5. Quantum Monte Carlo

Quantum Monte Carlo

- Not as easy as classical Monte Carlo

$$Z = \sum_c e^{-E_c / k_B T}$$

- Calculating the eigenvalues E_c is equivalent to solving the problem

- Need to find a mapping of the quantum partition function to a classical problem

$$Z = \mathrm{Tr}\, e^{-\beta H} \equiv \sum_c p_c$$

- "Negative sign" problem if some $p_c < 0$

Quantum Monte Carlo

- Feynman (1953) lays foundation for quantum Monte Carlo
- Map quantum system to classical world lines

THE

PHYSICAL REVIEW

A journal of experimental and theoretical physics established by E. L. Nichols in 1893

SECOND SERIES, VOL. 91, NO. 6 SEPTEMBER 15, 1953

Atomic Theory of the λ Transition in Helium

R. P. FEYNMAN
California Institute of Technology, Pasadena, California

(Received May 15, 1953)

Use Metropolis algorithm to update world lines

Diagrammatic QMC

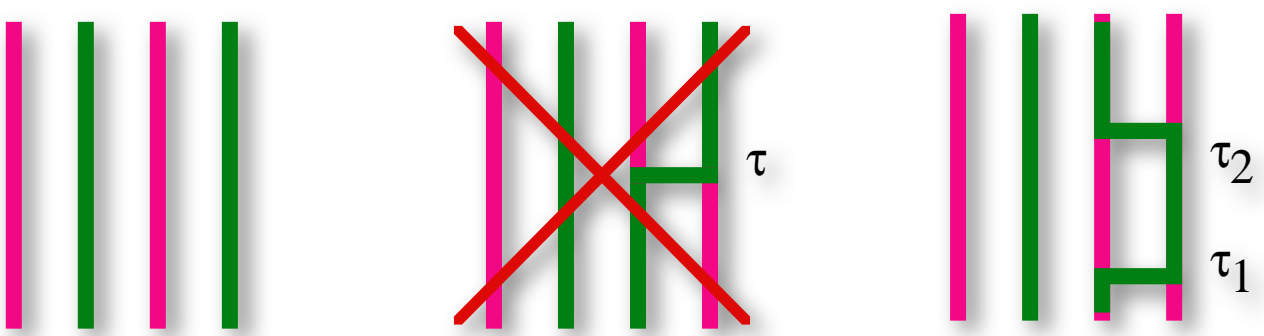
- Split the Hamiltonian into diagonal term H_o and perturbation V
- Then perform time-dependent perturbation theory

$$H = H_0 + V, \quad H_0 = \sum_{<i,j>} J_{ij}^z S_i^z S_j^z - \sum_i h S_i^z, \quad V = \sum_{<i,j>} J_{ij}^{xy} (S_i^x S_j^x + S_i^y S_j^y)$$

$$Z = \mathrm{Tr}(e^{-\beta H}) = \mathrm{Tr}(e^{-\beta H_0} \mathrm{T} e^{-\int_0^\beta d\tau V(\tau)})$$

$$Z = \mathrm{Tr}(e^{-\beta H_0} (1 - \int_0^\beta d\tau V(\tau) + \int_0^\beta d\tau_1 \int_{\tau_1}^\beta d\tau_2 V(\tau_1) V(\tau_2) + \ldots))$$

- Each term is represented by a diagram (world line configuration)



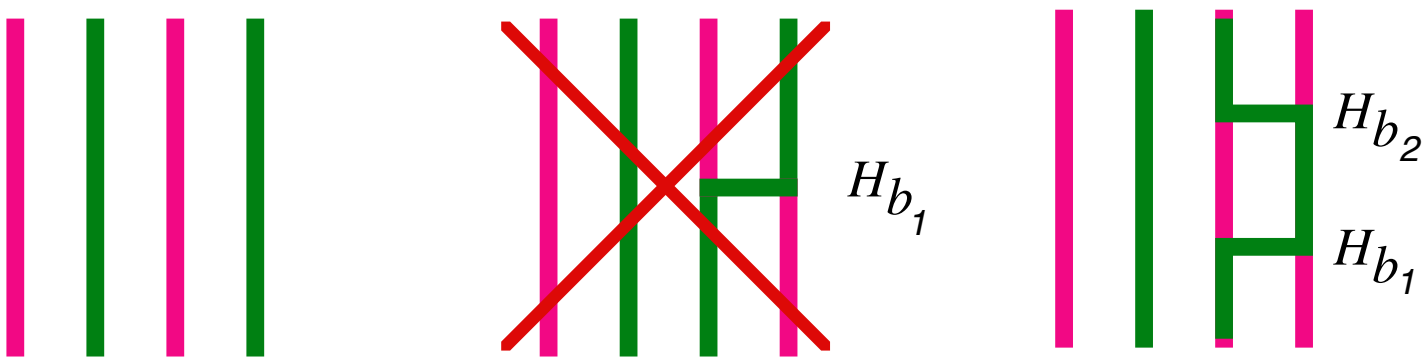
Stochastic Series Expansion

- based on high temperature expansion, developed by Sandvik

$$Z = \mathrm{Tr}(e^{-\beta H}) = \sum_{n=0}^{\infty} (-\beta)^n \, \mathrm{Tr}(H^n)$$

