A Quick introduction to Quantum Monte Carlo methods

Fabien Alet **LPT, Univ. Paul Sabatier Toulouse**

Contact : alet@irsamc.ups-tlse.fr

Quantum Monte Carlo

- What is Quantum Monte Carlo (QMC) ?
  - Most general definition: “A stochastic method to solve the Schrödinger equation”
- QMC comes in many different flavours
  - Zero Temperature or $T \neq 0$ $\times 2$
  - Statistics of particles : Fermions, Bosons or “Boltzmannons” $\times 3$
  - Continuum or discrete space $\times 2$
  - Finite number of particles or Thermodynamic limit $\times 2$
  - …

At least 24 different QMC communities !

- Today
  - Spin models on lattice : finite T, Bosons (spins), Discrete space, Thermodynamic limit
- Performances
  - Large systems ($\approx$ thermodynamic limit, $\approx$ T=0) : up to 10^6 spins !
  - Quantitative modeling of quantum magnets
  - Accurate description of phase transitions, Asymptotic regime (critical exponents)

QMC = a large family tree

- « Variational Monte Carlo » family
- « Diffusion Monte Carlo » family
  - Projector Monte Carlo, Green Function Monte Carlo, Diffusion Monte Carlo (+ Fixed Node, + Stochastic Reconfiguration) …
  - (Partial) Ref. for lattice models : S. Sorella and L. Capriotti, Phys. Rev. B **61**, 2599 (2000)
- « Determinental Monte Carlo » family
  - Auxiliary Field Monte Carlo, Determinental Monte Carlo, Hirsch-Fye …
  - (Partial) Ref. for lattice models : F. F. Assaad, Lecture Notes, NIC series **10**, 99 (2003)
- « Path integral Monte Carlo » family
  - Lattice systems (cluster algorithms)
  - Basic Idea : Mapping Quantum system in dimension d → Classical system in dimension d+1

    Then do classical Monte Carlo on the equivalent problem

A Quick introduction to Quantum Monte Carlo methods for lattice quantum spin models

- Derivation of configuration space : Path integral approach
- Monte Carlo moves : quantum cluster algorithms
- The sign problem
- Practical considerations

Quantum Monte Carlo simulations

- Not as « easy » as classical Monte Carlo

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

- Calculating the energy eigenvalue E_c = solving the problem
- Need to find a mapping of the quantum partition function to a classical problem

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

- Different approaches
  - Path integrals (time-dependent perturbation theory in imaginary time)
  - *Stochastic Series Expansion* (high temperature expansion)
- Sign problem if some $p_c < 0$ (thus try to avoid this)
- Then need efficient updates for the equivalent classical problem

Hamiltonian of spin ½ models

- Example : XXZ model in a field
  - Anisotropic exchange interactions : J_{XY} , J_Z
  - Magnetic field h

$$H_{XXZ} = \sum_{\langle i,j \rangle} J_{XY}(S_i^x S_j^x + S_i^y S_j^y) + J_z S_i^z S_j^z - h\sum_i S_i^z$$

$$= \sum_{\langle i,j \rangle} \frac{J_{XY}}{2}(S_i^+ S_j^- + S_i^- S_j^+) + J_z S_i^z S_j^z - h\sum_i S_i^z$$

  - Heisenberg model : $J_{XY} = J_Z = J$

$$H = J\sum_{\langle i,j \rangle} \vec{S}_i \vec{S}_j - h\sum_i S_i^z$$

- Hamiltonian matrix in 2-site basis

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

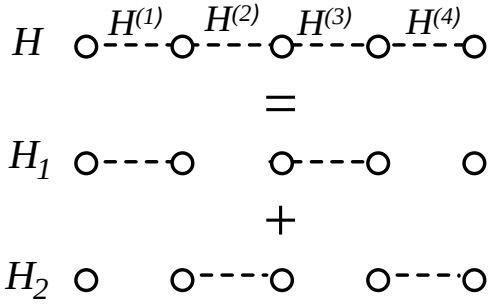
$$H_{ij} = \begin{pmatrix} \frac{J_z}{4} + h & 0 & 0 & 0 \\ 0 & -\frac{J_z}{4} & \frac{J_{xy}}{2} & 0 \\ 0 & \frac{J_{xy}}{2} & -\frac{J_z}{4} & 0 \\ 0 & 0 & 0 & \frac{J_z}{4} - h \end{pmatrix}$$

Trotter decomposition

- Basis of most QMC algorithms
  - Here : Generic mapping of a quantum spin system onto a classical Ising model
  - Not limited to special cases
- Split Hamiltonian into two easily diagonalizable pieces

$$H = H_1 + H_2$$

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



- Obtain a decomposition of the partition function

$$Z = \mathrm{Tr}\, e^{-\beta H} = \mathrm{Tr}\, e^{-\beta(H_1+H_2)} = \mathrm{Tr}[(e^{-\Delta\tau(H_1+H_2)})^M]$$

$$= \mathrm{Tr}[(e^{-\Delta\tau H_1} e^{-\Delta\tau H_2})^M] + O(\Delta\tau^2) \qquad (\Delta\tau = \beta / M)$$

  - Insert 2M sets of complete basis states

$$= \sum_{i_1,\ldots,i_{2M}} \langle i_1 | e^{-\Delta\tau H_1} | i_{2M} \rangle \langle i_{2M} | e^{-\Delta\tau H_2} | i_{2M-1} \rangle \cdots \langle i_3 | e^{-\Delta\tau H_1} | i_2 \rangle \langle i_2 | e^{-\Delta\tau H_2} | i_1 \rangle$$

