Algorithms in Lattice Gauge Theory and Spin Systems
IACS, Kolkata, Jan 27 - Feb 1, 2020

Stochastic Series Expansion and Stochastic Analytic Continuation

Anders W Sandvik
Boston University and
Institute of Physics, Chinese Academy of Sciences, Beijing

References:

- AIP Conf. Proc. **1297**,135 (2010); arXiv:1101.3281
- arXiv:1909.10591(book chapter on SSE and related)
- PRB **82**, 024407 (2010) (valence-bond projector QMC)
- PRE **68**, 056701 (2003) (transverse-field Ising)
- PRX **7**, 041072 (2017) (dynamics)

NSF
SIMONS FOUNDATION
BOSTON UNIVERSITY
Institute of Physics CAS

Lecture plan

Lecture I

Stochastic Series Expansion and ground-state projection with valence bonds

- SSE: Basic idea and implementation for S=1/2 Heisenberg model
- The valence-bond basis for S=1/2 systems
- Projection of the ground state from an “amplitude product state”

Lecture II

Dynamics; spectral functions and the numerical analytic continuation problem

- Stochastic Analytic Continuation

Lecture III

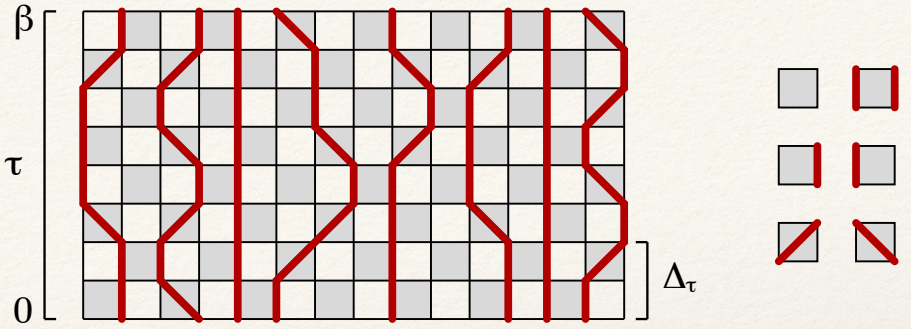
Applications

- Quick review of finite-size scaling
- Examples of conventional quantum phase transitions
- J-Q models
  - valence-bond-solid states
  - deconfined quantum criticality
  - emergent symmetries

QMC algorithms for quantum spins (and bosons)

From operators to numbers

$$\langle A \rangle = \frac{\mathrm{Tr}\{A e^{-\beta H}\}}{\mathrm{Tr}\{e^{-\beta H}\}} \rightarrow \frac{\sum_c A_c W_c}{\sum W_c}$$



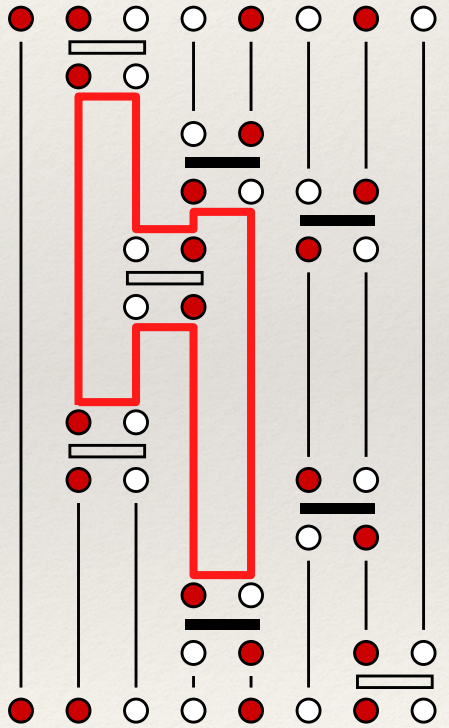
"sign problem" if W_c not positive-definite
- consider sign-free models

- **Trotter slicing;** discrete imaginary time; world line methods (Suzuki 1971,...)

- **Taylor expansion;** stochastic series expansion (SSE) (Handscomb -61,… Sandvik, Kurkijärvi -91,…)

- **Continuous time;** take $\Delta_\tau \rightarrow 0$ limit before programming (Beard, Wiese, -96, Prokof'ev et al. -96,…)

- From local updates to **loops, worms, directed loops....** (Evertz et al. -93, Beard, Wiese, -96, Prokof'ev et al. -96, Sandvik -99, Sandvik, Syljuåsen -02)



Related: ground-state projection

$$|\Psi_\beta\rangle \sim \mathrm{e}^{-\beta H}|\Psi_0\rangle \qquad |\Psi_\beta\rangle \rightarrow |0\rangle \;\; \text{when} \;\; \beta \rightarrow \infty$$

- Differs only in time boundary condition (open vs periodic)

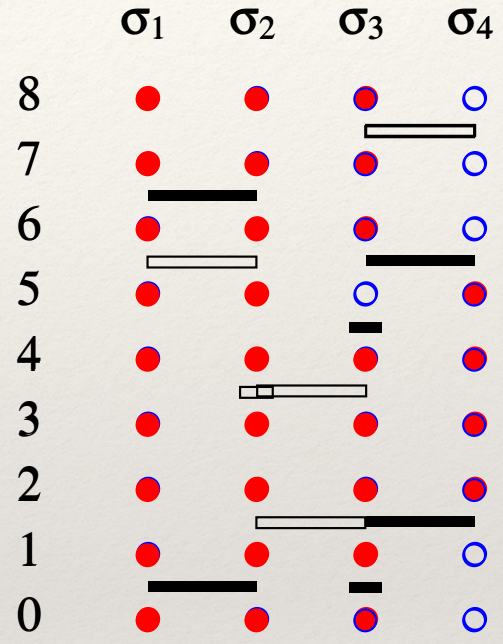
Series expansion representation of quantum stat mech

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)

Path weight: $W(S_n, \alpha_0) = \frac{(-\beta)^n}{n!} \prod_{p=1}^{n} \langle \alpha_p | H_{a_p} | \alpha_{p-1} \rangle$

Easy to calculate

- use as MC sampling weight

Expectation values

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

Simplest case: Operator A **diagonal** in the chosen basis:

$$\langle A\rangle = \sum_{n=0}^{\infty}\sum_{\alpha_0}\sum_{S_n} W(S_n,\alpha_0)A(\alpha_0) \qquad A(\alpha_0)\to\frac{1}{n}\sum_{p=0}^{n-1}A(\alpha_p)$$

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

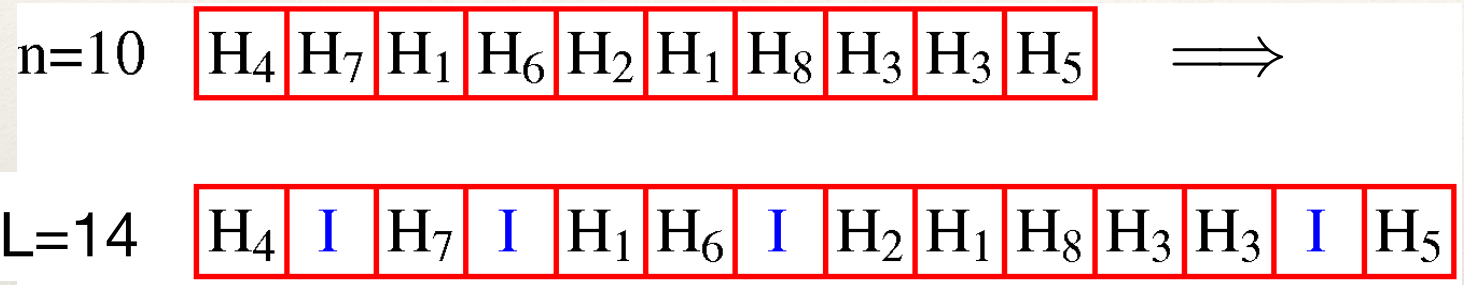
Therefore the energy is: $E = -\langle n\rangle/\beta$

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**



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

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

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)

Relation to the expansion in interaction representation

For ***H=D+V***, diagonal *D*, off-diagonal *V*

Beard, Wiese (1996)
Prokovev, Svistunov, Tupitsyn (1996)
Sandvik, Singh, Campbell (1997)

$$Z=\sum_{n=0}^{\infty}(-1)^n\int_0^{\beta}d\tau_1\int_0^{\tau_1}d\tau_2\cdots\int_0^{\tau_{n-1}}d\tau_n\ \mathrm{Tr}\{e^{-\beta\hat{D}}\hat{V}(\tau_1)V(\tau_2)\cdots V(\tau_n)\}$$

Proceed as in SSE, only off-diagonal operators in diagrams

$$Z=\sum_{\alpha}\sum_{n=0}^{\infty}\sum_{T_n}\int_0^{\beta}d\tau_1\int_0^{\tau_1}d\tau_2\cdots\int_0^{\tau_{n-1}}d\tau_n W(\alpha,T_n,\{\tau\})$$

What is better, SSE or interaction rep?

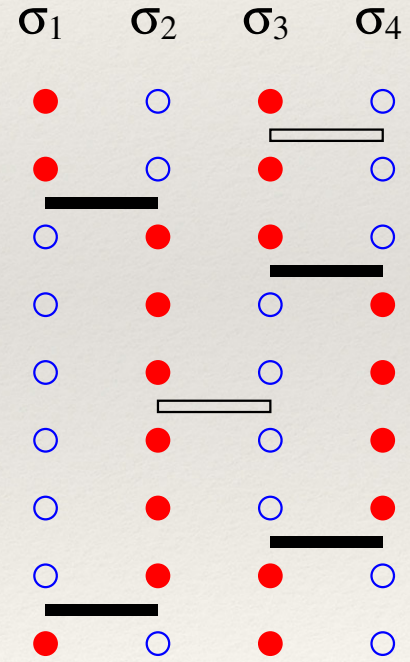
- depends on balance of diagonal and off-diagonal energy
- Interaction rep better if diagonal energy dominates
- SSE often better if that is not the case