$$= \sum_{n=0}^{\infty} \frac{\beta^n}{n!} \sum_{|\alpha\rangle} \sum_{(b_1,\ldots,b_n)} \langle \alpha | \prod_{i=1}^{n} (-H_{b_i}) | \alpha \rangle$$

$$\text{with } H = \sum_i H_i$$



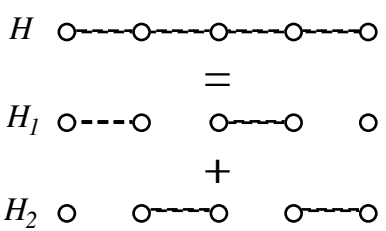
- Similar world line representation but without times assigned

The Suzuki-Trotter Decomposition

- Generic mapping of a quantum spin system to Ising model
  - basis of most discrete time QMC algorithms
  - not limited to special models
- Split Hamiltonian into two easily diagonalized pieces

$$H = H_1 + H_2$$

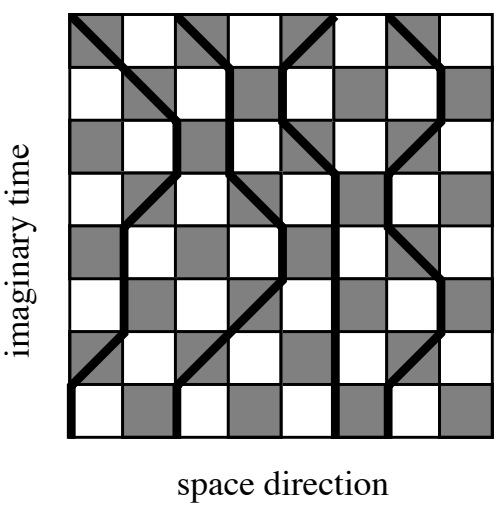
$$e^{-\varepsilon H} = e^{-\varepsilon(H_1+H_2)} = e^{-\varepsilon H_1} e^{-\varepsilon H_2} + O(\varepsilon^2)$$



- Obtain the checkerboard decomposition

$$Z = \mathrm{Tr}[\exp(-\beta H)] = \mathrm{Tr}\left[e^{-\beta(H_1+H_2)}\right]$$

$$= \mathrm{Tr}\left[e^{-(\beta/M)H_1} e^{-(\beta/M)H_2}\right]^M + O(\beta^3/M^2)$$



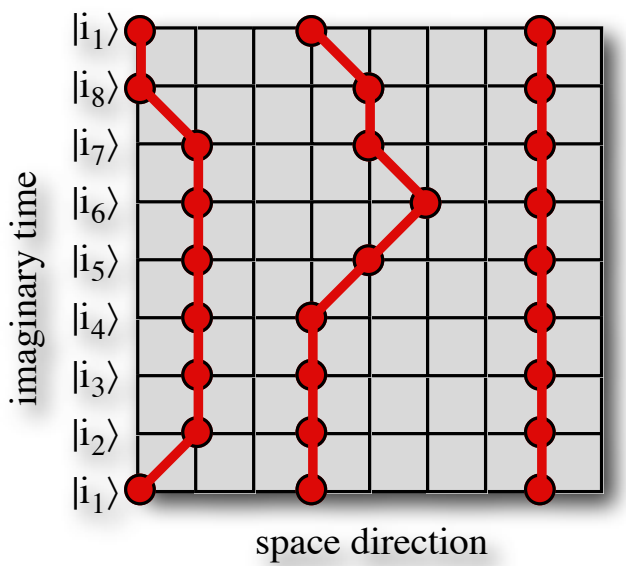
Path integral QMC

- Use Trotter-Suzuki or a a simple low-order formula

$$Z = \mathrm{Tr}\, e^{-\beta H} = \mathrm{Tr}\, e^{-M\Delta\tau H} = \mathrm{Tr}\left(e^{-\Delta\tau H}\right)^M = \mathrm{Tr}(1-\Delta\tau H)^M + O(\beta\Delta\tau)$$

$$= \sum_{\{(i_1 \ldots i_M)\}} \langle i_1 | 1-\Delta\tau H | i_2 \rangle \langle i_2 | 1-\Delta\tau H | i_3 \rangle \cdots \langle i_M | 1-\Delta\tau H | i_1 \rangle$$

- gives a mapping to a (d+1)-dimensional classical model



place particles (spins)

for Hamiltonians conserving particle number (magnetization) we get world lines

- partition function of quantum system is sum over classical world lines

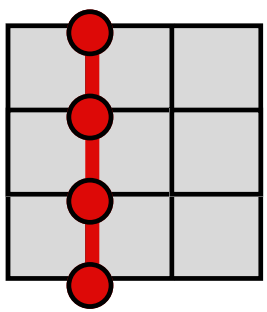
Calculating configuration weights

$$Z = \mathrm{Tr}\, e^{-\beta H} = \mathrm{Tr}\, e^{-M\Delta\tau H} = \mathrm{Tr}\left(e^{-\Delta\tau H}\right)^M = \mathrm{Tr}(1-\Delta\tau H)^M + O(\beta\Delta\tau)$$