Example : Spin ½ Heisenberg model

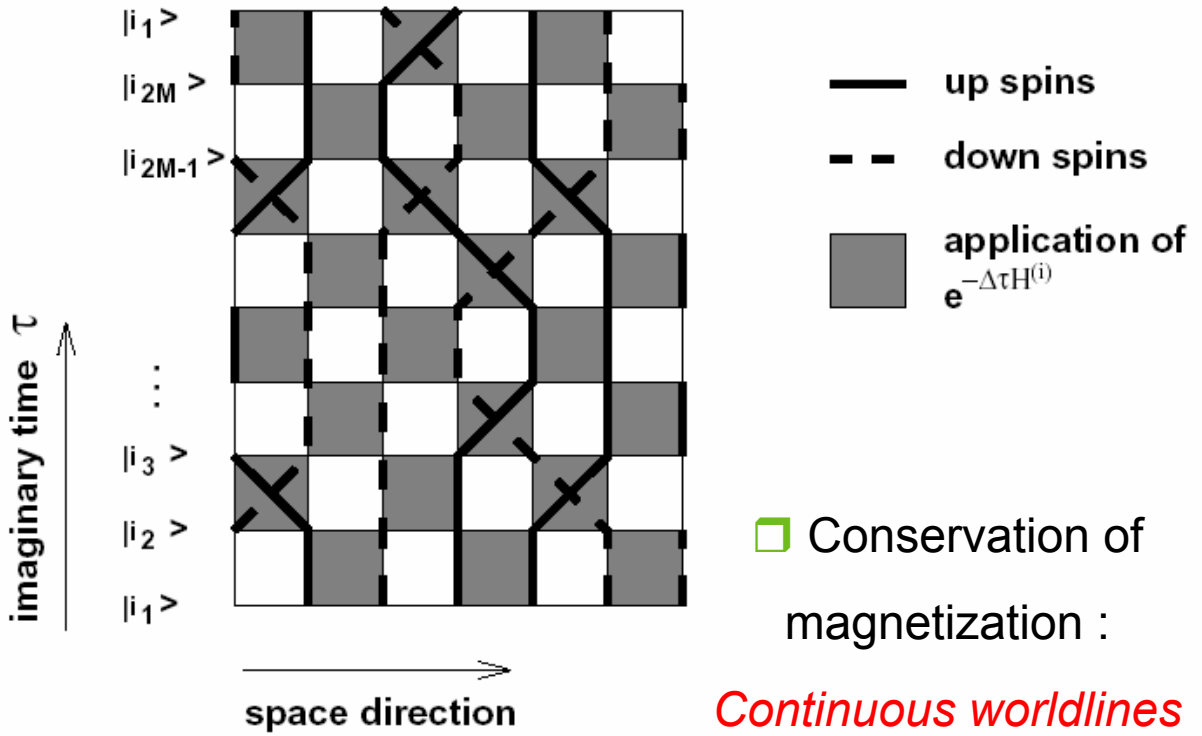
- Quantum problem in d dimensions maps onto a classical problem in d+1
  - Expand the states $|i_\alpha\rangle$ in the S^z eigenbasis
  - Effective Ising-model in d+1 dimensions with 2- and 4-sites interaction terms

$$Z = \sum_{i_1,\ldots,i_{2M}} \langle i_1 | e^{-\Delta\tau H_1} | i_{2M} \rangle \langle i_{2M} | e^{-\Delta\tau H_2} | i_{2M-1} \rangle \cdots \langle i_3 | e^{-\Delta\tau H_1} | i_2 \rangle \langle i_2 | e^{-\Delta\tau H_2} | i_1 \rangle$$

- Each of the matrix elements

$$\langle i_{j+1} | e^{-\Delta\tau H_{1,2}} | i_j \rangle$$

corresponds to a row of shaded plaquettes and equals the product over those plaquettes



- Conservation of magnetization :

Continuous worldlines

Weights for the spin ½ Heisenberg model

- The partition function becomes a sum of products of plaquette weights

$$Z = \sum_C W(C) = \sum_C \prod_{plaquettes\ p} w(C_p)$$

- The only allowed plaquette-configurations are (here $h=0$)

C_p			
$w(C_p)$	$e^{-\Delta\tau J/4}$	$e^{\Delta\tau J/4} ch(\Delta\tau J/2)$	$e^{\Delta\tau J/4} sh(-\Delta\tau J/2)$

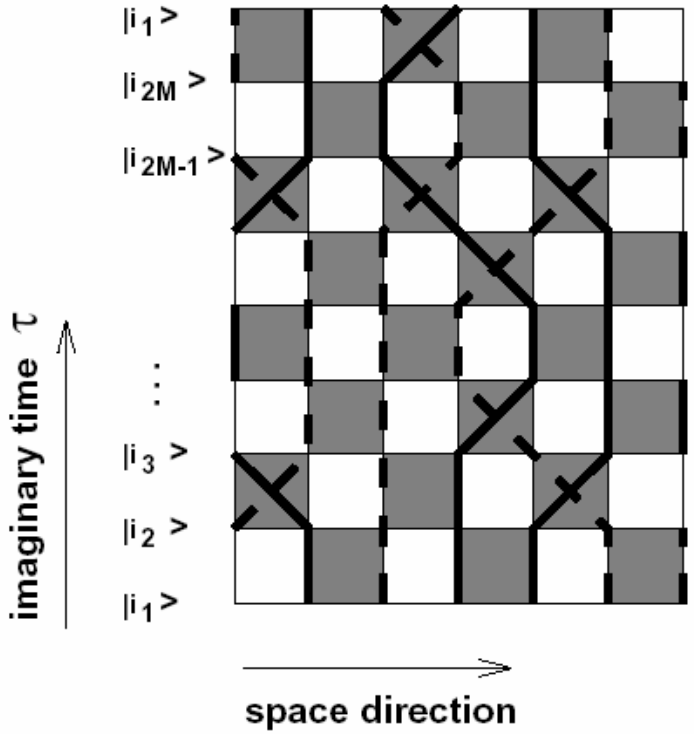
- Ferromagnet (J<0) : All weights are positive
- Antiferromagnet on a bipartite lattice :
  - perform a gauge transformation on one sublattice

$$\frac{J}{2}\left(S_i^+ S_j^- + S_i^- S_j^+\right) \xrightarrow{S_i^\pm \to (-1)^{|i|} S_i^\pm} -\frac{J}{2}\left(S_i^+ S_j^- + S_i^- S_j^+\right)$$

- Frustrated antiferromagnet : we have a **sign problem**

Worldline approach : Summary

- Each valid configuration = *continuous* worldlines on checkerboard



up spins

down spins

application of $e^{-\Delta\tau H^{(i)}}$

$$W(C) = \prod_{\text{plaquettes } p} w(C_p)$$

$\mathcal{C}_p$			
$w(\mathcal{C}_p)$	$e^{-\Delta\tau J/4}$	$e^{\Delta\tau J/4} ch(\Delta\tau J/2)$	$e^{\Delta\tau J/4} sh(-\Delta\tau J/2)$

- Worldline QMC = Sampling over all (important) worldline configurations
  - According to the above weight
  - Try to generate a new configuration from a given one
- Already 2 problems arise …

Intermezzo 1 : Stochastic Series Expansion (SSE)

- Alternative approach : Expansion in inverse temperature (*Sandvik*, 1992)