Extreme case: Only off-diagonal operators

- for example, XY model in z basis

Time integrals in interaction rep give $\beta^n/n!$

- configurations identical to SSE
- SSE avoids time integrals

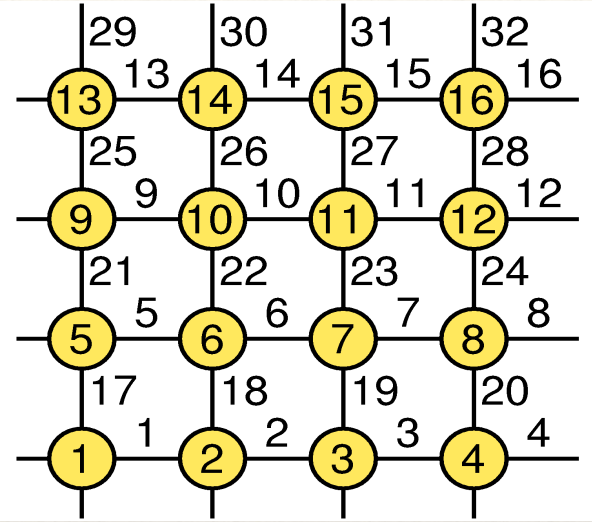


Stochastic Series expansion (SSE): S=1/2 Heisenberg model

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

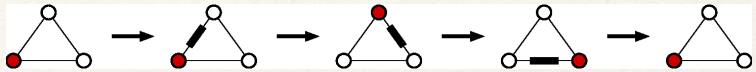
Index sequence: $S_n = [a(0), b(0)], [a(1), b(1)], \ldots, [a(n-1), b(n-1)]$

For fixed-length scheme

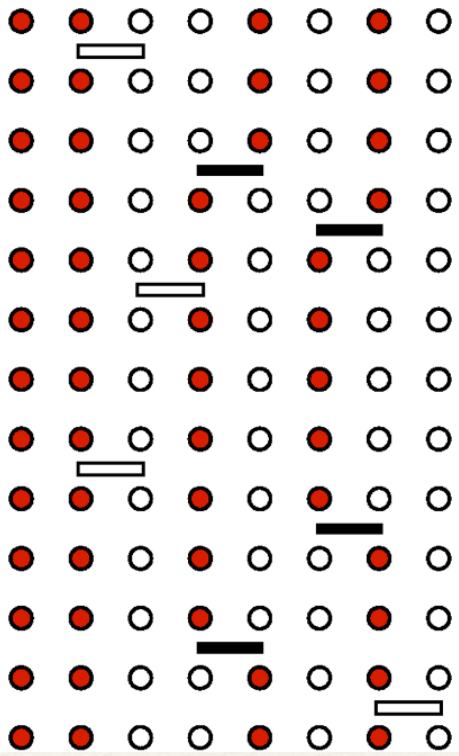
$$Z = \sum_{\alpha} \sum_{S_L} (-1)^{n_2} \frac{\beta^n (L-n)!}{L!} \left\langle \alpha \left| \prod_{p=0}^{L-1} H_{a(p),b(p)} \right| \alpha \right\rangle \qquad W(\alpha, S_L) = \left(\frac{\beta}{2}\right)^n \frac{(L-n)!}{L!}$$

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

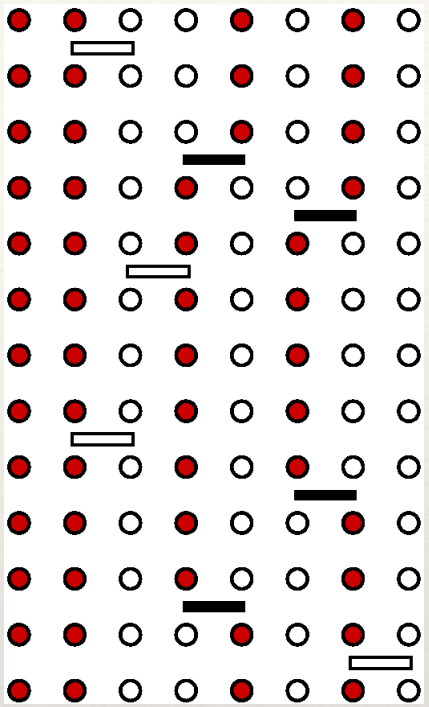
SSE effectively provides a discrete representation of the time continuum!
- computational advantage; only integer operations in sampling

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

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

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

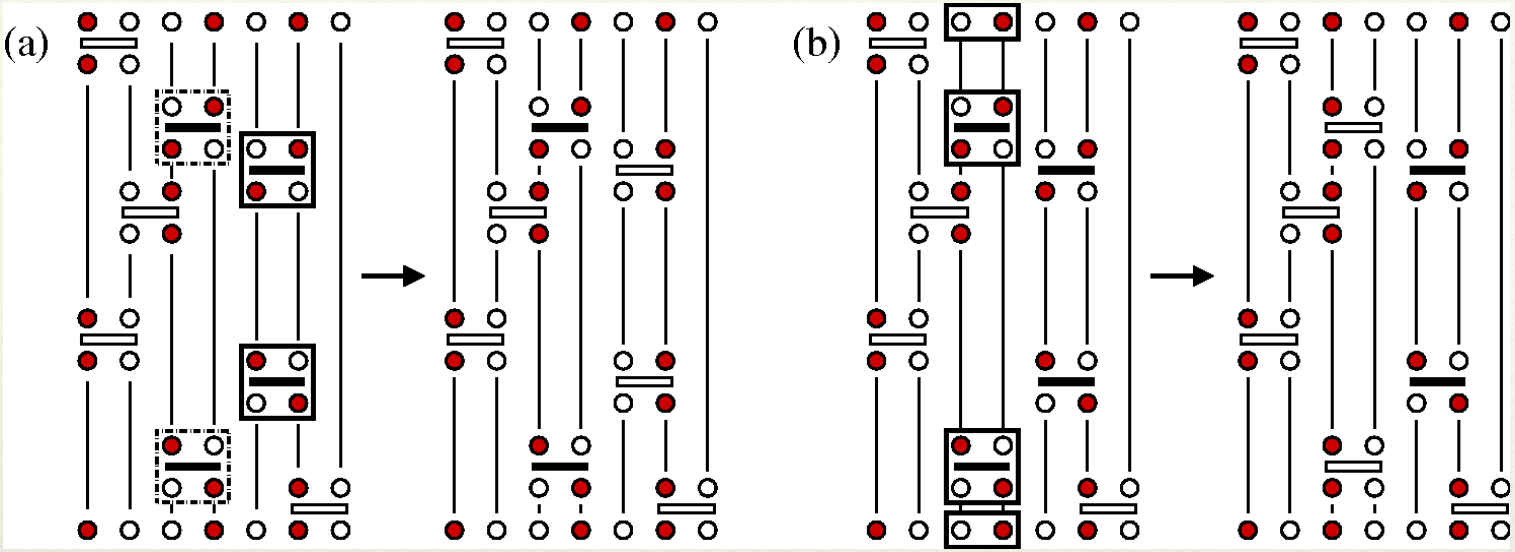
Pseudocode: Sweep of diagonal updates

```
do p = 0 to L − 1
    if (s(p) = 0) then
        b = random[1, . . . , N_b]
        if  σ(i(b)) = σ(j(b)) cycle
        if (random[0 − 1] < P_insert(n)) then s(p) = 2b; n = n + 1 endif
    elseif (mod[s(p), 2] = 0) then
        if (random[0 − 1] < P_remove(n)) then s(p) = 0; n = n − 1 endif
    else
        b = s(p)/2; σ(i(b)) = −σ(i(b)); σ(j(b)) = −σ(j(b))
    endif
enddo
```

Code explanation:

- To insert operator, bond b generated at random among 1,...,N_b
  - can be done only if connected spins i(b),j(b) are anti-parallel
  - if so, do it with probability $P_{insert}(n)$
- Existing diagonal operator can always be removed
  - do it with probability $P_{remove}(n)$
- If off-diagonal operator, advance the state
  - extract bond b, flip spins at i(b),j(b)