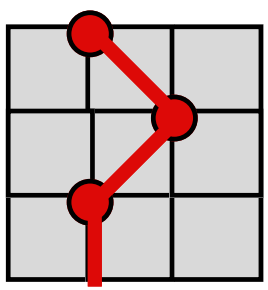
$$= \sum_{\{(i_1 \ldots i_M)\}} \langle i_1|1-\Delta\tau H|i_2\rangle\langle i_2|1-\Delta\tau H|i_3\rangle \cdots \langle i_M|1-\Delta\tau H|i_1\rangle$$

- Examples: particles with nearest neighbor repulsion

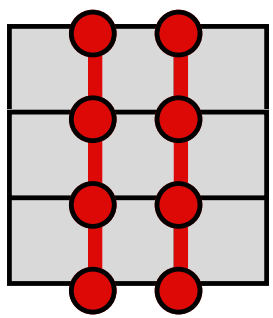
$$H = -t\sum_{\langle i,j\rangle}(a_i^\dagger a_j + a_j^\dagger a_i) + V\sum_{\langle i,j\rangle} n_i n_j$$



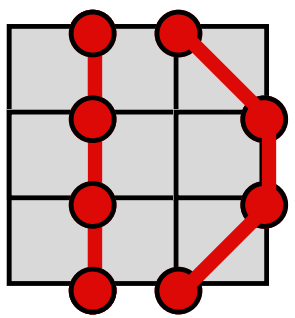
1



$(\Delta\tau t)^2$



$(1-\Delta\tau V)^3$

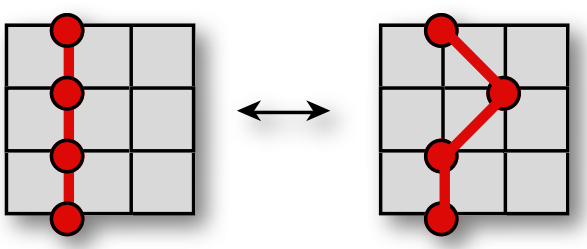


$(\Delta\tau t)^2$

Monte Carlo updates

- just move the world lines locally
  - probabilities given by matrix element of Hamiltonian
  - example: tight binding model $H = -t \sum_{\langle i,j \rangle} \left(c_i^\dagger c_{i+1} + c_{i+1}^\dagger c_i \right)$

introduce or remove two kinks:

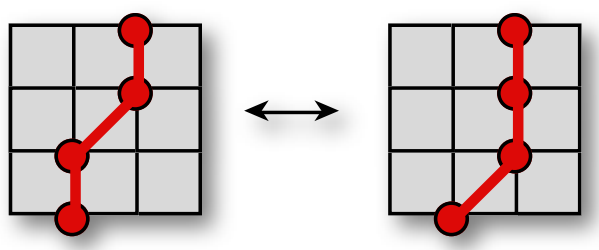


$P = 1$ $\qquad P = (\Delta\tau t)^2$

$P_\rightarrow = \min\left[1,(\Delta\tau t)^2\right]$

$P_\leftarrow = \min\left[1,1/(\Delta\tau t)^2\right]$

shift a kink:

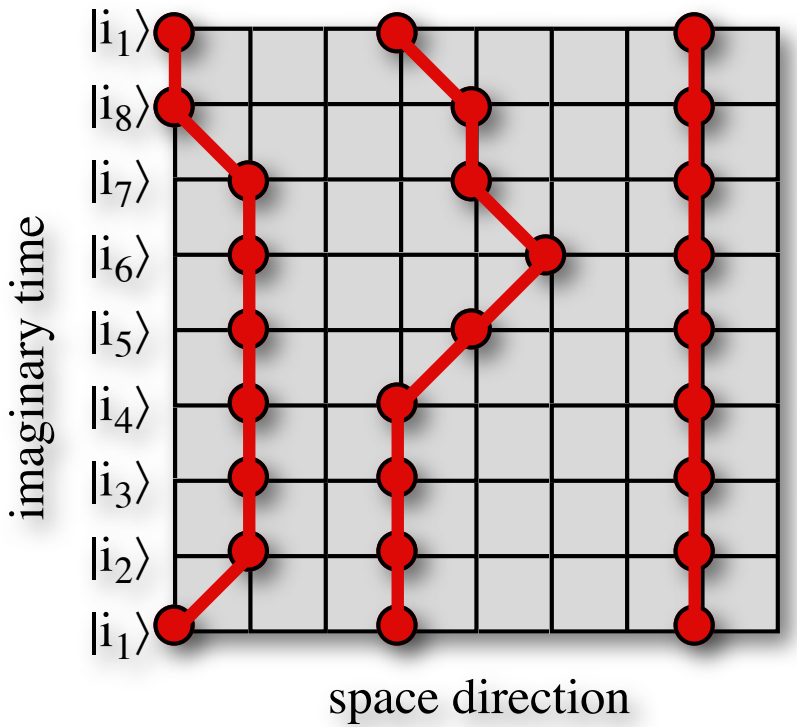


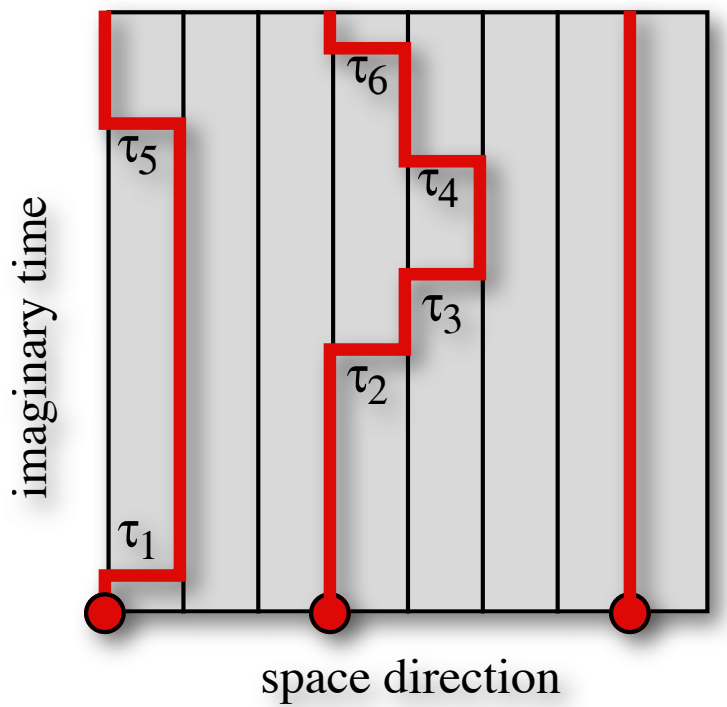
$P = \Delta\tau t$

$P_\rightarrow = P_\leftarrow = 1$

The continuous time limit

- the limit $\Delta\tau \to 0$ can be taken in the algorithm
  [Prokof'ev *et al.*, Pis'ma v Zh.Eks. Teor. Fiz. **64**, 853 (1996)]





- discrete time: store configuration at all time steps
- continuous time: store times at which configuration changes

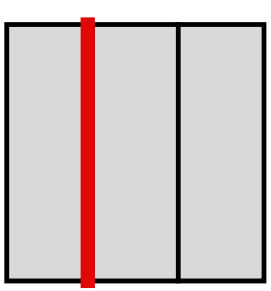
Calculating configuration weights

- Continuous time algorithms just sample time-depedent perturbation expansion

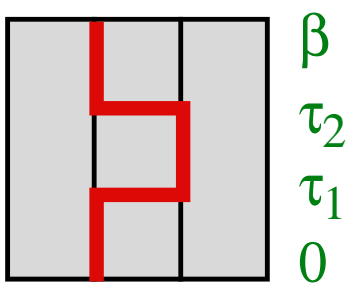
$$Z = \mathrm{Tr}\left(e^{-\beta H_0} \mathcal{T} e^{-\int_0^\beta d\tau \mathcal{V}(\tau)} \right)$$

- Examples: particles with nearest neighbor repulsion

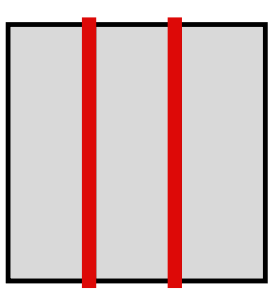
$$H_0 = V \sum_{\langle i,j \rangle} n_i n_j \qquad \mathcal{V} = -t \sum_{\langle i,j \rangle} (a_i^\dagger a_j + a_j^\dagger a_i)$$



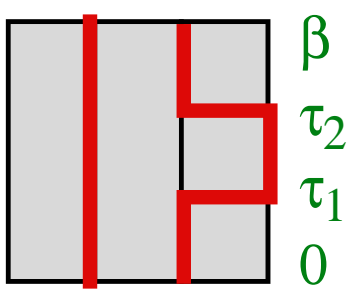
1



$t^2 d\tau_1 d\tau_2$



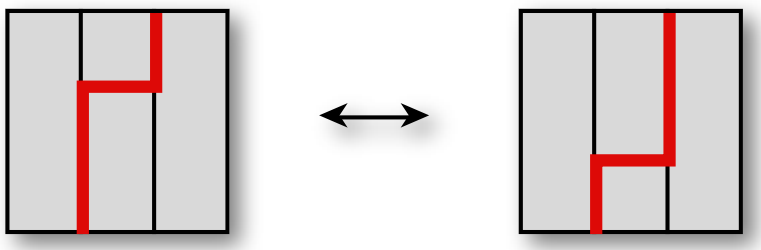
$e^{-\beta V}$



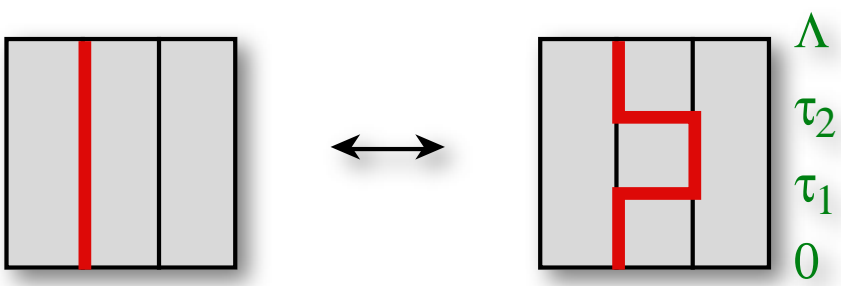
$e^{-\tau_1 V} e^{-(\beta-\tau_2)V} t^2 d\tau_1 d\tau_2$