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

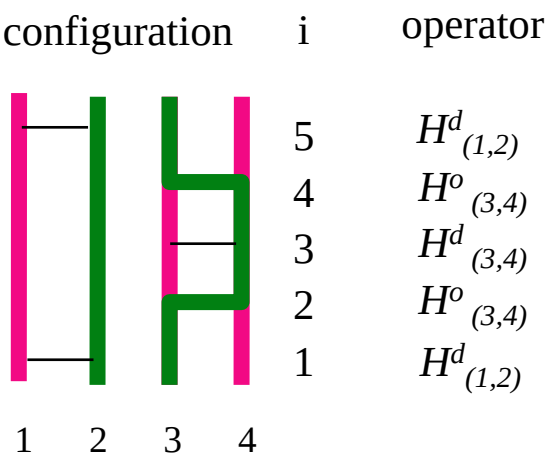
$$= \sum_{n=0}^{\infty} \frac{\beta^n}{n!} \sum_{|\alpha\rangle} \sum_{(b_1,\ldots,b_n)} \langle \alpha | \prod_{i=1}^{n} (-H_{b_i}) | \alpha \rangle$$

$$H = \sum_{b=(i,j)} H_b$$

- Using the bond Hamiltonians

$$H^o_{(i,j)} = \frac{J_{xy}}{2}(S^+_i S^-_j + S^-_i S^+_j)$$

$$H^d_{(i,j)} = J_z S^z_i S^z_j - \frac{h}{z}\left(S^z_i + S^z_j\right)$$



- Similar to path integral approach
- (Minor) difference in the treatment of diagonal terms

1st solved problem : the continuous time limit

- Systematic error due to finite value of $\Delta\tau$ (« Trotter error »)
  - Need to perform an extrapolation to $\Delta\tau \to 0$ from simulations with different values of $\Delta\tau$ (or Trotter number *M*)

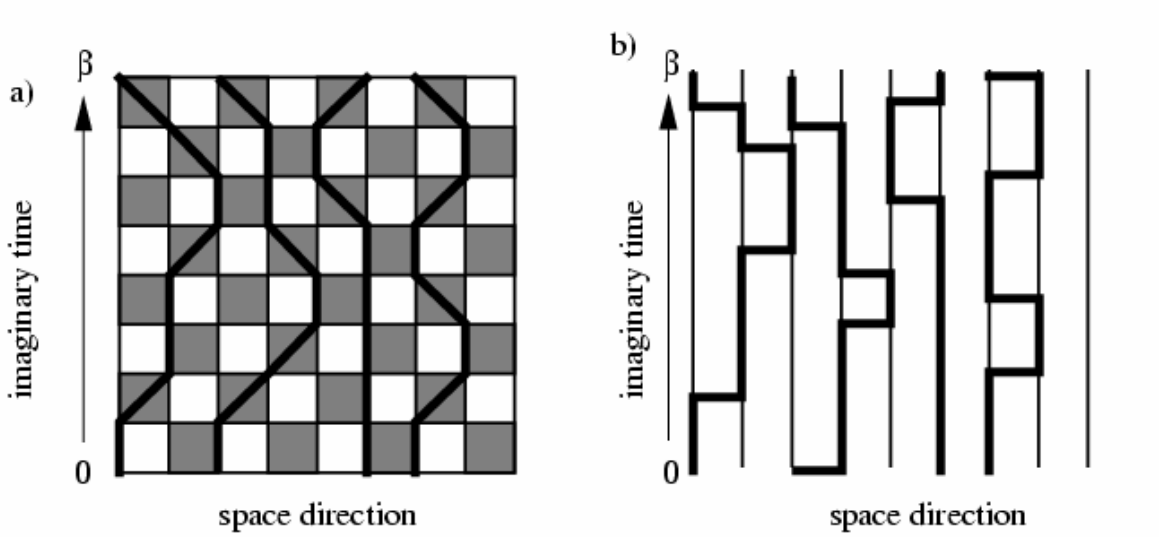
- The limit $\Delta\tau \to 0$ can be taken directly in the construction of the algorithm !

(*Prokof'ev et al.*, 1996)

  - Number of changes

$$N_c = M\frac{\Delta\tau J}{2} \to \frac{\beta J}{2}$$

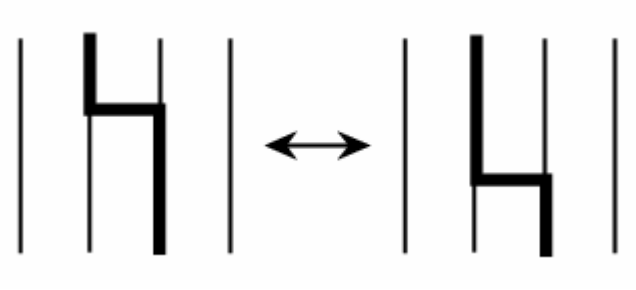
stays finite as $\Delta\tau \to 0$



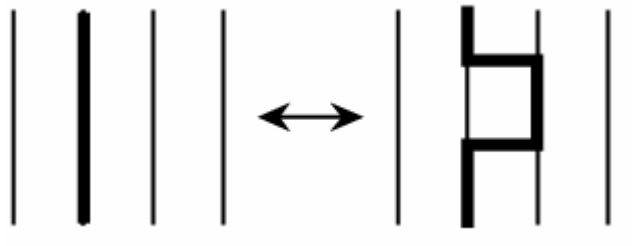
- Different computational approach:
  - Discrete time : store configuration at *all* time steps
  - Continuous time : store times at which configuration *changes* (+ initial state)