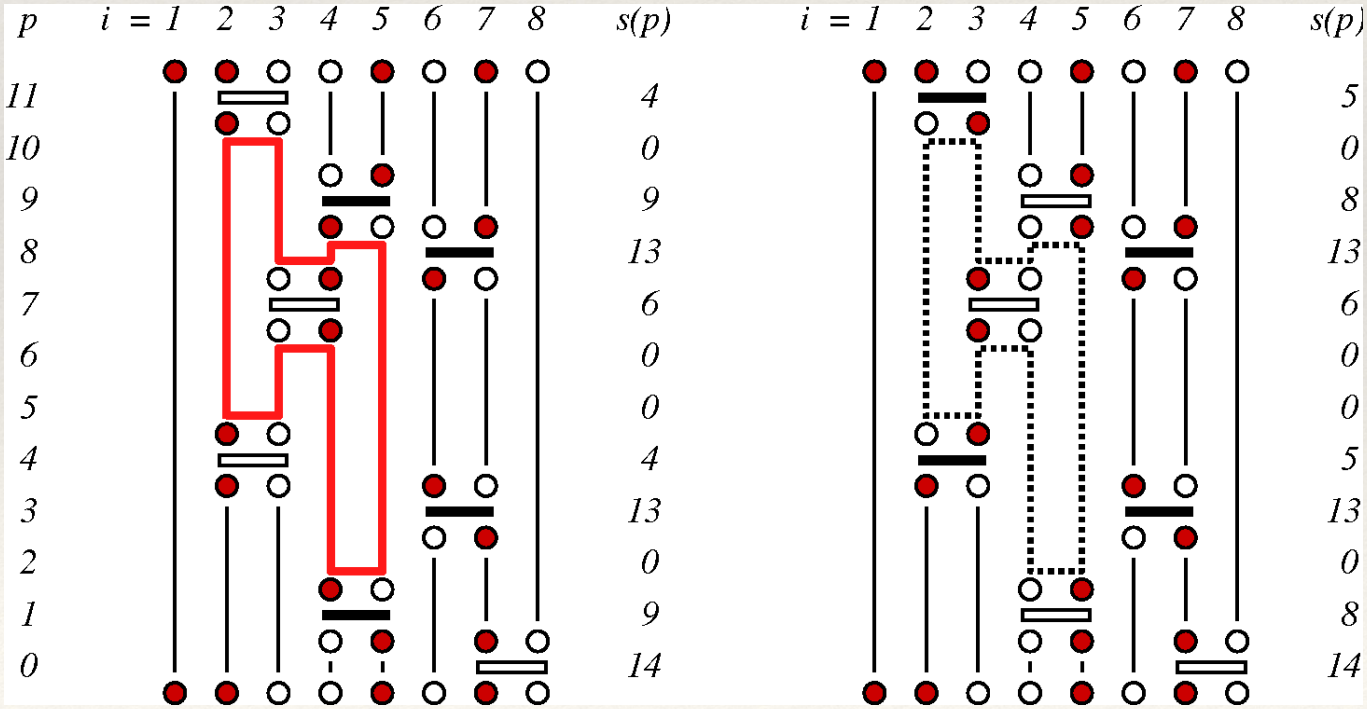
Off-diagonal updates



Local update

Change the type of two operators

- constraints
- inefficient
- cannot change winding numbers

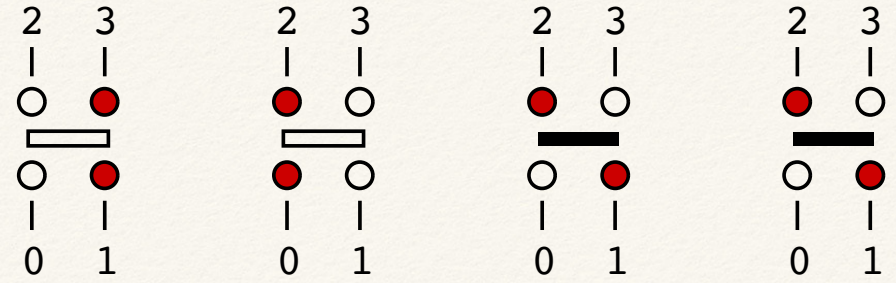


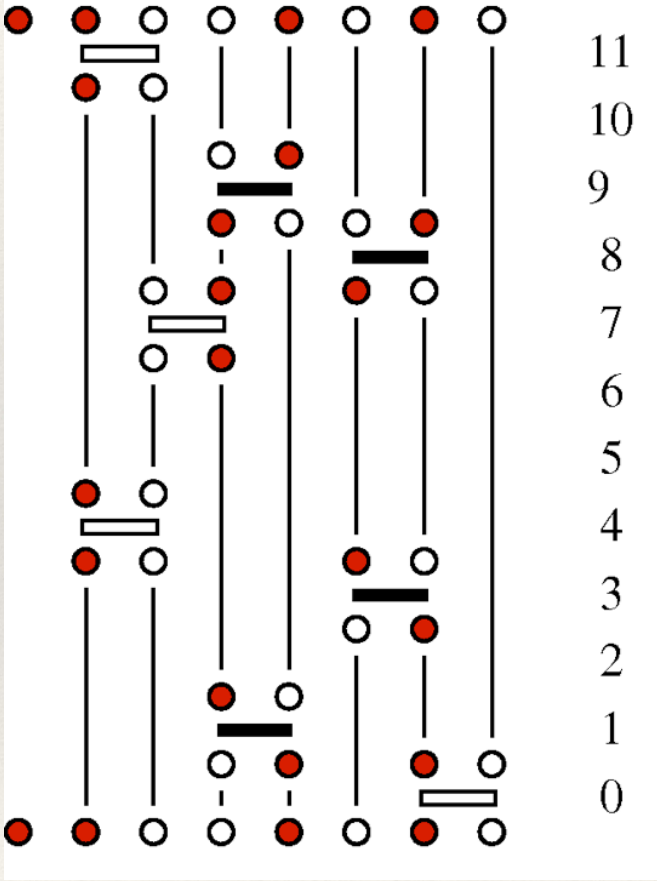
Operator-loop update

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

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

Pseudocode: Sweep of loop updates

constructing all loops, flip probability 1/2

```
do v_0 = 0 to 4L − 1 step 2
    if (X(v_0) < 0) cycle
    v = v_0
    if (random[0 − 1] < 1/2) then
        traverse the loop; for all v in loop, set X(v) = −1
    else
        traverse the loop; for all v in loop, set X(v) = −2
        flip the operators in the loop
    endif
enddo
```

- visited vertices are no longer needed and we set them to a negative value -1 or -2, to indicate that the loop has been visited (-1) or visited and flipped (-2)

construct and flip a loop

```
v = v_0
do
    X(v) = −2
    p = v/4; s(p) = flipbit(s(p), 0)
    v' = flipbit(v, 0)
    v = X(v'); X(v') = −2
    if (v = v_0) exit
enddo
```

- p is the location of the operator in the original length-L list of operatotors
- by flipping bit 0 of s(p), the operator changes from diagonal to off-diagonal, or vice versa
- moving on the vertex to the adjacent spin is also done with a bit flip

We also have to modify the stored spin state after the loop update

- we can use the information in $V_{first}()$ and X() to determine spins to be flipped
- spins with no operators, $V_{first}(i)=-1$, flipped with probability 1/2

```
do i = 1 to N
     v = V_first(i)
     if (v = −1) then
          if (random[0-1]< 1/2) σ(i) = −σ(i)
     else
          if (X(v) = −2) σ(i) = −σ(i)
     endif
enddo
```

$v=V_{first}(i)$ is the location of the first vertex leg on site i

- flip the spin if X(v)=–2
- (do not flip it if X(v)=–1)
- no operation on i if $v_{first}(i)=-1$; then it is flipped with probability 1/2

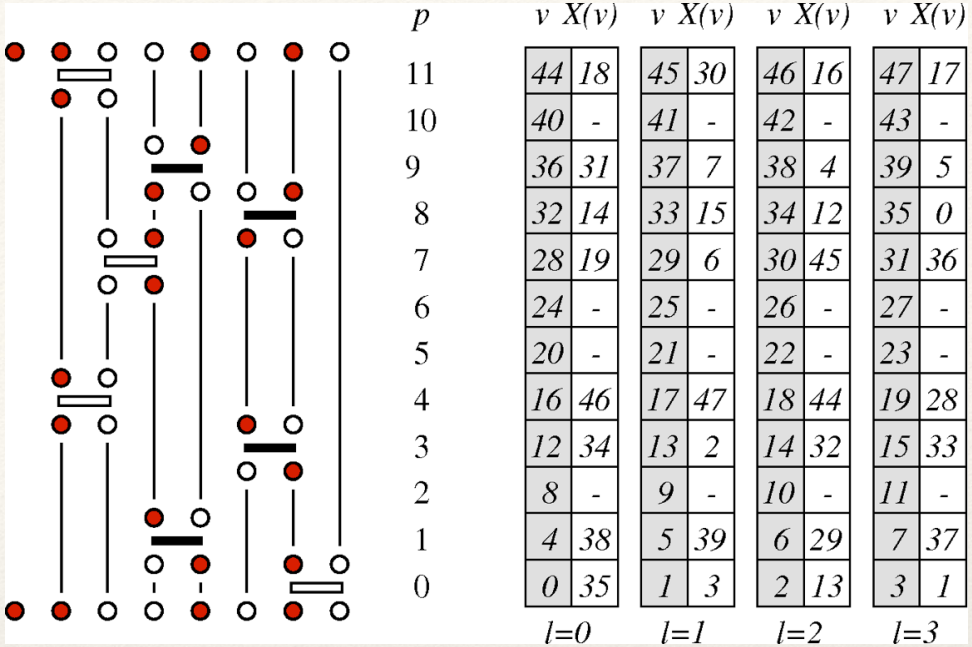
Constructing the linked vertex list

Traverse operator list *s(p)*, *p*=0,...,*L*–1

- vertex legs **v=4p,4p+1,4p+2,4p+3**

Use arrays to keep track of the first and last (previous) vertex leg on a given spin

- $V_{first}(i)$ = location v of first leg on site i
- $V_{last}(i)$ = location v of last (currently) leg
- these are used to create the links
- initialize all elements to –1



$V_{\rm first}(:) = -1;\ V_{\rm last}(:) = -1$
do $p = 0$ **to** $L-1$
  if $(s(p) = 0)$ **cycle**
  $v_0 = 4p;\ b = s(p)/2;\ s_1 = i(b);\ s_2 = j(b)$
  $v_1 = V_{\rm last}(s_1);\ v_2 = V_{\rm last}(s_2)$
  if $(v_1 \neq -1)$ **then** $X(v_1) = v_0;\ X(v_0) = v_1$ **else** $V_{\rm first}(s_1) = v_0$ **endif**
  if $(v_2 \neq -1)$ **then** $X(v_2) = v_0;\ X(v_0) = v_2$ **else** $V_{\rm first}(s_2) = v_0 + 1$ **endif**
  $V_{\rm last}(s_1) = v_0 + 2;\ V_{\rm last}(s_2) = v_0 + 3$
enddo

creating the last links across the "time" boundary

do $i = 1$ **to** N
  $f = V_{\rm first}(i)$
  if $(f \neq -1)$ **then** $l = V_{\rm last}(i);\ X(f) = l;\ X(l) = f$ **endif**
enddo

Determination of the cut-off L

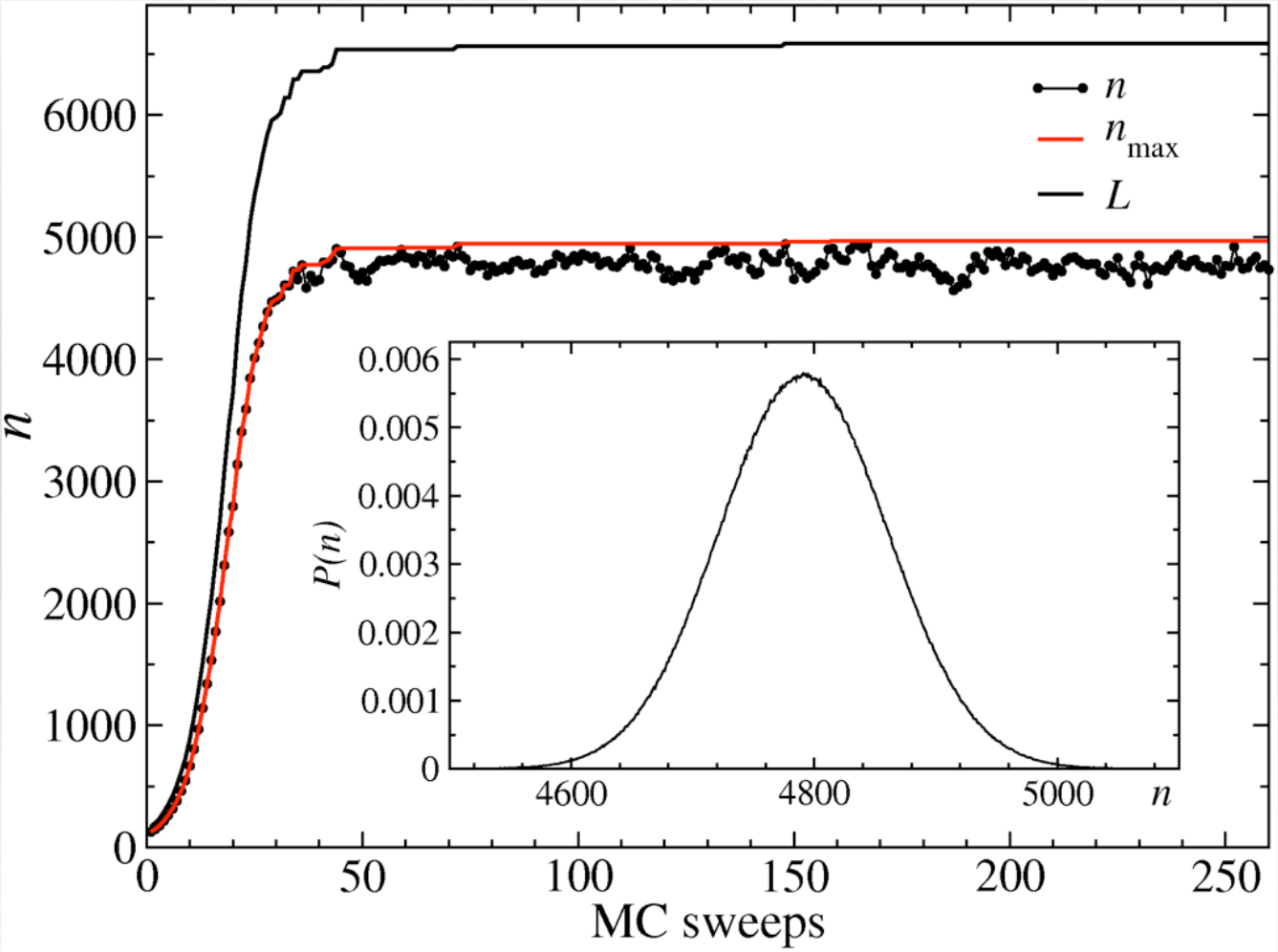
- adjust during equilibration
- start with arbitrary (small) n