Updates in continuous time

- Shift a kink to any new position:



- Insert a pair of kinks:



solution:
integrate over all possible insertions in an interval

$$P = 1$$

$$P = (\Delta\tau t)^2 \to 0$$

$$P_{\to} = \min\left[1,(\Delta\tau t)^2\right] \to 0$$

vanishing acceptance rate

$$P = \int_0^{\Lambda}\int_{\tau_1}^{\Lambda} t^2 d\tau_2 d\tau_1 = \frac{\Lambda^2 t^2}{2} \neq 0$$

$$P_{\to} = \min\left[1,\Lambda^2 t^2/2\right] \neq 0$$

Advantages of continuous time

- No need to extrapolate in time step
  - a single simulation is sufficient
  - no additional errors from extrapolation
- Less memory and CPU time required
  - Instead of a time step $\Delta\tau \ll t$ we only have to store changes in the configuration happening at mean distances $\approx t$
  - Speedup of $1 / \Delta\tau \approx 10$
- Conceptual advantage
  - we directly sample a diagrammatic perturbation expansion

6. Cluster updates and the loop algorithm

Problems with local updates

- Local updates cannot change global topological properties
  - number of world lines (particles, magnetization) conserved
  - winding conserved
  - braiding conserved
  - cannot sample grand-canonical ensemble
- Critical slowing down at second order phase transitions
  - solved by cluster updates (today)
- Tunneling problem at first order phase transitions
  - solved by extended sampling techniques (S. Trebst's lecture)

Cluster algorithms: the formal explanation

- Extend the phase space to configurations + graphs (C,G)

$$Z = \sum_C W(C) = \sum_C \sum_G W(C,G) \text{ with } W(C) = \sum_G W(C,G)$$

- Choose graph weights independent of configuration

$$W(C,G) = \Delta(C,G)V(G) \text{ where } \Delta(C,G) = \begin{cases} 1 & \text{graph } G \text{ allowed for } C \\ 0 & \text{otherwise} \end{cases}$$

- Perform updates

2. Discard configuration

4. Discard graph

$$C_i \rightarrow (C_i,G) \rightarrow G \rightarrow (C_{i+1},G) \rightarrow C_{i+1}$$

1. Pick a graph G $P[G] = \frac{V(G)}{W(C)}$

3. Pick any allowed new configuration

- Detailed balance is satisfied

$$\frac{P[(C_i,G) \rightarrow (C_{i+1},G)]}{P[(C_{i+1},G) \rightarrow (C_i,G)]} = \frac{1/\mathrm{N_C}}{1/\mathrm{N_C}} = 1 = \frac{\Delta(C_{i+1},G)V(G)}{\Delta(C_i,G)V(G)} = \frac{P[(C_{i+1},G)]}{P[(C_i,G)]}$$

Cluster algorithms: Ising model

- We need to find $\Delta(C,G)$ and $V(G)$ to fulfill $W(C) = \sum_G W(C,G) = \sum_G \Delta(C,G) V(G)$

$\Delta(C,G)$	o-o	o o	$W(C)$
↑↑, ↓↓	I	I	$e^{+\beta J}$
↑↓, ↓↑	O	I	$e^{-\beta J}$
$W(G)$	$e^{+\beta J} - e^{-\beta J}$	$e^{-\beta J}$	

- This means for: $C_i \rightarrow (C_i,G) \rightarrow G$
  - Parallel spins: pick connected graph o-o with $P(\text{ o-o }) = \dfrac{e^{+\beta J} + e^{-\beta J}}{e^{+\beta J}} = 1 - e^{-2\beta J}$
  - Antiparallel spins: always pick open graph o o
- And for: $G \rightarrow (C_{i+1},G) \rightarrow C_{i+1}$
  - Configuration must be allowed ⇒ connected spins must be parallel

  ⇒ connected spins flipped as one cluster

PHYSICAL REVIEW
LETTERS

VOLUME 70 | 15 FEBRUARY 1993 | NUMBER 7

Cluster Algorithm for Vertex Models

Hans Gerd Evertz,[1],[a] Gideon Lana,[2],[b] and Mihai Marcu[2],[c]
[1] *Supercomputer Computations Research Institute, Florida State University, Tallahassee, Florida 32306*
[2] *School of Physics and Astronomy, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, 69978 Tel Aviv, Israel*
(Received 17 November 1992)

