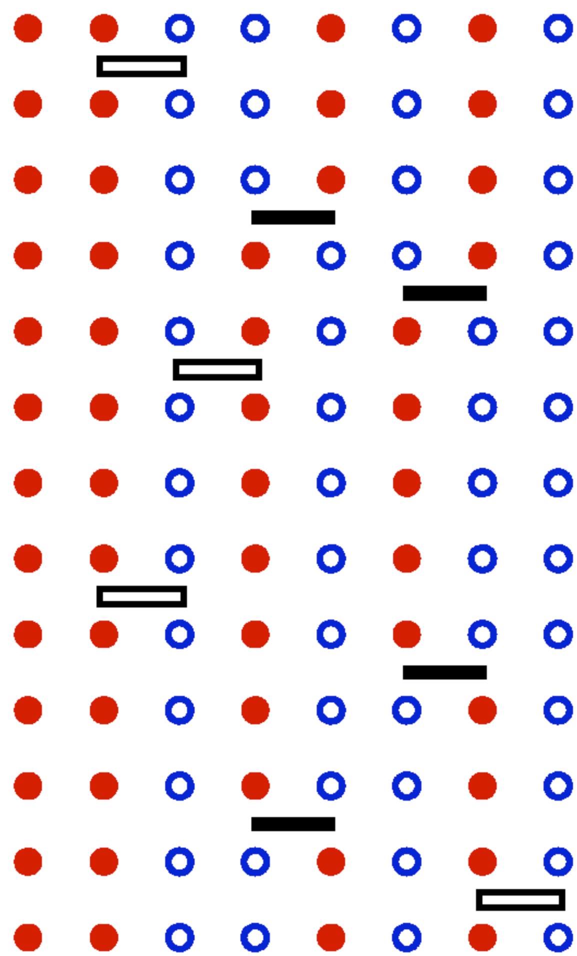
• For a frustrated lattice an odd number of flips is possible

Graphical representation

- 1D example; 8 spins, M=12

1D: bond b connects sites b and b+1

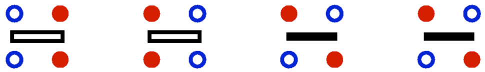
```
       i =  1  2  3  4  5  6  7  8
spin[i] = +1 +1 -1 -1 +1 -1 +1 -1
```



p	a(p)	b(p)	opstring[p]
1	1	2	4
2	0	0	0
3	2	4	9
4	2	6	13
5	1	3	6
6	0	0	0
7	0	0	0
8	1	2	4
9	2	6	13
10	0	0	0
11	2	4	9
12	1	7	14