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

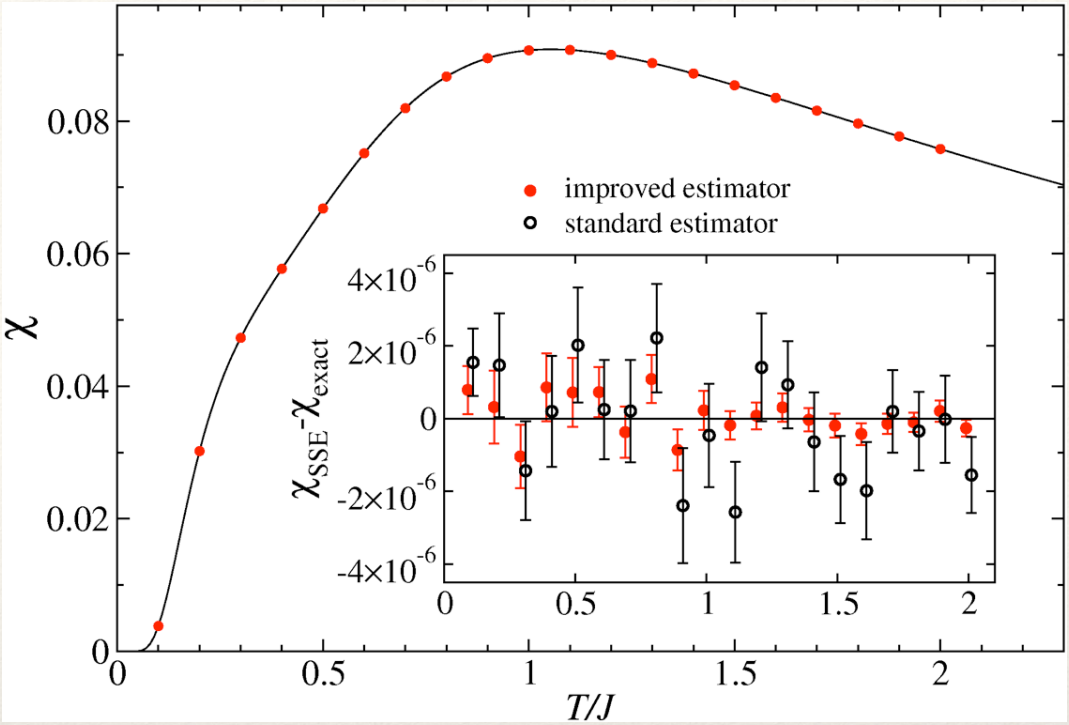


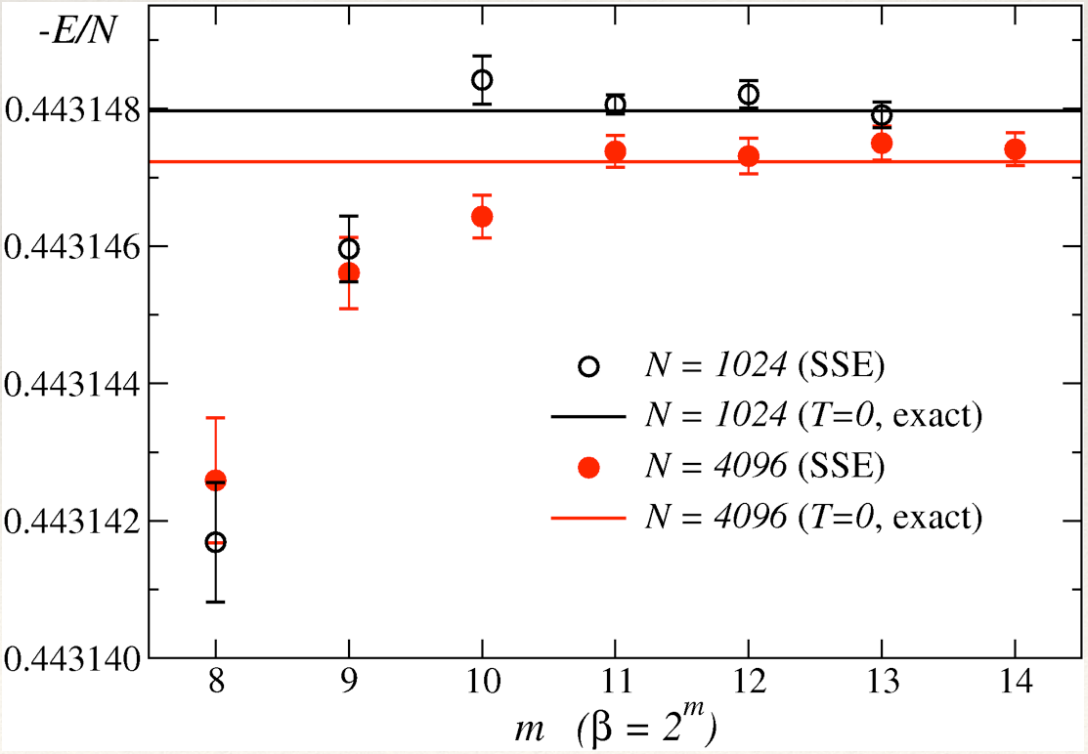
Does it work?
Compare with exact results

- 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

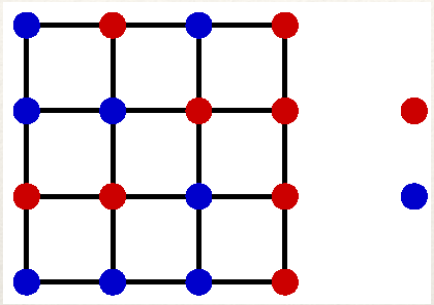
- SSE results for 10^6 sweeps
- Bethe Ansatz ground state E/N
- SSE can achieve the ground state limit (T→0)

Common bases for quantum spin systems

Lattice of S=1/2 spins, e.g., Heisenberg antiferromagnet

$$H = J \sum_{\langle i,j \rangle} \vec{S}_i \cdot \vec{S}_j = J \sum_{\langle i,j \rangle} \left[S_i^z S_j^z + (S_i^+ S_j^- + S_i^- S_j^+)/2 \right]$$

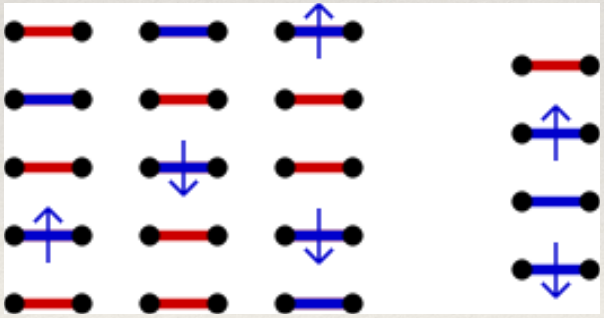
The most common basis is that of 'up' and 'down' spins



$$= |\uparrow\rangle = |S^z = +1/2\rangle$$
$$= |\downarrow\rangle = |S^z = -1/2\rangle$$

One can also use eigenstates of two or more spins

- dimer singlet-triplet basis



$$= (|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle)/\sqrt{2}$$
$$= |\uparrow\uparrow\rangle$$
$$= (|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle)/\sqrt{2}$$
$$= |\downarrow\downarrow\rangle$$

The Hamiltonian is more complicated in this basis

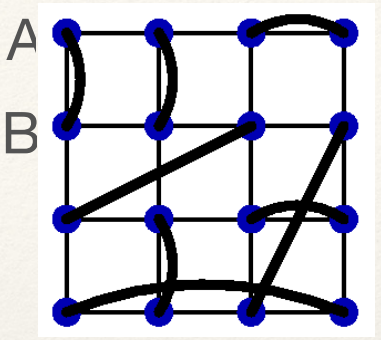
- but some times can be used to solve sign problems
  Alet, Damle, Pujari, PRL 2016; Honecker et al., PRB 2016

The valence bond basis for S=1/2 spins

Valence-bonds between sublattice A, B sites $(i,j) = (|\uparrow_i\downarrow_j\rangle - |\downarrow_i\uparrow_j\rangle)/\sqrt{2}$

Basis states; singlet products

$$|V_r\rangle = \prod_{b=1}^{N/2}(i_{rb}, j_{rb}), \quad r = 1, \ldots (N/2)!$$



The valence bond basis is overcomplete and non-orthogonal

- expansion of arbitrary singlet state, not unique

$$|\Psi\rangle = \sum_r f_r |V_r\rangle$$ all f_r positive for non-frustrated system (Marshall signs)