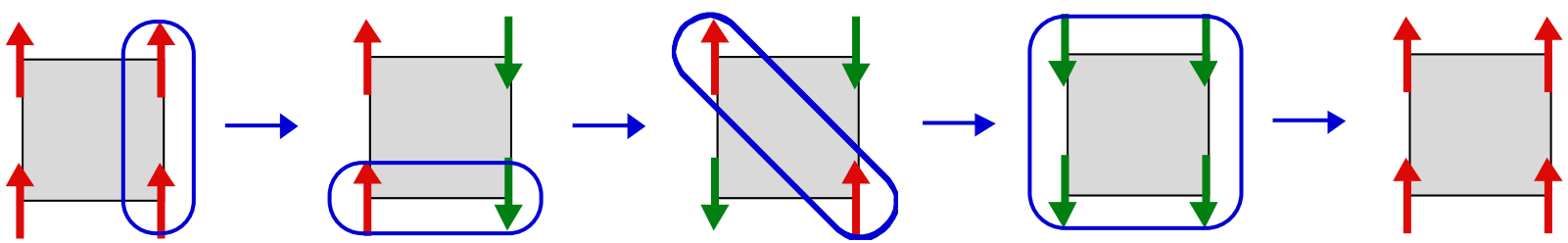
We present a new type of cluster algorithm that strongly reduces critical slowing down in simulations of vertex models. Since the clusters are closed paths of bonds, we call it the *loop algorithm*. The basic steps in constructing a cluster are the breakup and the freezing of vertices. We concentrate on the case of the F model, which is a subset of the six-vertex model exhibiting a Kosterlitz-Thouless transition. The loop algorithm is also applicable to simulations of other vertex models and of one- and two-dimensional quantum spin systems.

PACS numbers: 02.70.–c, 05.50.+q, 68.35.Rh, 75.10.Jm

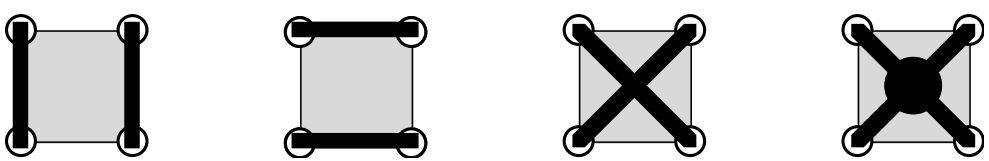
The loop algorithm (Evertz *et al*)

- Swendsen-Wang cluster algorithm for the Ising model
  - two choices on each bond: *connected (flip both spins)* or *disconnected*

  - all connected spins are flipped together
- Loop algorithm is a generalization to quantum systems
  - world lines may not be broken
  - always 2 or 4 spins must be flipped together



  - four different connection types



Hamiltonian of spin-1/2 models

- Consider a 2-site quantum spin-1/2 model

$$H_{XXZ} = J_{xz}(S_1^x S_2^x + S_1^y S_2^y) + J_z S_1^z S_2^z - h\left(S_1^z + S_2^z\right)$$

$$= \frac{J_{xz}}{2}(S_1^+ S_2^- + S_1^- S_2^+) + J_z S_1^z S_2^z - h\left(S_1^z + S_2^z\right)$$

  - Heisenberg model if $J_{xy} = J_z = J$

$$H = J\vec{S}_1\vec{S}_2 - h\left(S_1^z + S_2^z\right)$$

- Hamiltonian matrix in 2-site basis

$$\{\ |\uparrow\uparrow\rangle,\ |\uparrow\downarrow\rangle,\ |\downarrow\uparrow\rangle,\ |\downarrow\downarrow\rangle\ \}$$

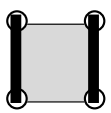
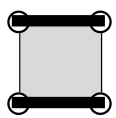
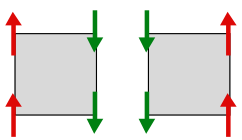
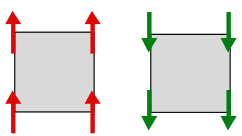
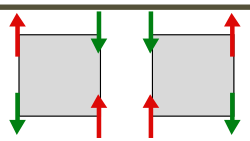
$$H_{XXZ} = \begin{pmatrix} \frac{J_z}{4} + h & 0 & 0 & 0 \\ 0 & -\frac{J_z}{4} & \frac{J_{xy}}{2} & 0 \\ 0 & \frac{J_{xy}}{2} & -\frac{J_z}{4} & 0 \\ 0 & 0 & 0 & \frac{J_z}{4} - h \end{pmatrix}$$

Cluster building rules: XY-like antiferromagnet

$$H_{XXZ} = \frac{J_{xz}}{2}\sum_{\langle i,j\rangle}(S_i^+ S_j^- + S_i^- S_j^+) + J_z \sum_{\langle i,j\rangle} S_i^z S_j^z$$

with $0 \le J_z \le J_{xy}$

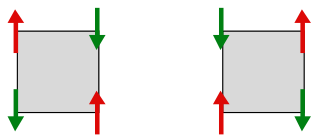
$$W(C) = \sum_G W(C,G) = \sum_G \Delta(C,G) V(G)$$

$\Delta(C,G)$				$W(C)$
	I	I	–	$1+ (J_z/4)\ d\tau$
	I	–	o	$1-(J_z/4)\ d\tau$
	–	I	I	$(J_{xy}/2)\ d\tau$
$V(G)$	$1-(J_z/4)\ d\tau$	$(J_z/2)\ d\tau$	$(J_{xy}- J_z)/2\ d\tau$	

- Connected spins form a cluster and have to be flipped together

How to deal with infinitesimals ?