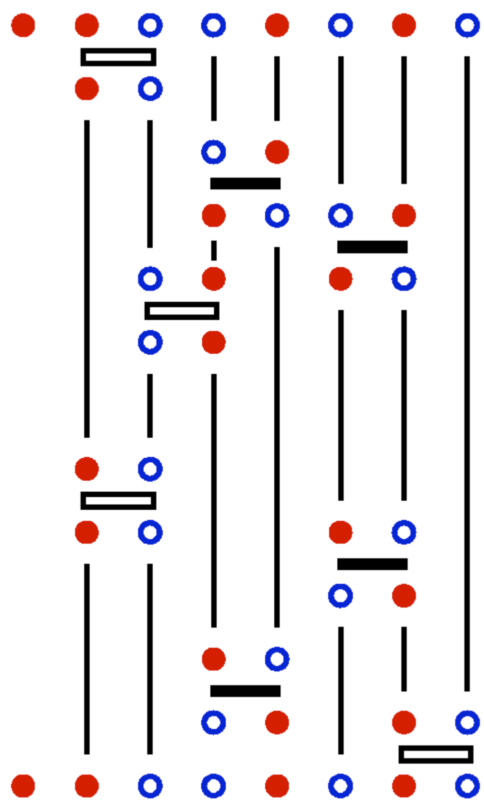
Linked-list representation

- vertex: operator and spins before and after the operator has acted



- replace spins between vertices by **links**

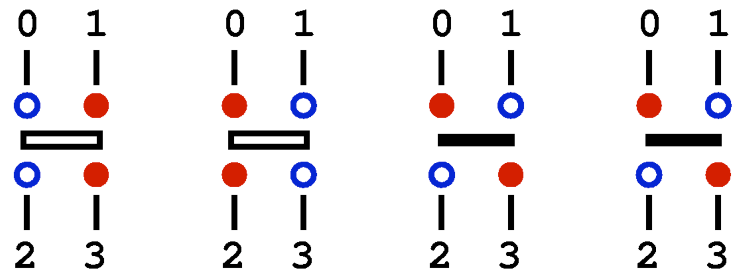
```
      i = 1  2  3  4  5  6  7  8
spin[i] = +1 +1 -1 -1 +1 -1 +1 -1
```



p	a(p)	b(p)	opstring[p]
1	1	2	4
2	0	0	0
3	2	4	9
4	2	6	13
5	1	3	6
6	0	0	0
7	0	0	0
8	1	2	4
9	2	6	13
10	0	0	0
11	2	4	9
12	1	7	14

- **linked vertex list** used in some parts of the program

A vertex has 4 “legs”, numbered l=0,1,2,3:



position **p** of operator in operator string `opstring[p]`, vertex leg **l**

□ position **v** in linked vertex list: `v=1+l+4*(p-1)`

`vertexlist[v]` contains the element # to which v is linked

	l = 0	1	2	3	p
[v] vertexlist[v]:	[1] 31	[2] 32	[3] 29	[4] 17	1
	[5] 0	[6] 0	[7] 0	[8] 0	2
	[9] 43	[10] 44	[11] 18	[12] 42	3
	[13] 35	[14] 47	[15] 33	[16] 34	4
	[17] 4	[18] 11	[19] 30	[20] 41	5
	[21] 0	[22] 0	[23] 0	[24] 0	6
	[25] 0	[26] 0	[27] 0	[28] 0	7
	[29] 3	[30] 18	[31] 1	[32] 2	8
	[33] 15	[34] 16	[35] 13	[36] 45	9
	[37] 0	[38] 0	[39] 0	[40] 0	10
	[41] 20	[42] 12	[43] 9	[44] 10	11
	[45] 36	[46] 48	[47] 14	[48] 46	12

Sampling the SSE configurations; updates

1) **Diagonal update**

   - replace unit operator by diagonal operator, and vice versa

   $$H_{0,0} \leftrightarrow H_{1,b}$$

2) **Off-diagonal update** (local or loop)

   - change the operator type, diagonal $\longleftrightarrow$ off-diagonal, for two (local) or several (loop) operators

   $$\{H_{a_1,b_1}, H_{a_2,b_2}, \ldots, H_{a_m,b_m}\} \leftrightarrow \{H_{3-a_1,b_1}, H_{3-a_2,b_2}, \ldots, H_{3-a_m,b_m}\}$$

3) **Flip spins** in the state |□>

   - unconstrained "free" spins; weight unchanged after flip
   - only possible at high temperatures; strictly not necessary

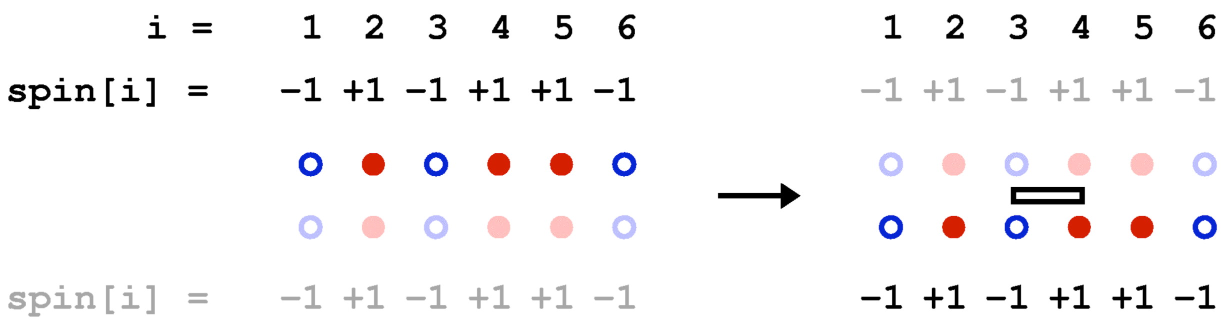
   $$S_i^z \rightarrow -S_i^z$$

Updates satisfy detailed balance:

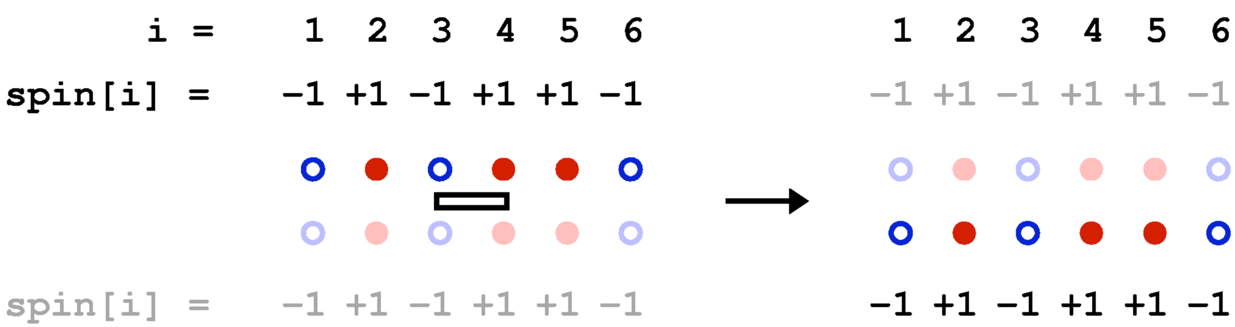
$$P_{\mathrm{accept}}(A \rightarrow B) = \min\left(\frac{W(B)P_{\mathrm{select}}(B \rightarrow A)}{W(A)P_{\mathrm{select}}(A \rightarrow B)}, 1\right)$$

Diagonal update

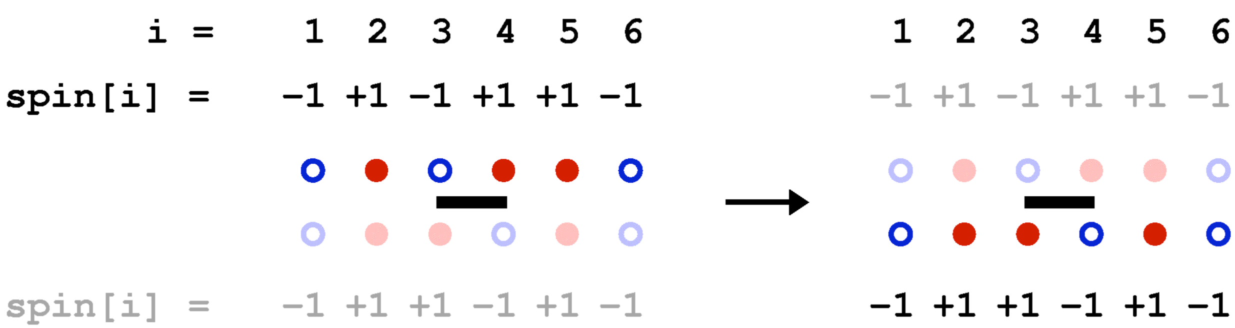
- Carried out in `opstring[p]` for p=1,...,M
- State $|\alpha(p-1)\rangle$ stored in `spin[]`



insertion of diagonal operator



removal of diagonal operator



off-diagonal no change, **propagate** state

Insertion of a diagonal operator if `opstring[p]=0`

Generate bond index b at random, attempt `opstring[p]=2*b`

- can only be done if `spin[i(b)]≠spin[j(b)]`
- n increases by 1; weight ratio

$$\frac{W(n+1)}{W(n)} = \frac{\beta/2}{M-n}$$

Removal of a diagonal operator if `opstring[p]≠0`

- n decreases by 1; weight ratio

$$\frac{W(n-1)}{W(n)} = \frac{M-n+1}{\beta/2}$$

B ways of selecting b but only one way of removing an operator;

$$\frac{P_{\text{select}}(b \to 0)}{P_{\text{select}}(0 \to b)} = B$$

Accept probabilities:

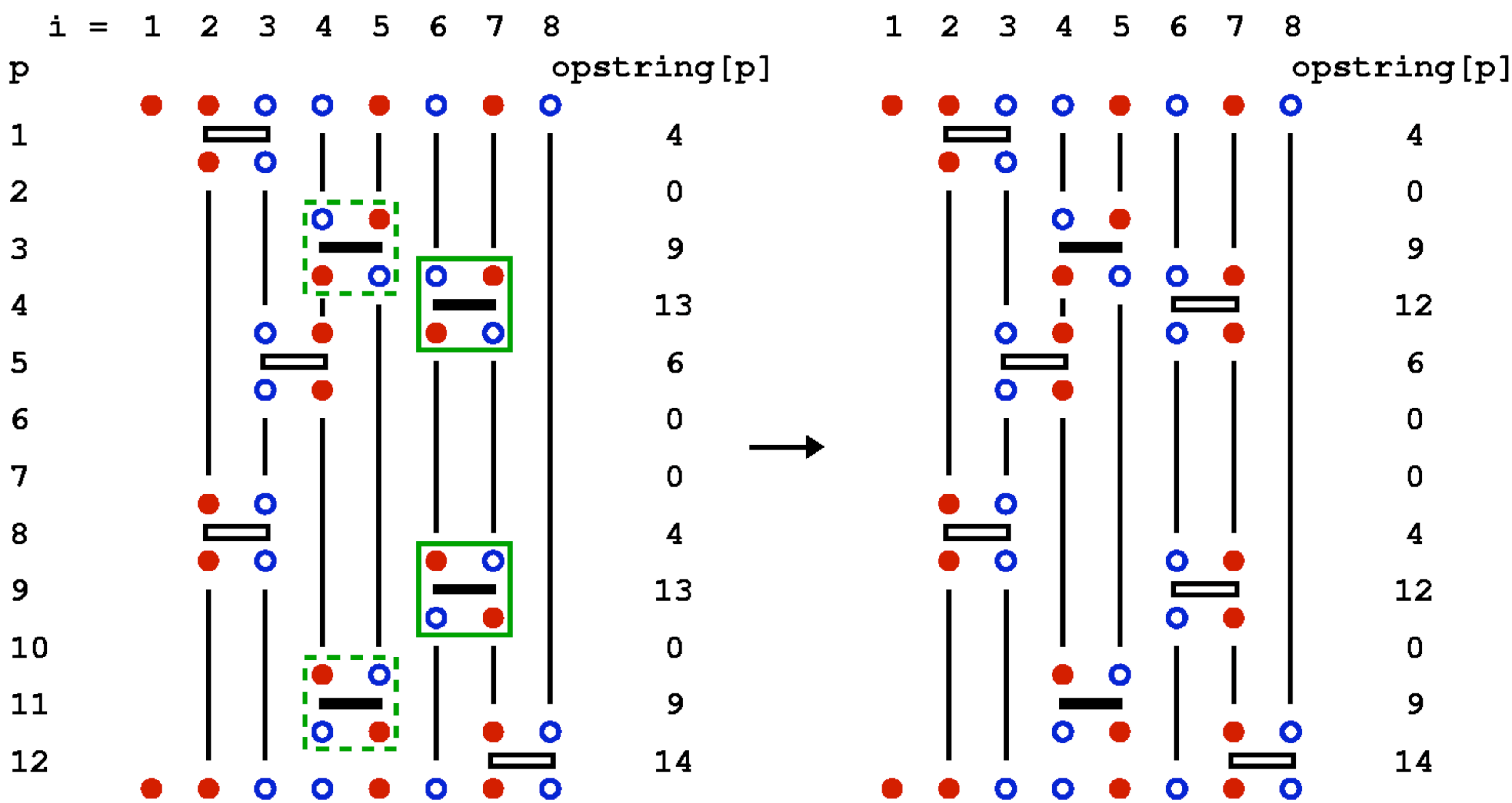
$$P_{\text{accept}}(n \to n+1) = \min\left(\frac{B\beta/2}{M-n}, 1\right)$$

$$P_{\text{accept}}(n \to n-1) = \min\left(\frac{M-n+1}{B\beta/2}, 1\right)$$

Local off-diagonal update (obsolete)