All valence bond states overlap with each other

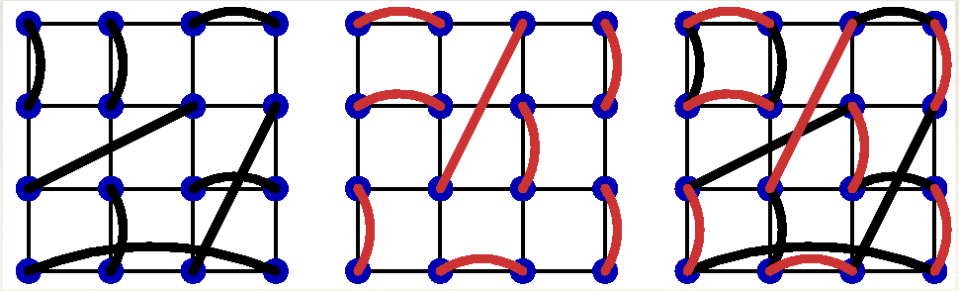
$$\langle V_l|V_r\rangle = 2^{N_\circ - N/2} \qquad N_\circ = \text{ number of loops in overlap graph}$$

Spin correlations from loop structure

$$\frac{\langle V_l|\vec{S}_i\cdot\vec{S}_j|V_r\rangle}{\langle V_l|V_r\rangle} = \begin{cases} \frac{3}{4}(-1)^{x_i - x_j + y_i - y_j} & \textit{(i,j in same loop)} \\ 0 & \textit{(i,j in different loops)} \end{cases}$$

More complicated matrix elements (e.g., dimer correlations) are also related to the loop structure

K.S.D. Beach and A.W.S.,
Nucl. Phys. B 750, 142 (2006)



Projector Monte Carlo in the valence-bond basis

Liang, 1991; AWS, Phys. Rev. Lett 95, 207203 (2005)

(-H)^n projects out the ground state from an arbitrary state

$$(-H)^n|\Psi\rangle = (-H)^n \sum_i c_i|i\rangle \rightarrow c_0(-E_0)^n|0\rangle$$

S=1/2 Heisenberg model

$$H = \sum_{\langle i,j\rangle} \vec{S}_i \cdot \vec{S}_j = -\sum_{\langle i,j\rangle} H_{ij}, \qquad H_{ij} = (\tfrac{1}{4} - \vec{S}_i \cdot \vec{S}_j)$$

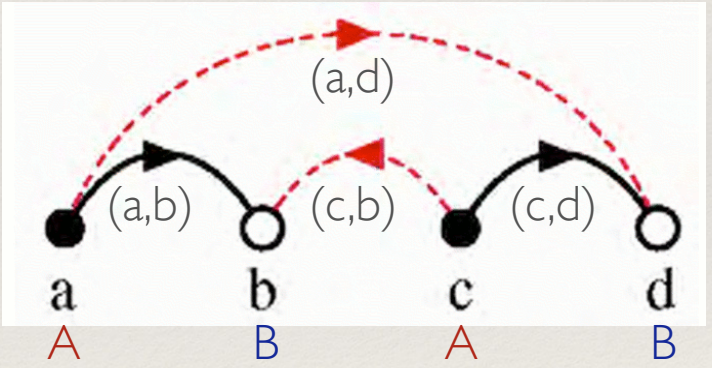
Project with string of bond operators

$$\sum_{\{H_{ij}\}} \prod_{p=1}^{n} H_{i(p)j(p)}|\Psi\rangle \rightarrow r|0\rangle$$ (r = normalization)

Action of bond operators

$$H_{ab}|...(a,b)...(c,d)...\rangle = |...(a,b)...(c,d)...\rangle$$

$$H_{bc}|...(a,b)...(c,d)...\rangle = \frac{1}{2}|...(c,b)...(a,d)...\rangle$$



$$(i,j) = (|\uparrow_i\downarrow_j\rangle - |\downarrow_i\uparrow_j\rangle)/\sqrt{2}$$

Simple reconfiguration of bonds (or no change; diagonal)

- no minus signs for A→B bond 'direction' convention
- sign problem does appear for frustrated systems

Expectation values: $\langle A \rangle = \langle 0|A|0\rangle$

Strings of singlet projectors

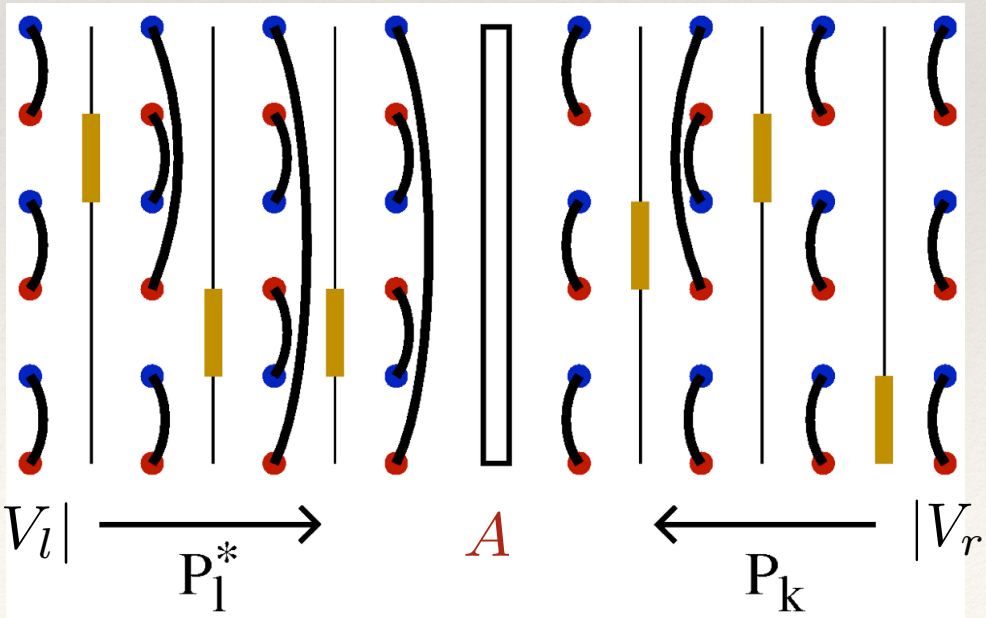
$$P_k = \prod_{p=1}^{n} H_{i_k(p) j_k(p)}, \quad k = 1, \ldots, N_b^n \quad (N_b = \text{number of interaction bonds})$$

We have to project bra and ket states

$$\sum_k P_k |V_r\rangle = \sum_k W_{kr} |V_r(k)\rangle \to (-E_0)^n c_0 |0\rangle$$

$$\sum_g \langle V_l| P_g^* = \sum_g \langle V_l(g)| W_{gl} \to \langle 0| c_0 (-E_0)^n$$

6-spin chain example:



$$\langle A \rangle = \frac{\sum_{g,k} \langle V_l| P_g^* A P_k |V_r\rangle}{\sum_{g,k} \langle V_l| P_g^* P_k |V_r\rangle} = \frac{\sum_{g,k} W_{gl} W_{kr} \langle V_l(g)|A|V_r(k)\rangle}{\sum_{g,k} W_{gl} W_{kr} \langle V_l(g)|V_r(k)\rangle}$$

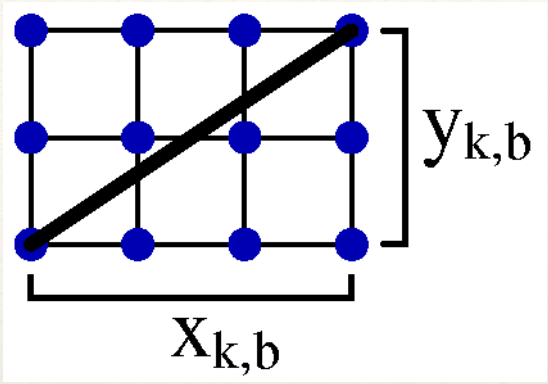
- Monte Carlo sampling of operator strings
- Estimators based on transition graphs

Sampling an amplitude-product state

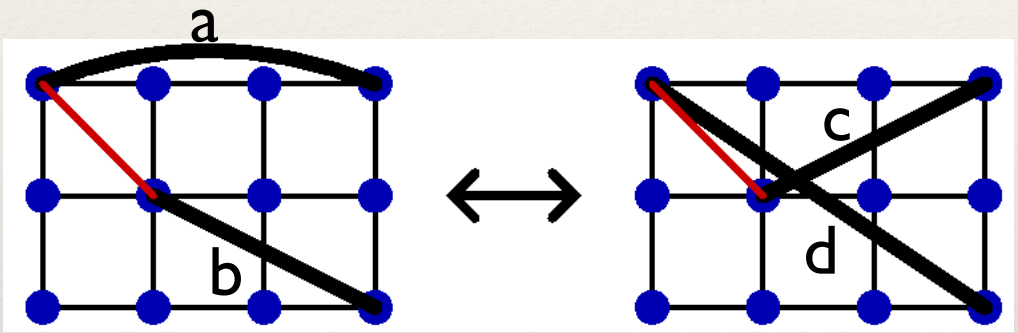
A better trial state leads to faster n convergence

- bond-amplitude product state [Liang, Doucot, Anderson, 1990]

$$|\Psi_0\rangle = \sum_k \prod_{b=1}^{N/2} h(x_{rb}, y_{rb})|V_k\rangle$$



Update state by reconfiguring two bonds



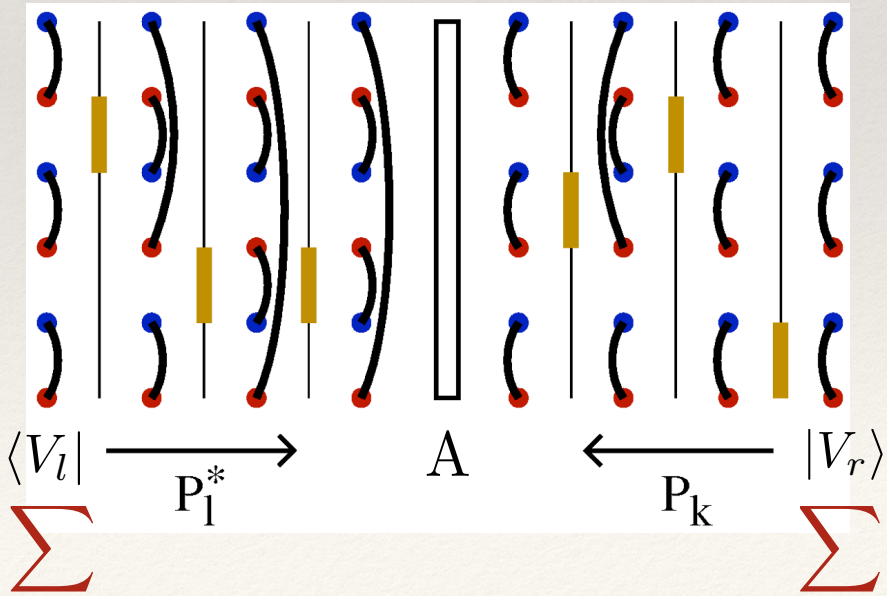
$$P_{\text{accept}} = \frac{h(x_c, y_c)h(x_d, y_d)}{h(x_a, y_a)h(x_b, y_b)}$$

If reconfiguration accepted

- calculate change in projection weight
- used for final accept/reject prob.