2nd solved problem solved

- Shift a kink



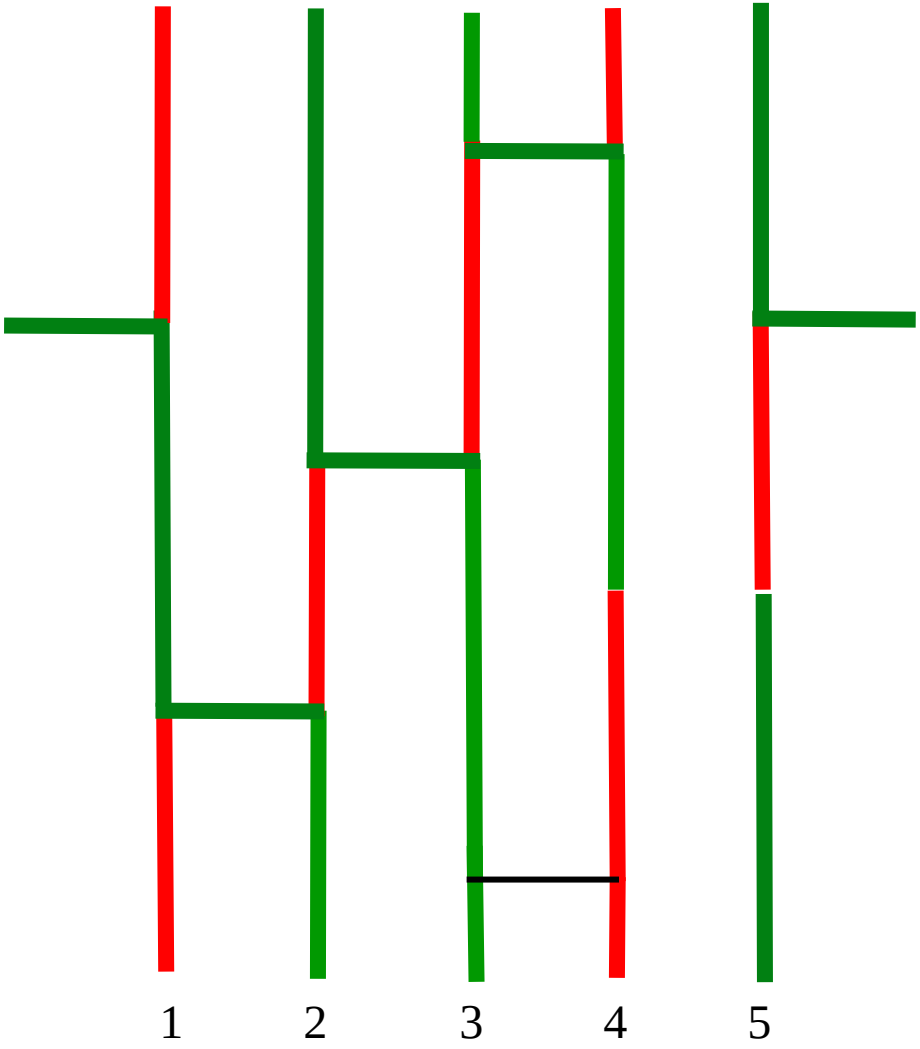
- Insert or remove two kinks (kink-antikink pair creation process)



- Problems with local updates
  - Restricted to canonical ensemble
  - No change of magnetization, particle number, winding number
  - Critical slowing down
- Solution for classical Monte Carlo was *cluster algorithms*
- Generalization to quantum case is possible !
  - Loop, Directed loops, Worm algorithms

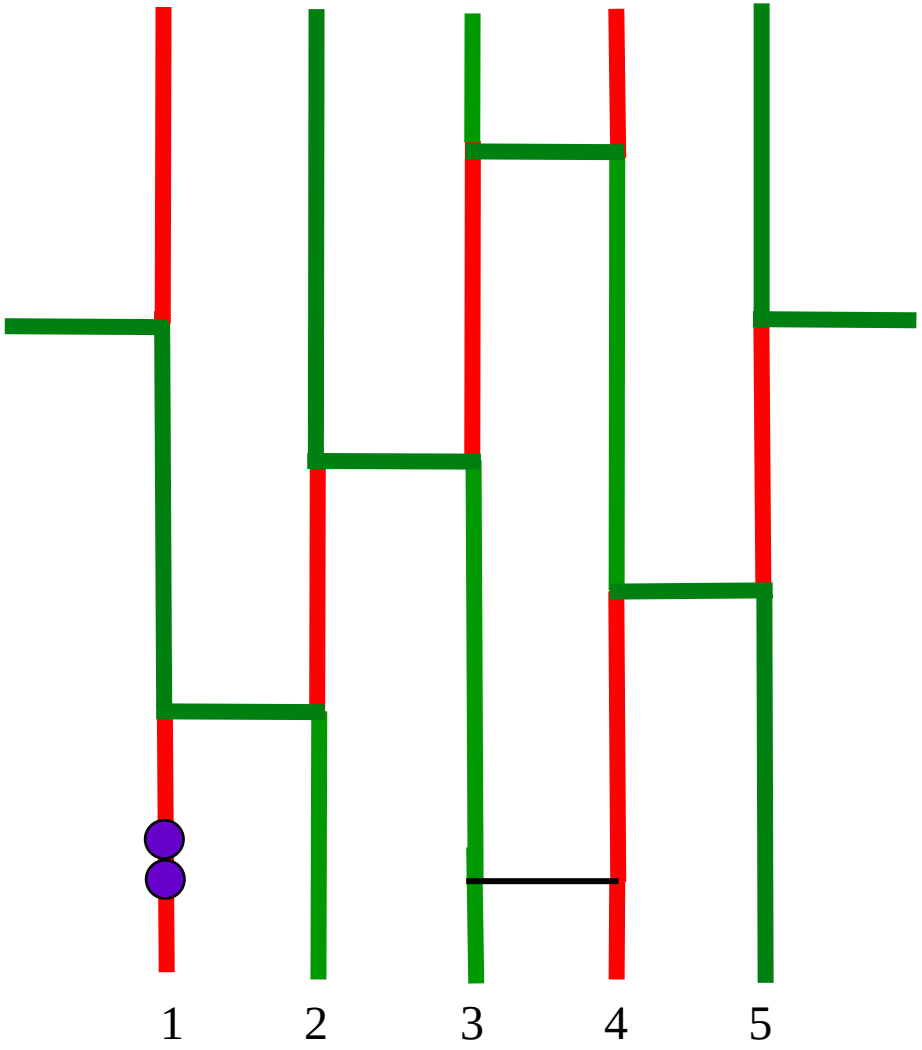
Worm algorithm : intuitive view

- Q : How to update non-locally this configuration ?