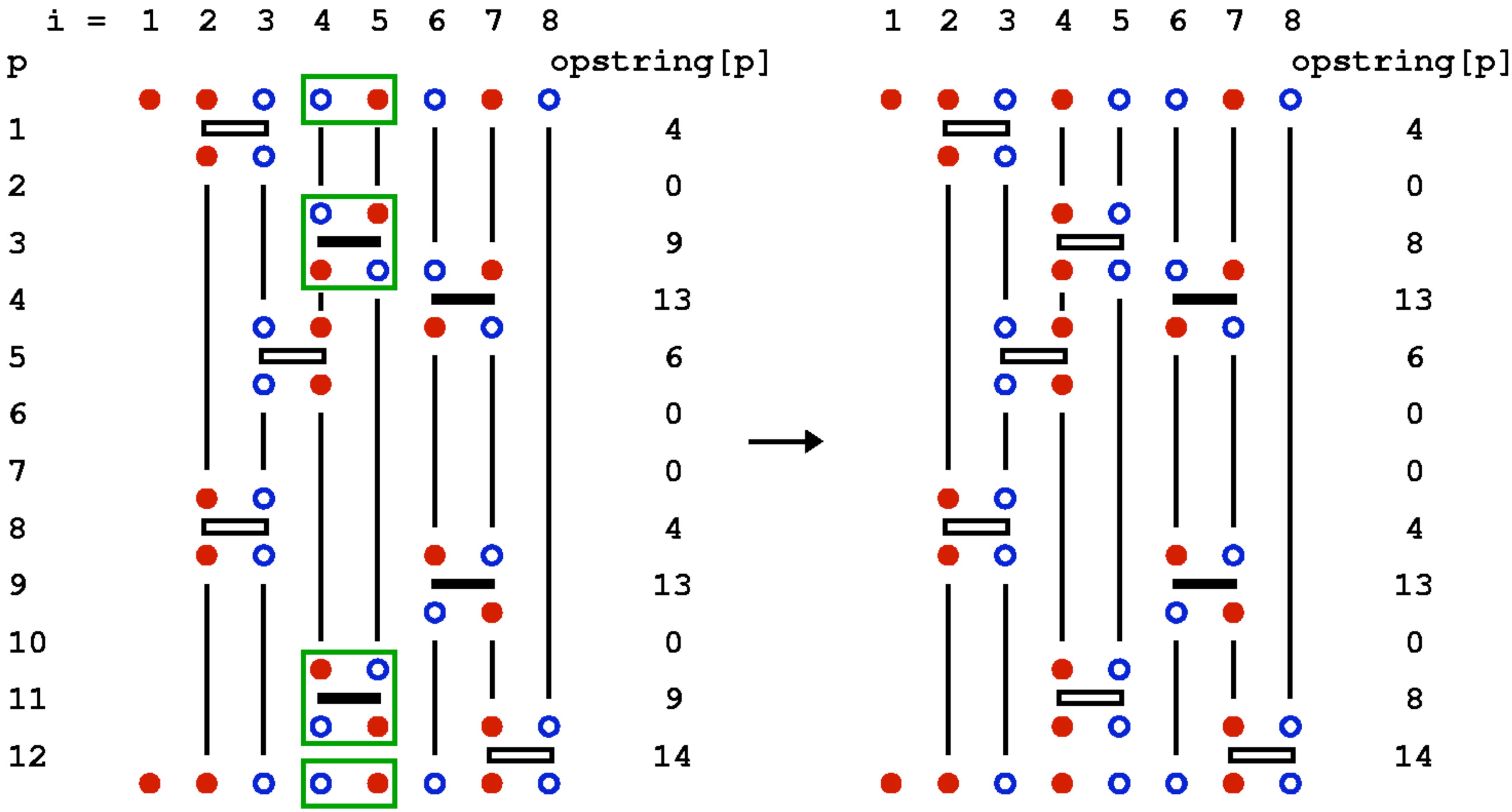
Change type of **2 operators on the same bond**