- How do we deal with the vanishing $d\tau$ terms in continuous time?
- First example: the exchange process

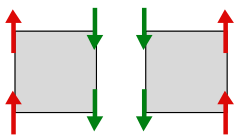


  - Possible graph connections:

  - Graph weights: $\frac{J_z}{2} d\tau$ $\frac{J_{xy} - J_z}{2} d\tau$
  - Probability to pick graph: (divide weight by sum) $\frac{J_z}{J_{xy}}$ $\frac{J_{xy} - J_z}{J_{xy}}$

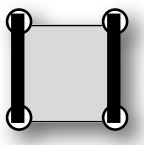
- The infinitesimal $d\tau$ terms cancel out
  - Randomly pick one of the graphs (with appropriate probabilities)

How to deal with infinitesimals ?

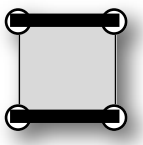
- How do we deal with the vanishing $d\tau$ terms in continuous time?
- Second example: the "decay" process 
  - Possible graph connections: 
  - Graph weights: $1-\frac{J_z}{4}d\tau$ $\frac{J_z}{2}d\tau$
  - Probability to pick graph: (divide weight by sum) $1-\frac{J_z}{2}d\tau$ $\frac{J_z}{2}d\tau$
- The infinitesimal $d\tau$ terms remain
- Infinitesimal acceptance rate at infinitely many time steps?

How to deal with infinitesimals ?

- We have to tackle the problem of vanishing probabilities
  - Example: Heisenberg antiferromagnet



$$P_{\parallel} = 1 - \frac{J}{2}\Delta\tau \to 1$$



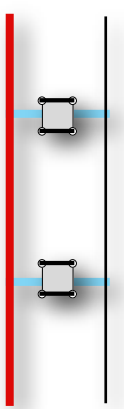
$$P_{=} = \frac{J}{2}\Delta\tau \to 0$$

- Interpret the  connection as a “decay process” where the loop jumps

$$P_{=} = \frac{J}{2}d\tau$$

$$\text{decay constant } \lambda = \frac{J}{2}$$

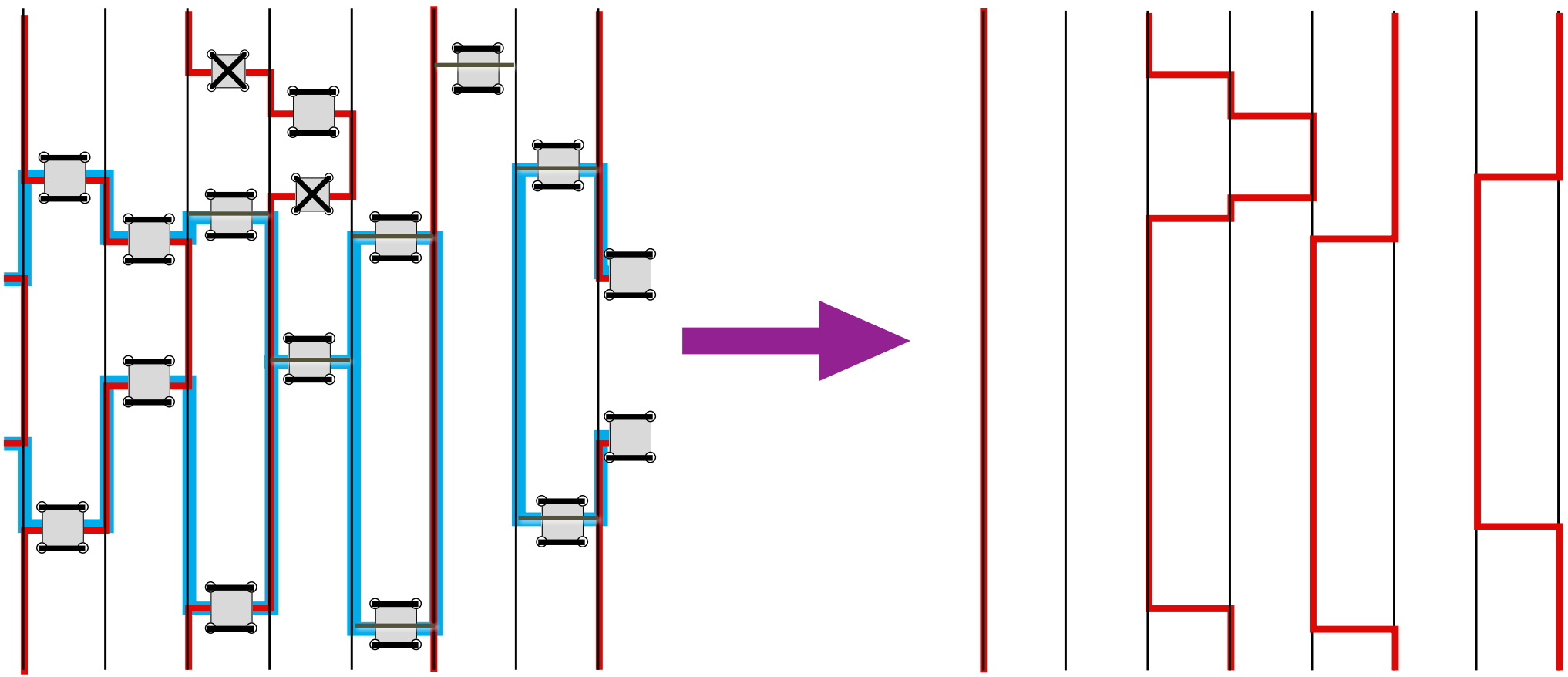
$$\text{mean distance } d = \frac{2}{J}$$