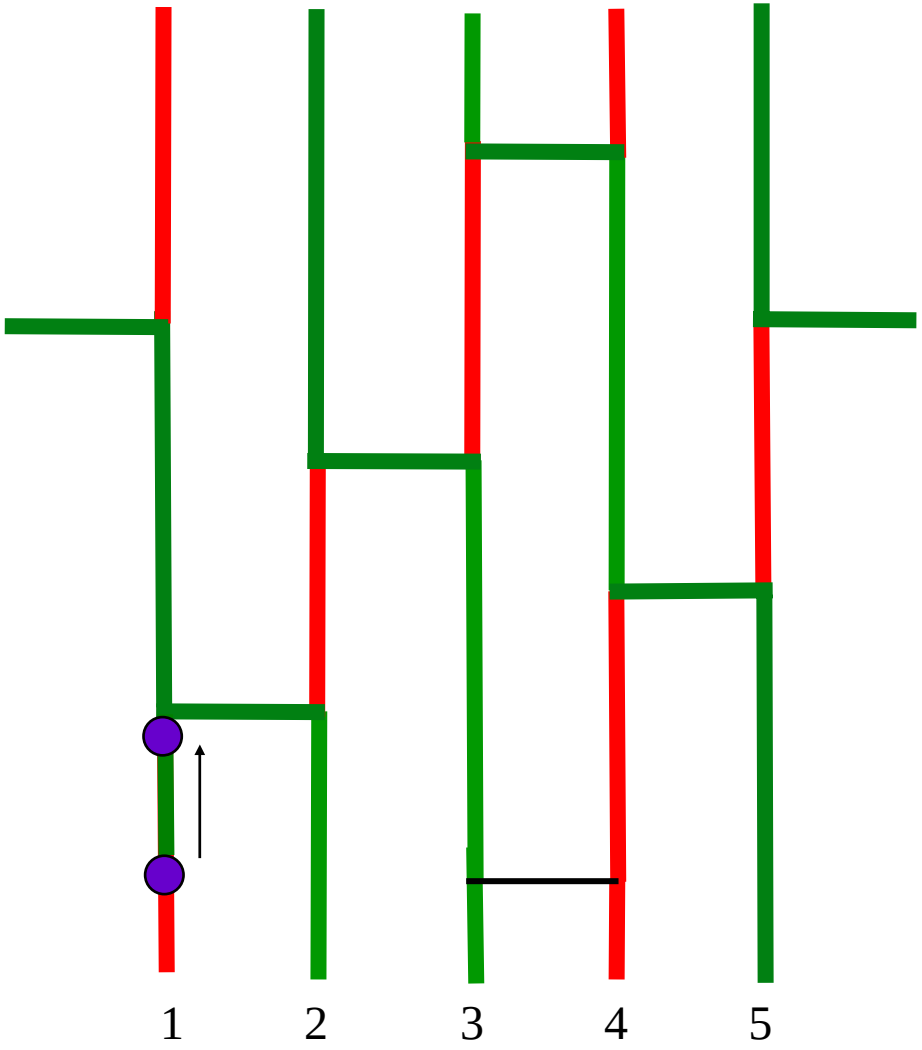
(SSE Representation)

Worm algorithm : intuitive view



1) Insert a worm (head and tail)

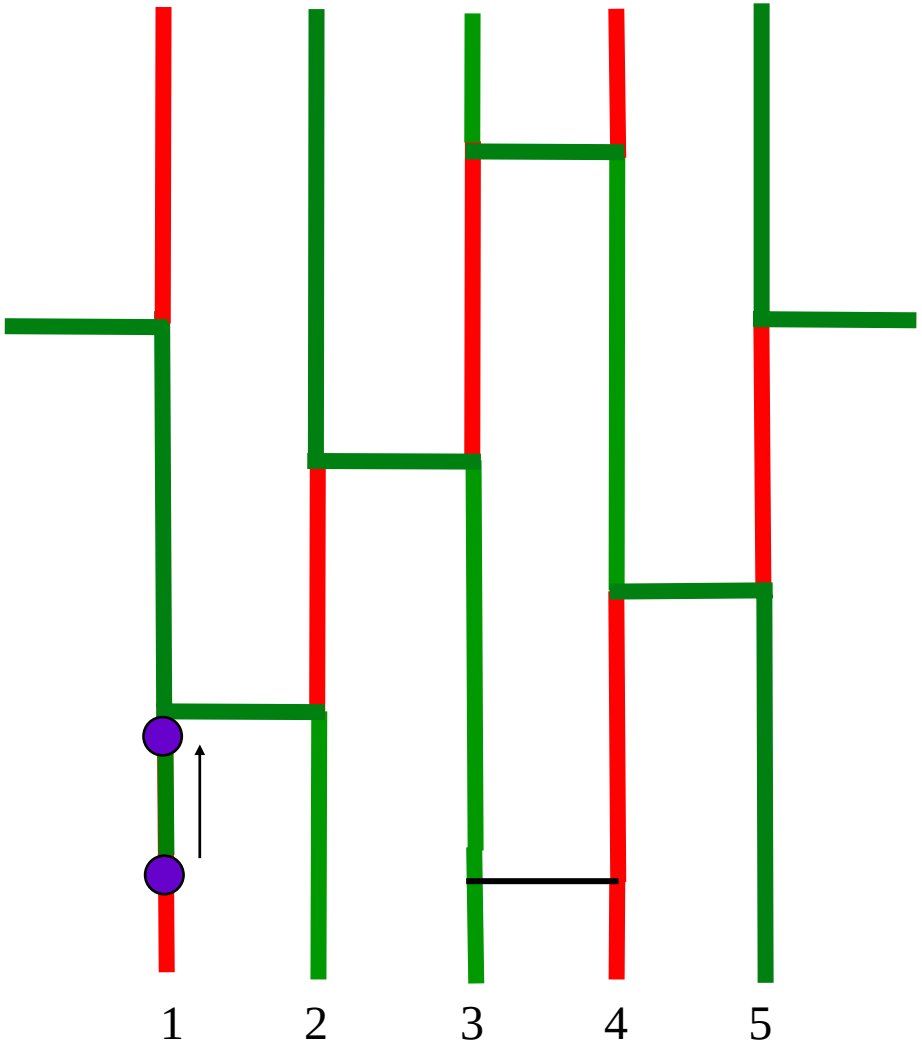
Worm algorithm : intuitive view



1) Insert a worm (head and tail)

2) Move the worm head – Modify locally the configuration on the fly

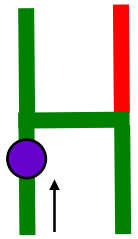
Worm algorithm : intuitive view

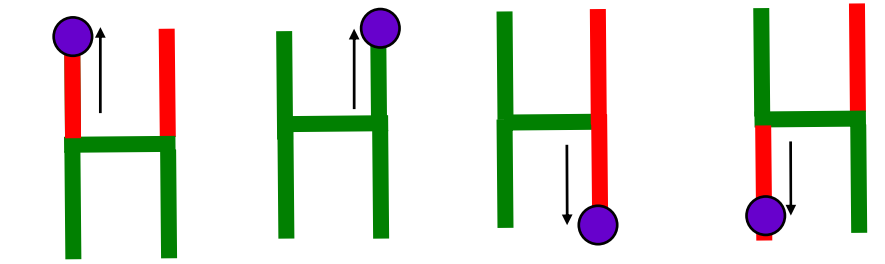


1) Insert a worm (head and tail)