- cannot always be done; check for constraining operators
- no weight change; accept with fixed probability (e.g., P=1)



Note: periodic boundary conditions in the "propagation" direction

- update spanning across the boundary affects the stored state **|□>**

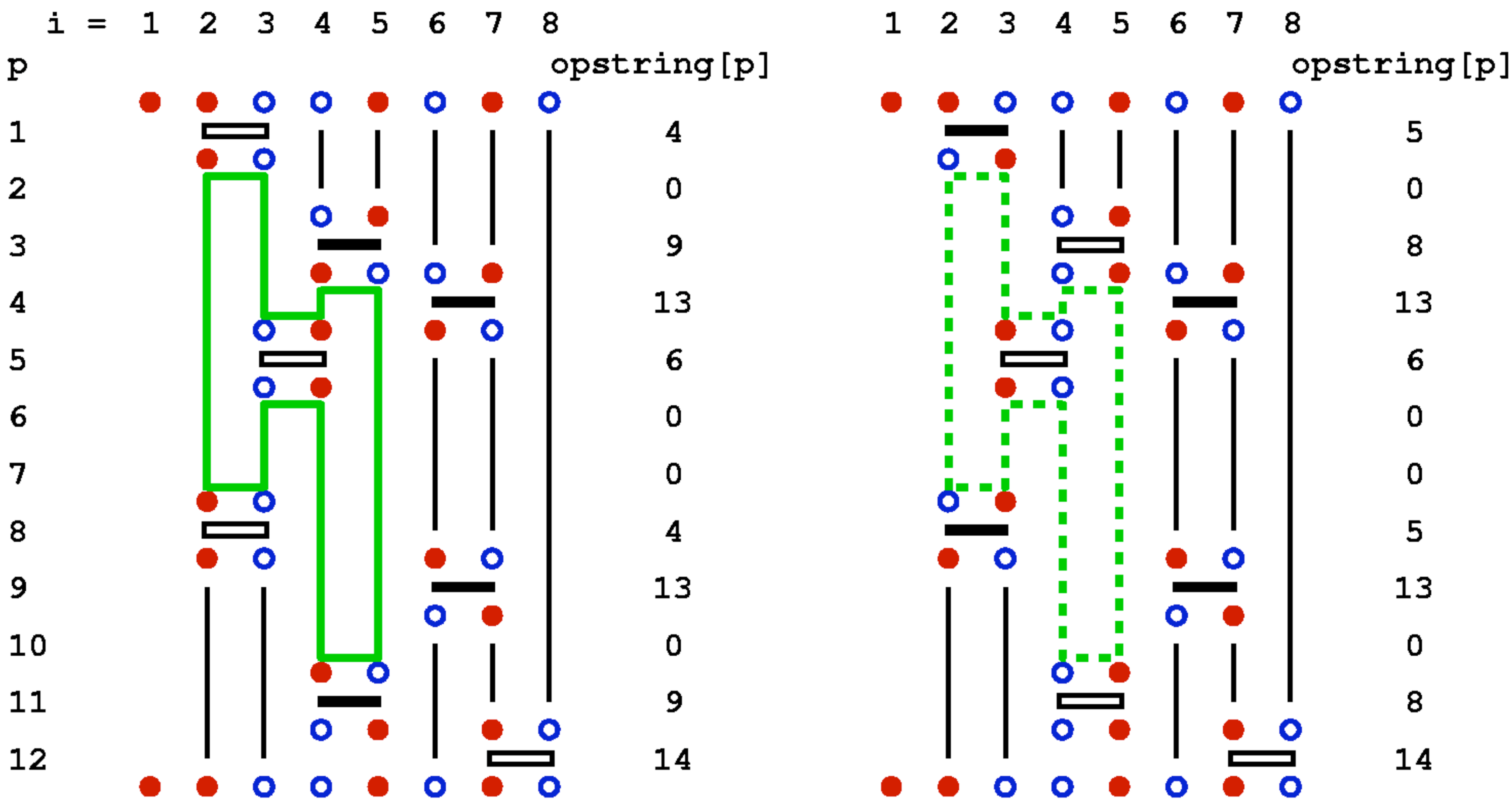


Local updates typically are **not very efficient**

- critical slowing-down
- no winding-number or particle-number fluctuations

Loop update

- carried out in the linked-vertex-list representation
- move “vertically” along links and “horizontally” on the same operator
- spins flipped at all vertex-legs visited; operator type changes; weight unchanged
- construct all loops, flip with probability 1/2 (as in Swendsen-Wang)



Monte Carlo step

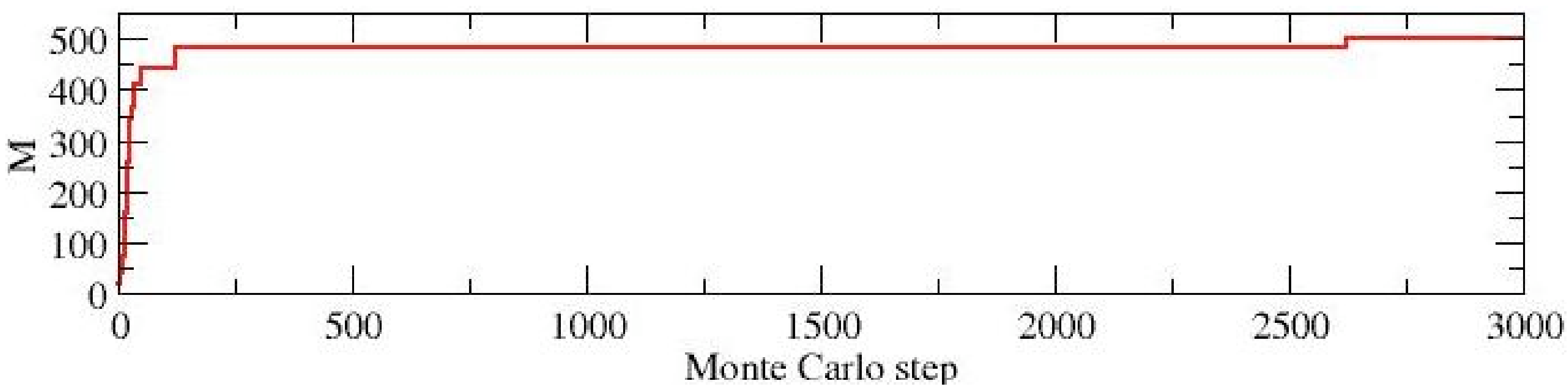
- a cycle of diagonal updates (`p=1,...,M in opstring[p]`)
- construction of the linked vertex list
- construct all loops, flip each with probability 1/2
- map updated vertex list back to `opstring[], spin[]`

Starting the simulation

- "empty" perator string, `opstring[p]=0, p=1,...,M`
- `M` is arbitrary, e.g., `M=20`
- random spin state; `spin[p]=+1,-1`

Determining the cut-off M

- after each, MC step, compare expansion order n with M
- **if M-n<n/a**, with, e.g., a=3, then **M=n+n/a**



Generalization of loop update; directed loops

In the case of the **isotropic S=1/2 model**

- There are only 4 non-0 vertices
- The operators uniquely define all loops
- Loops are non-self-intersecting

Directed loops

- In general, there are more than 4 allowed vertices
- A vertex is entered at some entrance leg
- The path can proceed (exit) through any of the 4 legs
- Exit probabilities are obtained from directed-loop equations
- Loops can back-track ("bounce") and self-intersect
- Bounces can be avoided for some models (more efficient)