S. Liang [PRB 42, 6555 (1990)]

- used parametrized state amplitudes
- determined parameters variationally
- improved state by projection

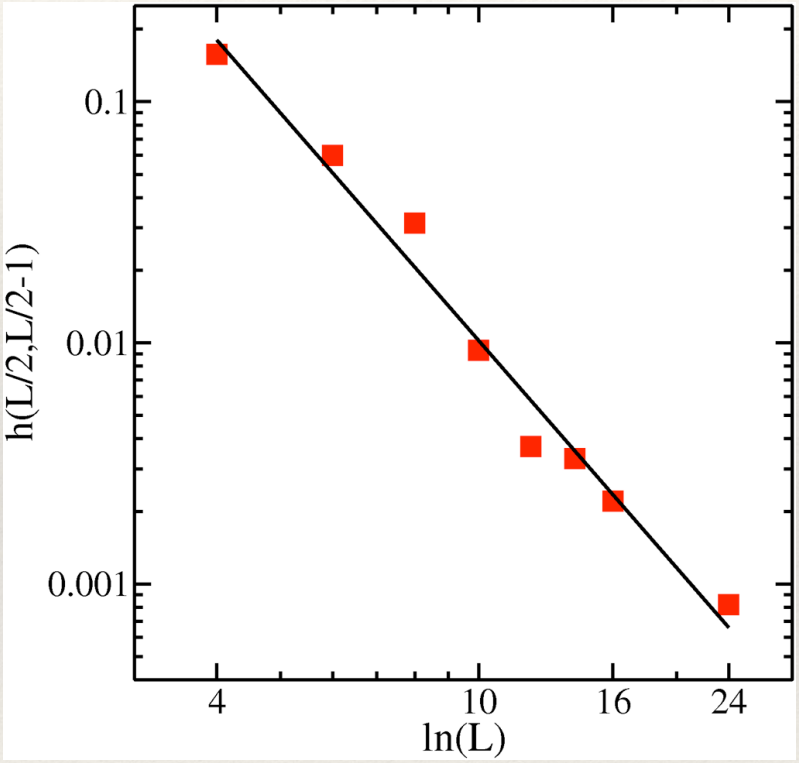


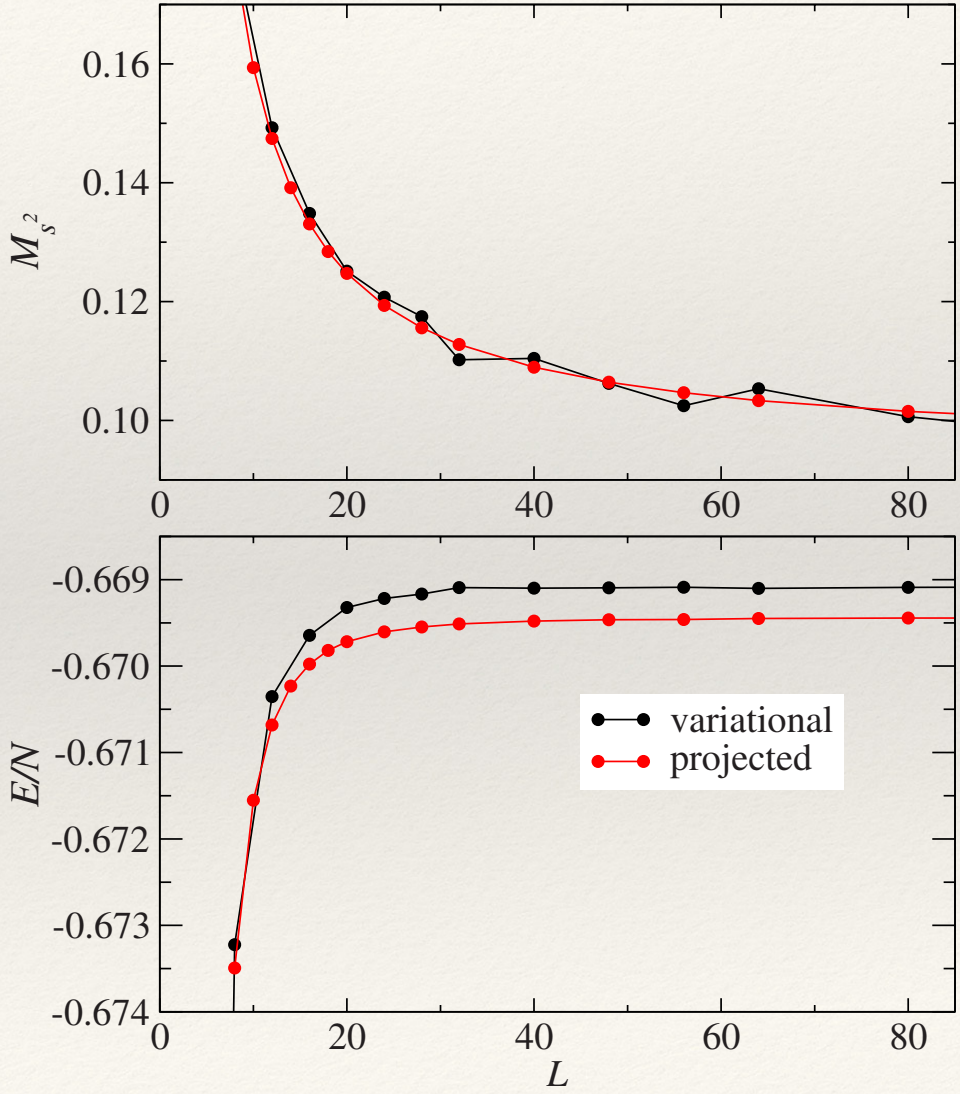
Variational wave function (2D Heisenberg)

All amplitudes h(x,y) can be optimized

[J. Lou and A.W.S., PRB 2007, AWS and H.-G. Evertz, PRB 2010]

- variational energy error 50% smaller than previously best (<0.1%)
- spin correlations deviate by less than 1% from exact values
- amplitudes decay as ~$1/r^3$





Variational energy can be further improved by including optimized bond correlations; Lin et al. PRB 2012 (posted on course web site)

More efficient ground state QMC algorithm → larger lattices

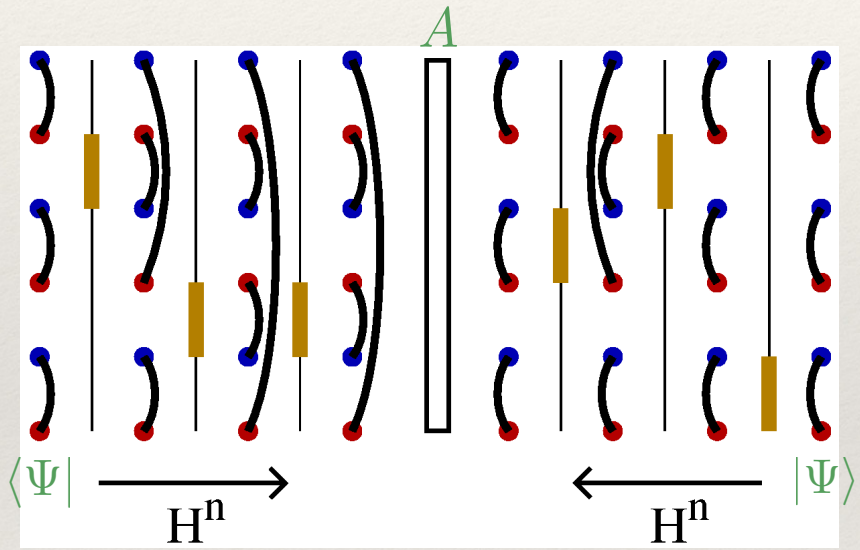
Loop updates in the valence-bond basis

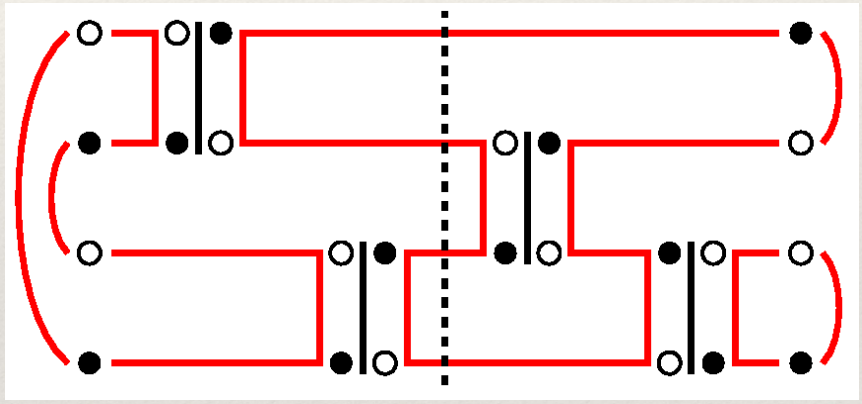
AWS and H. G. Evertz, PRB 2010

Put the spins back in a way compatible with the valence bonds

$$(a_i, b_i) = (\uparrow_i\downarrow_j - \downarrow_i\uparrow_j)/\sqrt{2}$$

and sample in a combined space of spins and bonds





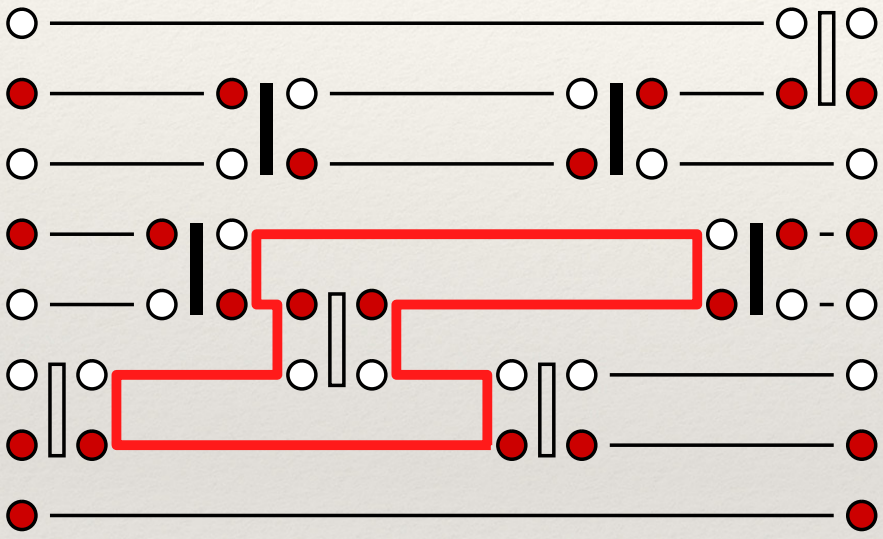
Loop updates similar to those in finite-T methods
(world-line and stochastic series expansion methods)