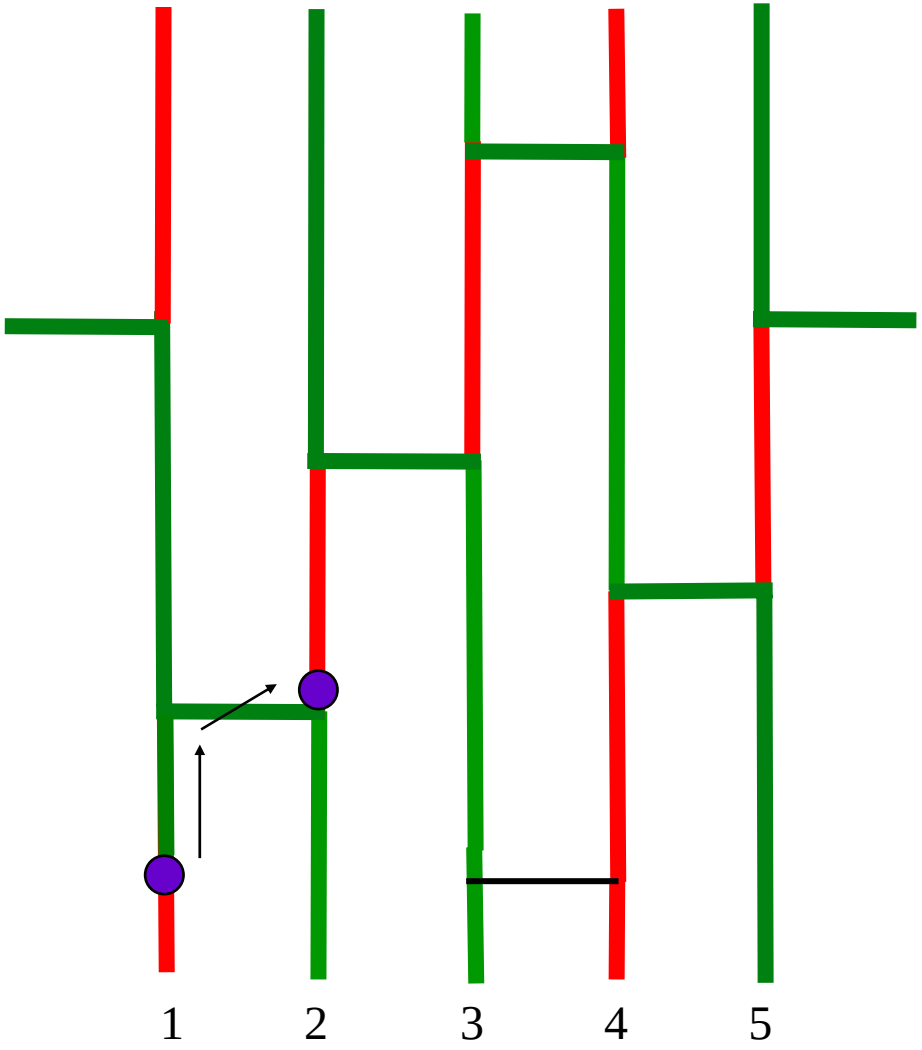
2) Move the worm head – Modify locally the configuration on the fly

3) At every “vertex” : Choose an exit





Worm algorithm : intuitive view



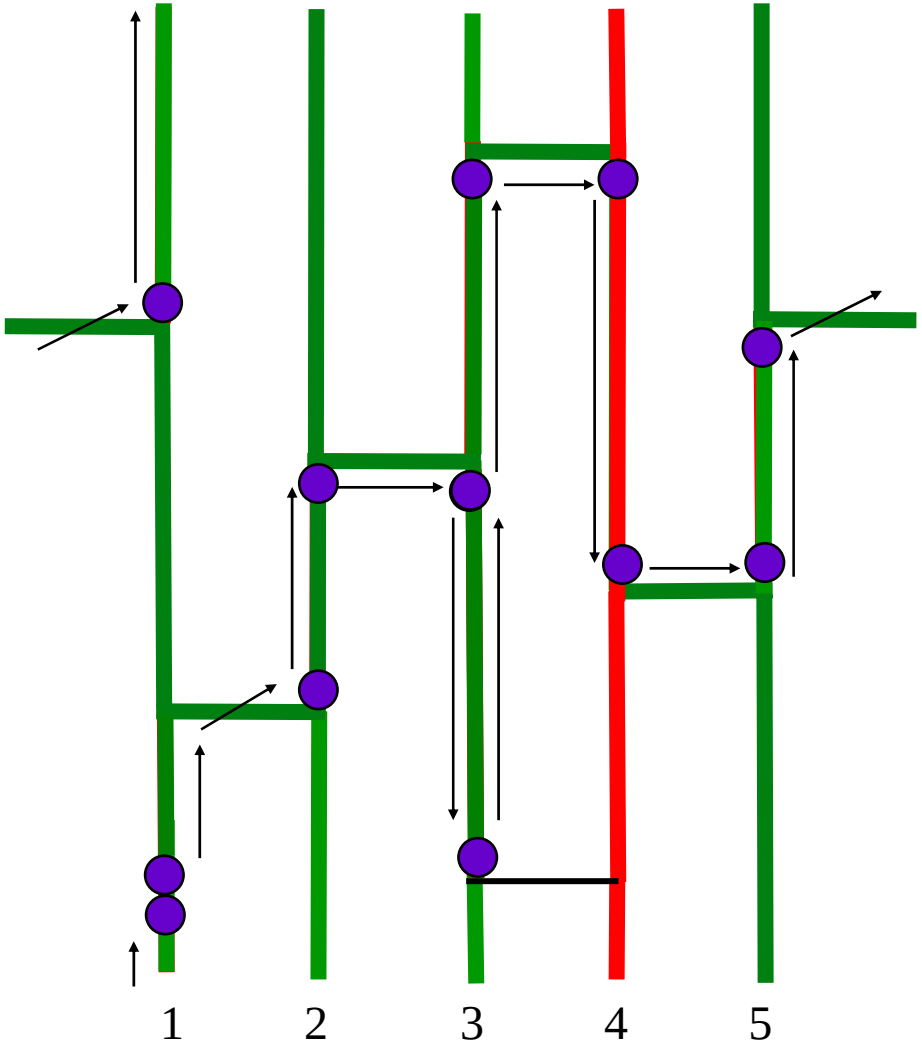
1) Insert a worm (head and tail)