Loop-cluster updates

1. Connect spins according to loop-custer building rules

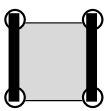
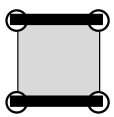
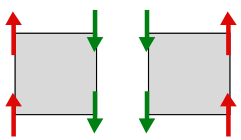
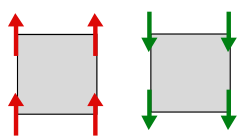
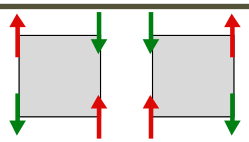
2. Build and flip loop-cluster



Heisenberg antiferromagnet

$$H_{\text{Heisenberg}} = J \sum_{\langle i,j \rangle} \vec{S}_i \vec{S}_j$$

$$W(C) = \sum_G W(C,G) = \sum_G \Delta(C,G) V(G)$$

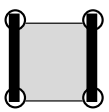
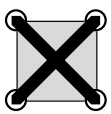
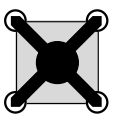
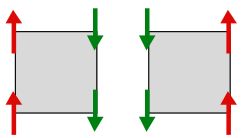
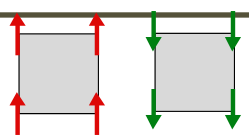
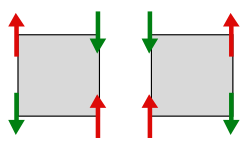
$\Delta(C,G)$			$W(C)$
	1	1	$1+(J/4)\,d\tau$
	1	0	$1-(J/4)\,d\tau$
	0	1	$(J/2)\,d\tau$
$V(G)$	$1-(J/4)\,d\tau$	$(J/2)\,d\tau$	

- Connected spins form a cluster and have to be flipped together
- Very simple and deterministic for Heisenberg model

Ising-like ferromagnet

$$H_{XXZ} = -\frac{J_{xz}}{2}\sum_{\langle i,j\rangle}(S_i^+ S_j^- + S_i^- S_j^+) - J_z \sum_{\langle i,j\rangle} S_i^z S_j^z \qquad W(C) = \sum_G W(C,G) = \sum_G \Delta(C,G) V(G)$$

with $0 \le J_{xy} \le J_z$

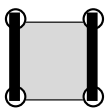
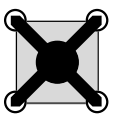
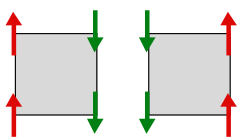
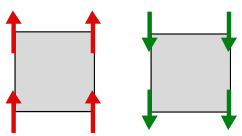
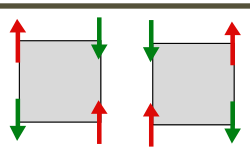
$\Delta(C,G)$				$W(C)$
	1	0	0	$1 - (J_z/4)\, d\tau$
	1	1	1	$1 + (J_z/4)\, d\tau$
	0	1	0	$(J_{xy}/2)\, d\tau$
$V(G)$	$1-(J_z/4)\, d\tau$	$(J_{xy}/2)\, d\tau$	$(J_z - J_{xy})/2\, d\tau$	

- Now 4-spin freezing graph is needed: connects (freezes) loops

Ising ferromagnet

$$H_{\text{Ising}} = -J\sum_{\langle i,j\rangle} S_i^z S_j^z = -\frac{J}{4}\sum_{\langle i,j\rangle}\sigma_i\sigma_j$$

$$W(C) = \sum_G W(C,G) = \sum_G \Delta(C,G)V(G)$$

$\Delta(C,G)$			$W(C)$
	I	O	1- $(J/4)\ d\tau$
	I	I	1+ $(J/4)\ d\tau$
	O	O	0
$V(G)$	$1-(J/4)\ d\tau$	$(J/2)\ d\tau$	

- Two spins are frozen if there is any freezing graph along the world line

$$P_{\text{no freezing}} = \lim_{M\to\infty}(1-(\beta/M)J/2)^M = \exp(-\beta J/2) = \exp(-2\beta J_{classical})$$

- We recover the Swendsen Wang algorithm: probability for no freezing

Modern Monte Carlo algorithms

- Which system sizes can be studied?

temperature	Metropolis	modern algorithms
3D Tc	16'000 spins	16'000'000 spins
0.1 J	200 spins	1'000'000 spins
0.005 J	—	50'000 spins
3D Tc	32 bosons	1'000'000 bosons
0.1 t	32 bosons	10'000 bosons