- good valence-bond trial wave functions can be used
- larger systems accessible
- sample spins, but measure using the valence bonds

T>0 and T=0 algorithms side-by-side

Finite-temperature QMC
(world lines, SSE,...)

$$\mathrm{tr}\{\mathrm{e}^{-\beta H}\} = \sum_n \frac{\beta^n}{n!} \langle\alpha|(-H)^n|\alpha\rangle$$

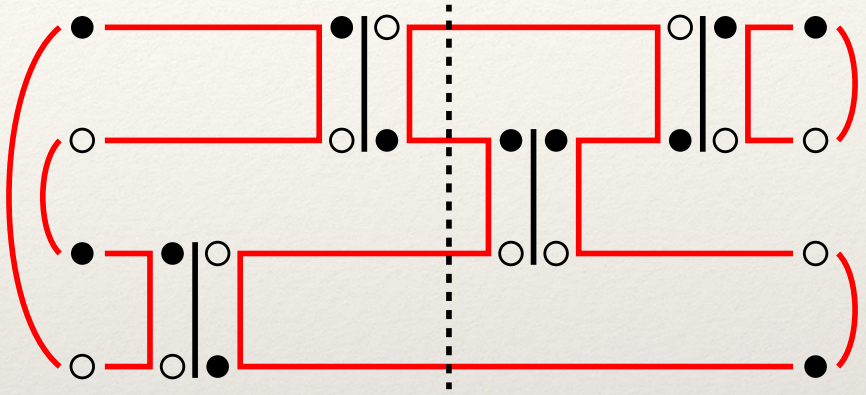


periodic time boundary conditions

- Computer implementations similar

Ground state projection

$$\sum_{\alpha\beta} f_\beta f_\alpha \langle\beta|(-H)^m|\alpha\rangle$$



open boundaries capped by valence bonds (2-spin singlets) [AWS, HG Evertz, 2010]

Trial state can conserve relevant ground state quantum numbers (S=0, k=0,...)

Convergence

Trial state expanded in H-eigenstates

$$|\psi_0\rangle = \sum_n c_n |n\rangle$$

Projected state after m-th power

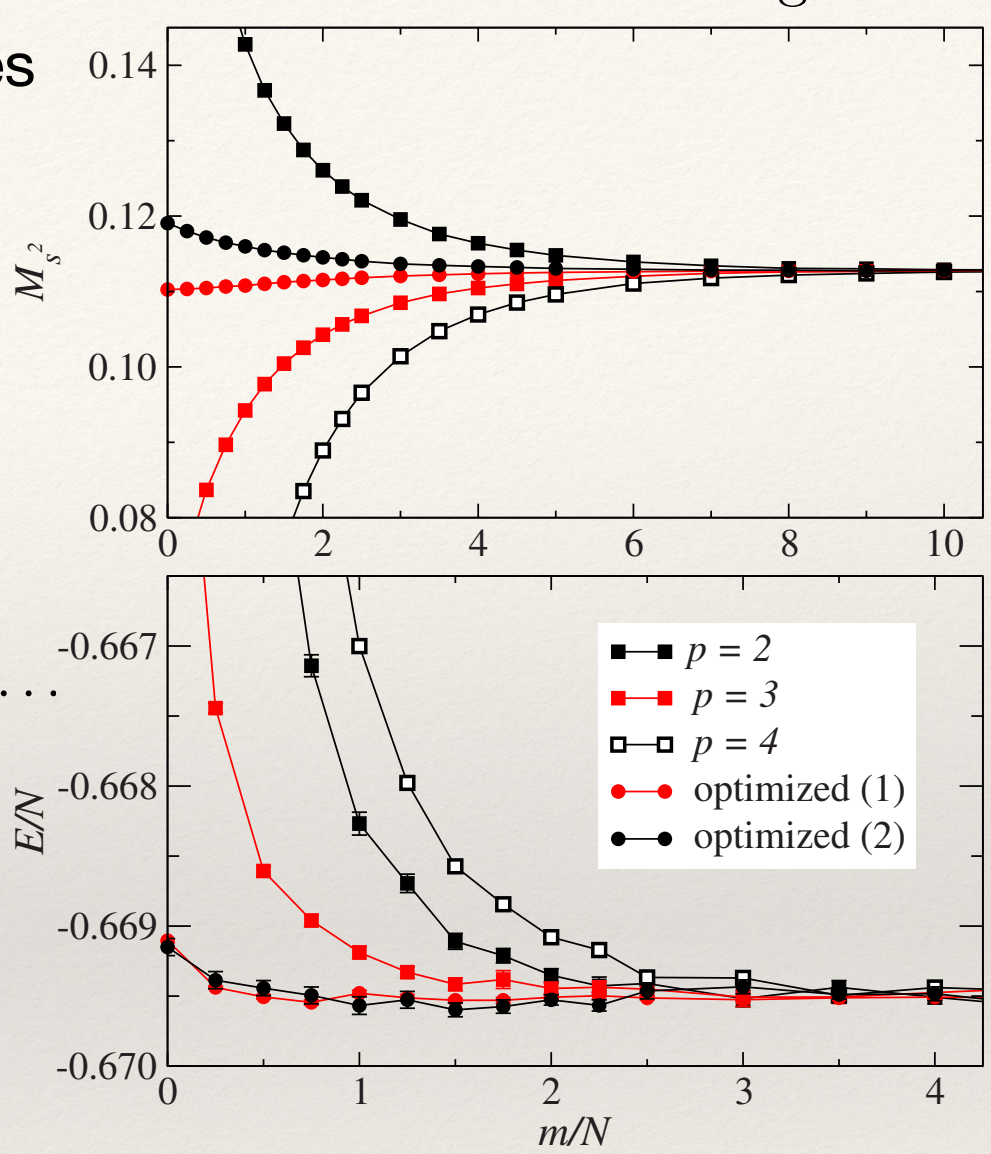
$$|\psi_m\rangle = H^m |\psi_0\rangle = \sum_n c_n E_n^m |n\rangle$$

Expectation value

$$\langle A\rangle_m = \langle 0|A|0\rangle + 2\langle 1|A|0\rangle \frac{c_1}{c_0}\left(\frac{E_1}{E_0}\right)^m + \dots$$

$$\langle A\rangle_m = \langle 0|A|0\rangle + c \times \exp\left(-\frac{m}{N}\frac{\Delta}{|e_0|}\right)$$

$$e_0 = E_0/M, \quad \Delta = E_1 - E_0$$

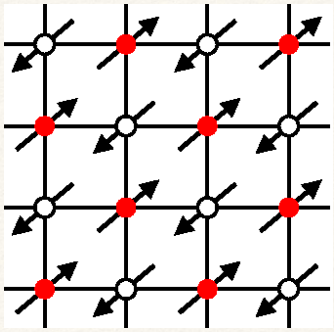


Conclusion:

- m/N >> e_0/Δ
- in valence-bond basis Δ is the singlet-singlet gap
- trial state also can have fixed momentum k=0 (e.g., ampl. product state)
  - only k=0 excited states (gap)

Results for 2D Heisenberg model

$$\mathbf{H} = \mathbf{J} \sum_{\langle \mathbf{i},\mathbf{j} \rangle} \mathbf{S_i} \cdot \mathbf{S_j}$$



Sublattice magnetization

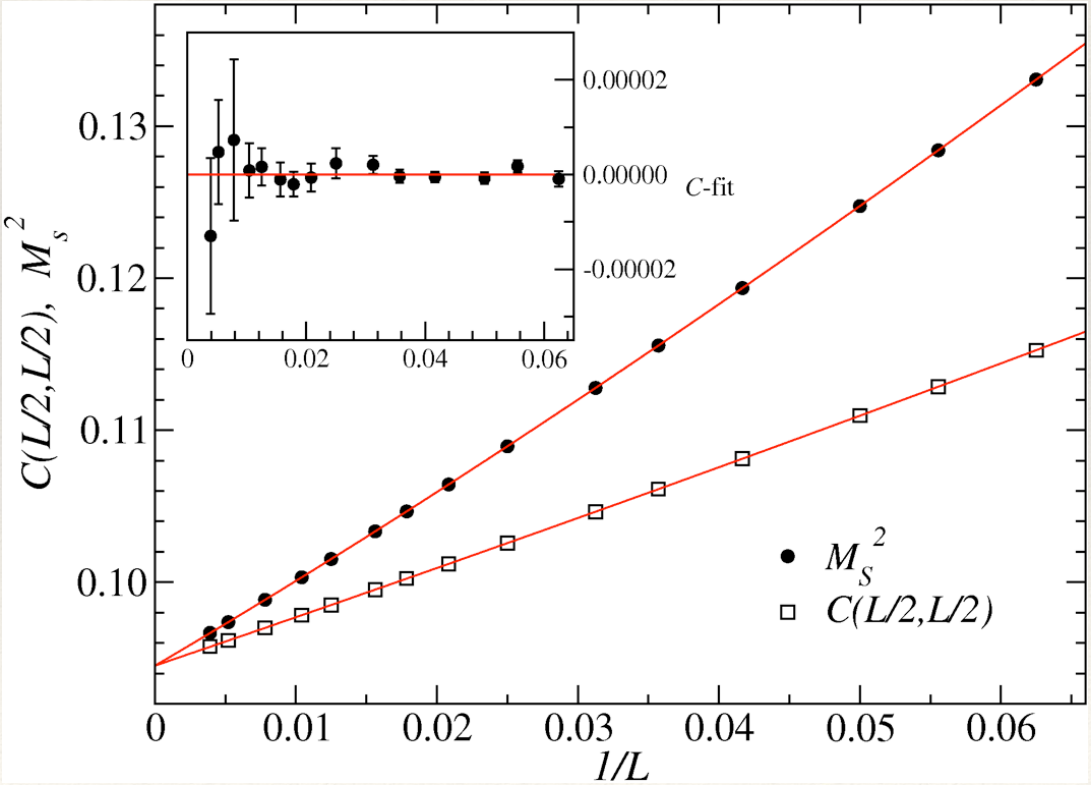
$$\vec{m}_s = \frac{1}{N} \sum_{i=1}^{N} \phi_i \vec{S}_i, \qquad \phi_i = (-1)^{x_i + y_i}$$