2) Move the worm head – Modify locally the configuration on the fly

3) At every "vertex" : Choose an exit

"straight", "jump", "turn", "bounce"

Worm algorithm : intuitive view



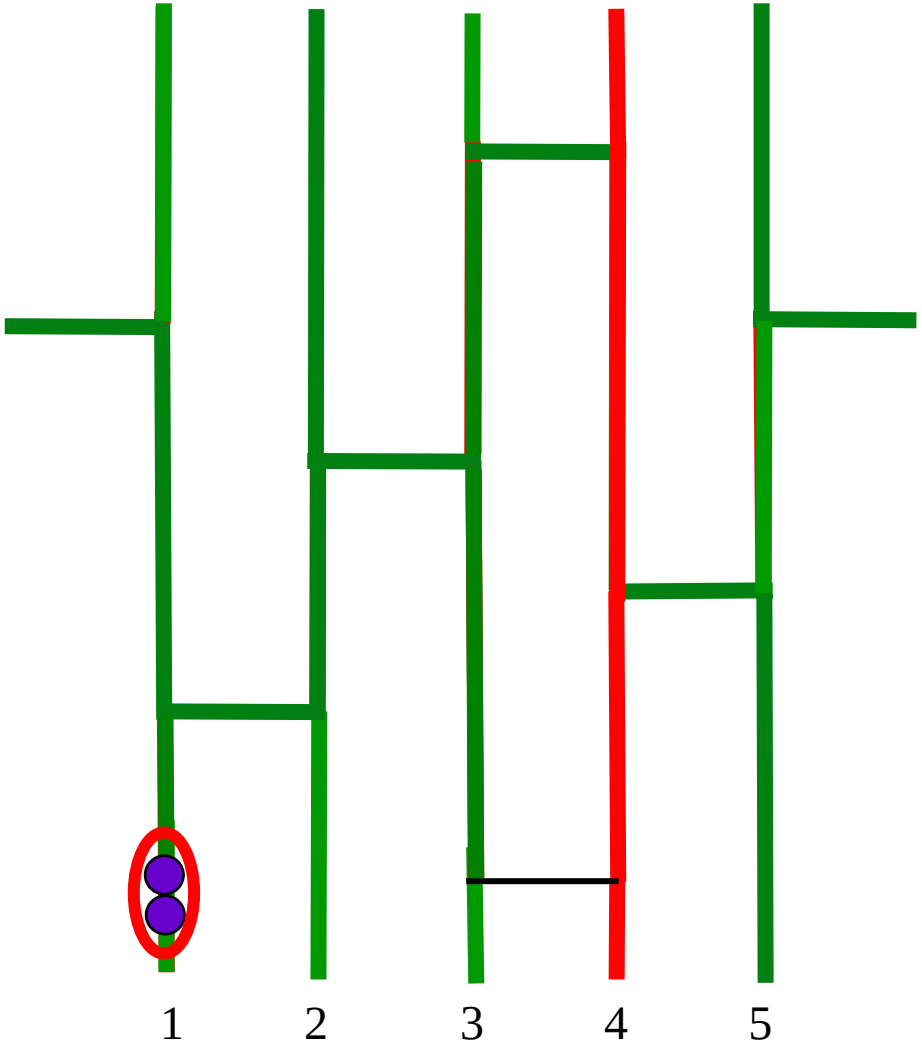
1) Insert a worm (head and tail)

2) Move the worm head – Modify locally the configuration on the fly

3) At every "vertex" : Choose an exit

"straight", "jump", "turn", "bounce"

Worm algorithm : intuitive view



1) Insert a worm (head and tail)

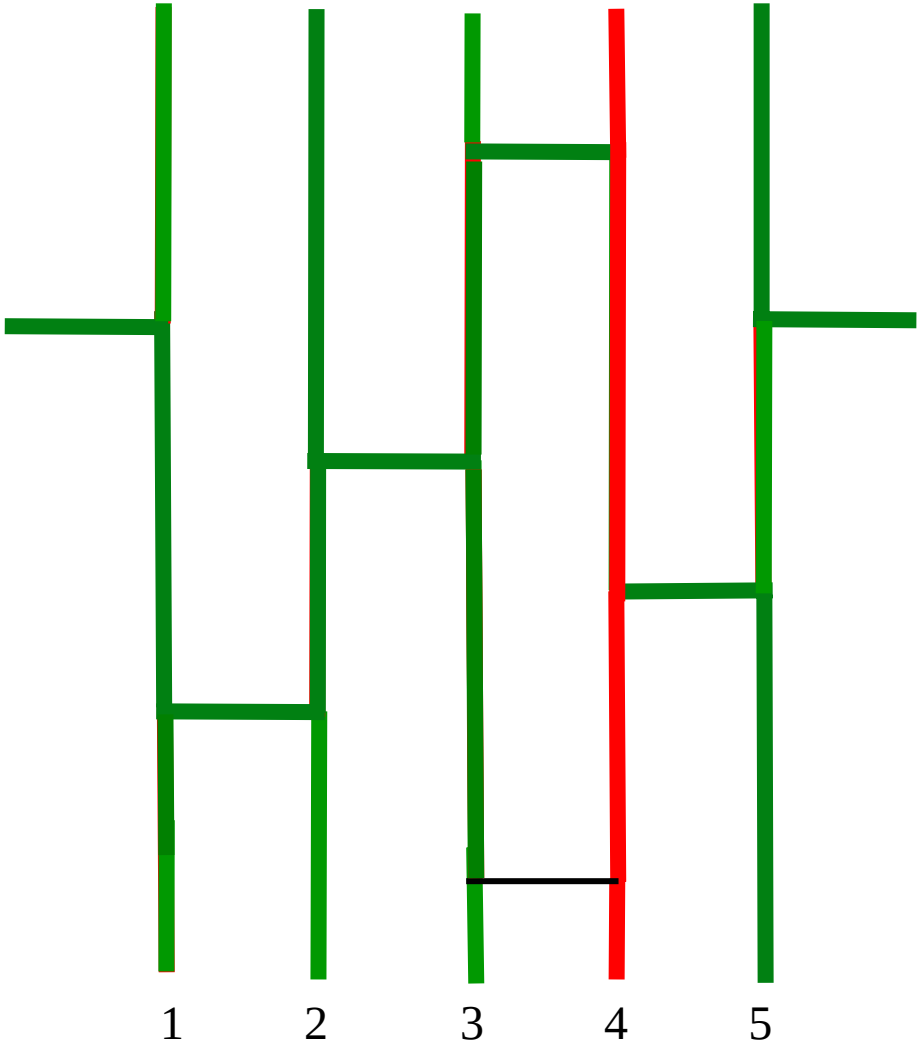
2) Move the worm head – Modify locally the configuration on the fly

3) At every "vertex" : Choose an exit

"straight", "jump", "turn", "bounce"

4) Annihilate worms

Worm algorithm : intuitive view



1) Insert a worm (head and tail)

2) Move the worm head – Modify locally the configuration on the fly

3) At every “vertex” : Choose an exit

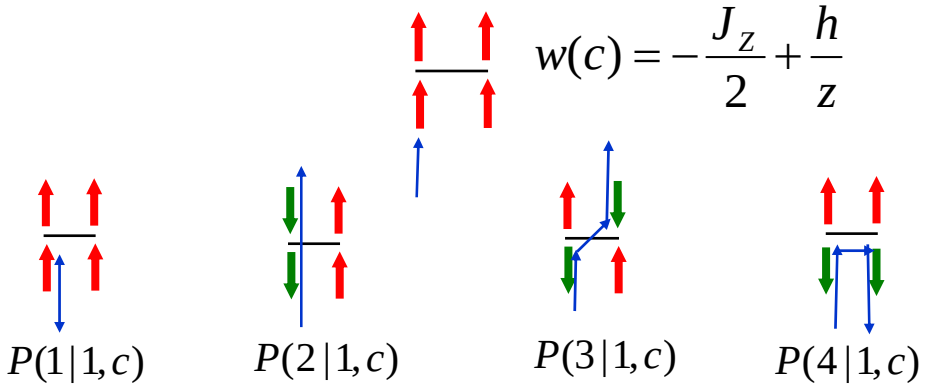
“straight”, “jump”, “turn”, “bounce”

4) Annihilate worms

NEW CONFIGURATION

Intermezzo 2 : How to choose the worm's moves ?

- *Given a "vertex" (plaquette) configuration and an entrance leg, where to exit ?*
- Consider exit leg e, given entrance leg i at a vertex in configuration c – weight $w(c)$
  - Assign this path a probability $P(e \mid i,c)$
  - Sum over all paths must equal unity $\sum_e P(e \mid i,c) = 1$



  - Choose exit leg e with probability $P(e \mid i,c)$ leading to configuration $\bar{c}$ – weight w($\bar{c}$)
- Consider the reverted path $P(i \mid e,\bar{c})$, leading back to c
  - Impose Local detailed balance $P(e \mid i,c)w(c) = P(i \mid e,\bar{c})w(\bar{c})$

Quantum cluster algorithms

- Tiny differences in how the worm exactly moves lead to different algorithms
  - Worm algorithm : random Metropolis move
  - Directed loop algorithm : locally improved « clever » move
  - Loop algorithm : deterministic move
- Difference in Representation
  - SSE : usually slighlty faster
  - Path Integral (PI) : good when large diagonal terms
- Codes in ALPS : Which qlgorithm to choose ?
  - **`looper`** = Loop algorithm in PI/SSE : For models with spin inversion symmetry

    e.g. : Heisenberg, XY, XXZ model wihout field
  - **`dirloop-sse`** = Directed loops in SSE : Models without inversion symmetry (most genera

    e.g. : models with field, boson models etc
  - **`worm`** = Worm algorithm in PI : Models with large diagonal elements

    e.g. : large Ising anisotropy
- All these algorithms give accurate results for large systems …

Intermezzo 3 : The sign problem

- In mapping of quantum to classical system

$$\langle A \rangle = \frac{\mathrm{Tr}\left[A \exp(-\beta H)\right]}{\mathrm{Tr}\left[\exp(-\beta H)\right]} = \frac{\sum_i A_i p_i}{\sum_i p_i}$$

- « Sign problem » if some of the $p_i<0$
  - Cannot interpret p_i as probabilities
  - Appears in simulation of fermions and frustrated magnets
- "Way out" : Perform simulations using $|p_i|$ and measure the sign :

$$\langle A \rangle = \frac{\sum_i A_i p_i}{\sum_i p_i} = \frac{\sum_i A_i \,\mathrm{sgn}\, p_i |p_i| \Big/ \sum_i |p_i|}{\sum_i \mathrm{sgn}\, p_i |p_i| \Big/ \sum_i |p_i|} \equiv \frac{\langle A\, Sign \rangle_{|p|}}{\langle Sign \rangle_{|p|}}$$

  - Sampling $Z_{|p|} = \sum_i |p_i|$

Intermezzo 3 : The sign problem

- The average sign becomes very small

$$\langle Sign \rangle_{|p|} = \frac{1}{Z_{|p|}} \sum_i \mathrm{sgn}\ p_i |p_i| = \frac{Z}{Z_{|p|}} = e^{-\beta V \Delta f}$$

  - Both in system size and inverse temperature
  - This is the origin of the sign problem !

- The error of the sign:

$$\frac{\Delta Sign}{\langle Sign \rangle_{|p|}} = \frac{\sqrt{\langle Sign^2 \rangle_{|p|} - \langle Sign \rangle_{|p|}^2}}{\sqrt{N} \langle Sign \rangle_{|p|}} \approx \frac{\sqrt{\langle 1 \rangle_{|p|}}}{\sqrt{N} \langle Sign \rangle_{|p|}} = \frac{e^{\beta V \Delta f}}{\sqrt{N}}$$

  - Need of the order $N = \exp(2\beta V \Delta f)$ measurements for sufficient accuracy
  - Similar problem occurs for the observables
  - Exponential growth ! Impossible to treat large systems or low temperatures

In practice, what can I do ?

- Simulate large samples of lattice quantum spin models
  - For non-frustrated models (Note : Diagonal (Ising) frustration is OK)
  - This includes most Heisenberg-like models
  - XY, Ising anisotropy, magnetic field, all values of S, single-ion anisotropy, …
  - *Frustrated models are possible : but quickly sign problem arises*
  - In all dimensions on all lattices *(one can define its own specific lattice)*
  - For all values of T *(including T→0)*
- Which quantities are accessible ?
  - A lot of observables : Energy, Specific heat, Susceptibilities, Structure factor, Spin stiffness, Green functions …
  - In some cases : one can define its own observable
- How much time ?
  - Typically simulation time scales as Number of spins x Inverse Temperature
- For more details : see hands-on session …

Thanks !