Long-range order: $\langle m_s^2 \rangle > 0$ for $N \to \infty$

Quantum Monte Carlo

- finite-size calculations
- no approximations
- extrapolation to infinite size

L×L lattices up to 256×256, T→0



Reger & Young (world-line) 1988

$$m_s = 0.30(2)$$

$$\approx 60\ \%\ \text{of classical value}$$

AWS & HG Evertz 2010

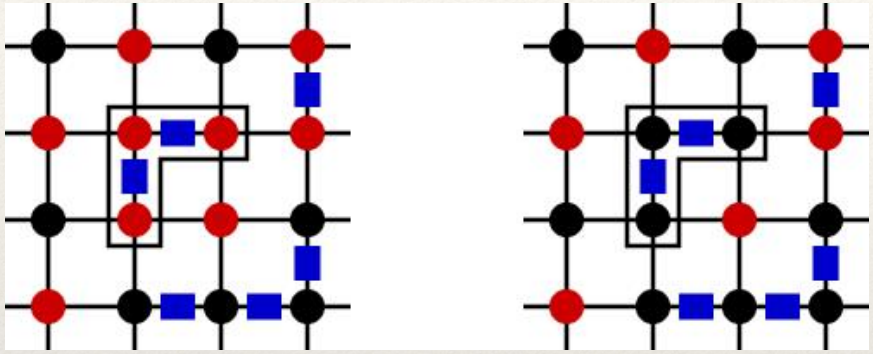
$$m_s = 0.30743(1)$$

Improved Estimators

Valence-bond projector QMC and SSE with loop updates are examples of cluster algorithms

- we can utilize improved estimators for many observables

Classical example: Swendsen-Wang Ising cluster algorithm



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

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

(unchanged after flip)

Write magnetization as sum over clusters of size n_C, sign s_C:

$$M = \sum_{i=1}^{N} \sigma_i = \sum_{C=1}^{N_{\rm clus}} \sum_{i \in C} \sigma_i = \sum_{C=1}^{N_{\rm clus}} s_C n_C \qquad \langle M^2 \rangle = \sum_{C=1}^{N_{\rm clus}} \sum_{C'=1}^{N_{\rm clus}} \langle n_C n_{C'} s_C s_{C'} \rangle$$

All cluster orientations (signs) have same weight

- average over all $2^{N_{\rm clus}}$ orientations →

$$\langle M^2 \rangle = \sum_{C=1}^{N_{\rm clus}} \langle n_C^2 \rangle$$

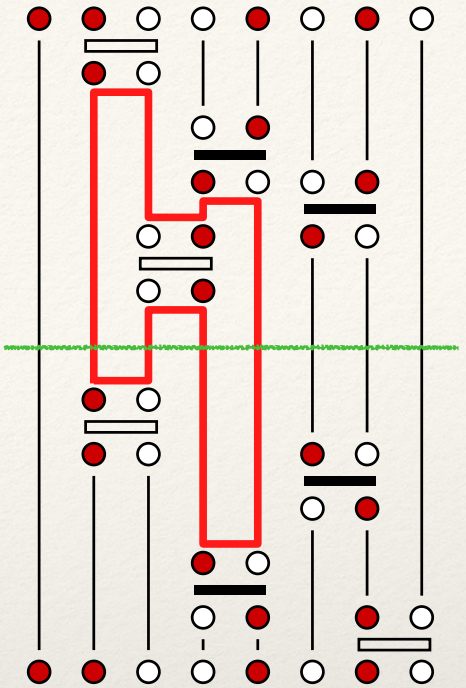
This is the improved estimator of $\langle M^2 \rangle$

- only depends on cluster structure

Improved estimators in SSE

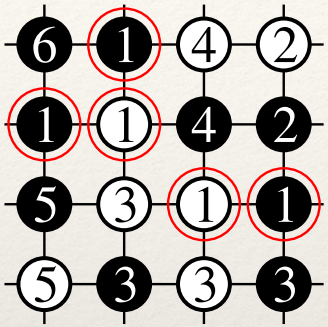
Consider a given slice (propagated state) of an SSE configuration

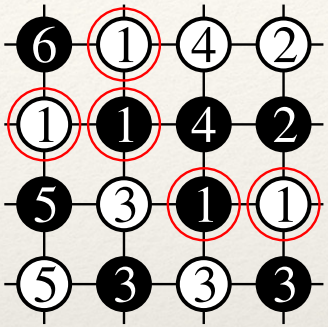
- label the sites according to the loops passing through → clusters



In given loop

All spins on given sub-lattice A or B are same, different on A, B





Staggered magnetization on a cluster is 1/2 of the size of the cluster

- changes sign when loop flipped
- similar to magnetization in SW algorithm

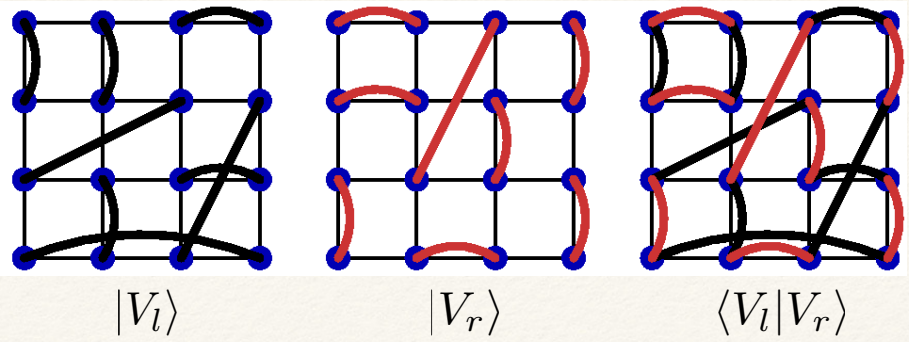
$$\langle M^2_{z,\mathrm{stagg}} \rangle = \frac{1}{4} \sum_{C=1}^{N_{\mathrm{clus}}} \langle n_C^2 \rangle$$

The uniform magnetization requires the staggered phases

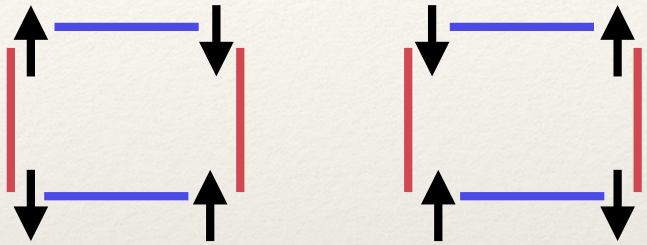
$$\chi = \frac{\beta}{4N} \left\langle \sum_{j=1}^{C} \left(\sum_{i=1}^{n_j} \phi_i \right)^2 \right\rangle \qquad \phi_i = \begin{cases} +1 & i \text{ on A site} \\ -1 & i \text{ on B site} \end{cases}$$

Valence-vond Projector QMC

The transition graphs give us improved estimators automatically



Put the spins back in:
- staggered spin configurations on each loop
- two 'orientations' (loop flips)



Average over all the two orientations of all the loops
- 2^{Nloop} configurations

$$\langle M^2_{z,\text{stagg}} \rangle = \frac{1}{4} \sum_{C=1}^{N_{\text{clus}}} \langle n_C^2 \rangle$$

Rotationally averaged correlation function

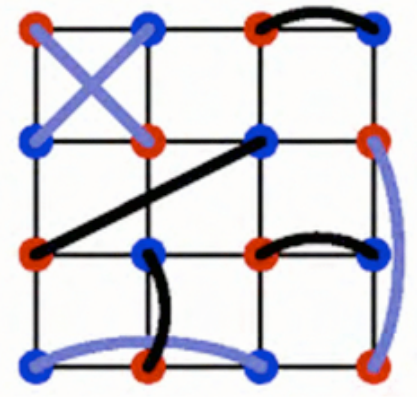
$$\frac{\langle V_l | \vec{S}_i \cdot \vec{S}_j | V_r \rangle}{\langle V_l | V_r \rangle} = \begin{cases} \frac{3}{4}(-1)^{x_i - x_j + y_i - y_j} & \text{(i,j in same loop)} \\ 0 & \text{(i,j in different loops)} \end{cases}$$

Multi-spin correlations: estimators with two or more loops
[Beach and Sandvik, Nucl. Phys. B (2006)]

Frustrated systems

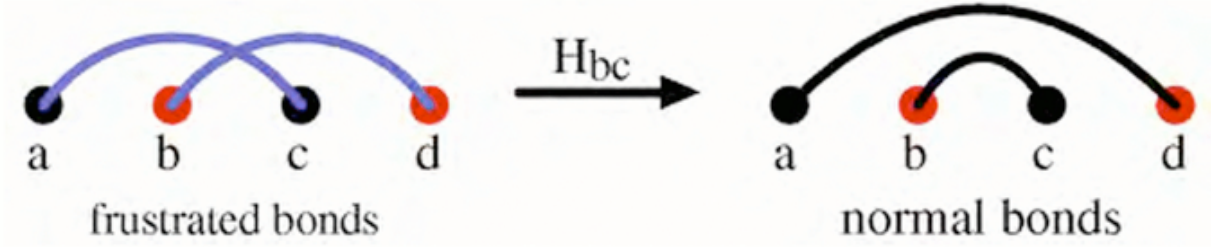
Consider the full valence-bond basis, including

- **normal bonds**, connecting A,B spins (sublattices)
- frustrated bonds, connecting A,A or B,B



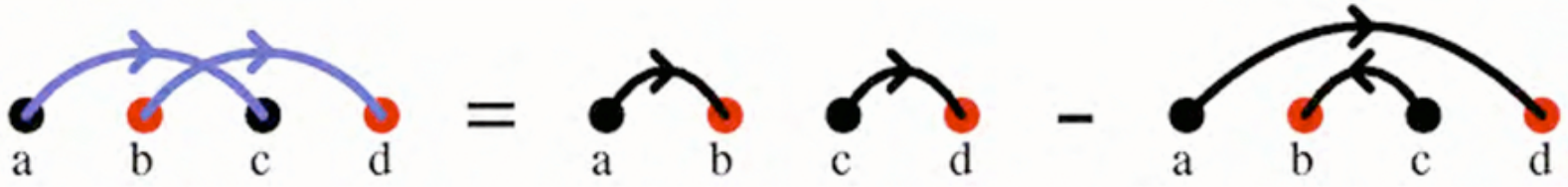
For a non-frustrated system

- projection eliminates frustarted bonds



For a frustrated system

- frustrated bonds remain and cause a sign problem
- frustrated bonds can be eliminated using over-completeness



In a simulation, one of the branches can be randomly chosen

- but there is a sign problem