Continuous Time SW Algorithm

Richard C. Brower

August 28, 2024

1 Introduction

The paper (https://arxiv.org/abs/cond-mat/9802104v1 on **Application of a continuous time cluster algorithm to the Two-dimensional Random Quantum Ising Ferromagnet** by Heiko Rieger and Naoki Kawashima has the correct formulation but lack details for write code

The paper does the Swendsen Wang update in the limit of zero time lattice spacing, a_t, for any Ising Hamiltonian on a spacial graph in 4 cycles illustrated in Fig.1.1

The partition function

$$Z = \sum_{s_i=\pm 1} e^{K^0_{t,i} s_{t,i} s_{t+1,i} + \sum_{\langle i,j \rangle} \widetilde{K}^{\perp}_{ij} s_{t,i} s_{t,j}} \tag{1.1}$$

becomes

$$Z = Tr[e^{-\beta \hat{H}}] \quad \text{where} \quad \hat{H} = -\sum_i \Gamma_i \sigma^x_i - \sum_{\langle i,j \rangle} J_{ij} \sigma^z_i \sigma^z_j \tag{1.2}$$

in the limit $a_t \to 0$ with $\widetilde{K}^{\perp}_{ij} = a_t J_{ij}$ and $e^{-2a_t K_i} = \tanh(a_t \Gamma_i)$.

2 Formalism

The state space is A in figure. It is a sequence of broken bounds at $0 < t_i \leq \beta$ in vertical lines. The number of broken of bonds (i.e decays sequence) in time Δt is the Poisson

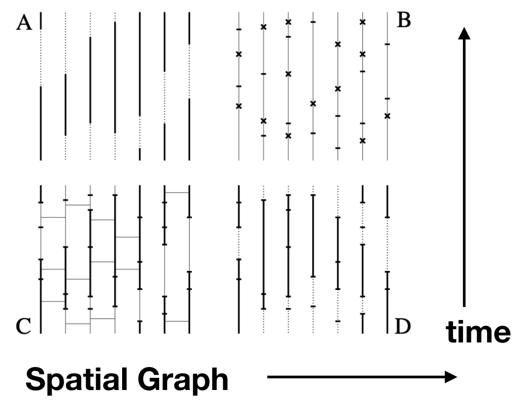
Continuous Time (aka Euclidian Hamiltonian) Cluster Monte Carlo*

Affine: $a_t \to 0$

$$S = -\sum_{t,i} K_i^0 s_{t,i} s_{t+1,i} - \sum_{\langle i,j \rangle} K_{ij}^{\perp} s_{t,i} s_{t,i} \qquad H = \sum_i \Gamma_i \sigma_i^x + \sum_{\langle i,j \rangle} \widetilde{K}_{ij} \sigma_i^z \sigma_j^z$$

$$K_{ij} = a_t \widetilde{K}_{ij} \quad , \quad e^{-2a_t K_i^0} = \tanh(a_t \Gamma_i)$$

- A state space (real value decay times)
- B Poisson Decays: $P(t) = \Gamma e^{-\Gamma t}$
- C. Spatial Percolation $P_{ij} = 1 - e^{-2\Delta t_{overlap} \widetilde{K}_{ij}}$
- D. SW Flip clusters for new state A



Pretty Easy to Program with Connected Components Graph algorithms:
Works for 1 + d Radial Quantization (Sphere) Ising/SUSY/Warped AdS etc

***See: 1998: H. Rieger, N. Kawashima**
Application of a continous time cluster algorithm to the Two-dimensional Random Quantum Ising Ferromagnet

Duality: $\cosh(2a_t K_i^0)\cosh(a_t \Gamma_i) = 1$

Figure 1.1: Slide from QuLAT meeting Jan 2024. I switch to Heiko Rieger and Naoki Kawashima notation: $J_{ij} = \widetilde{K}_{ij}$.

distribution $n = 0, 1, 2, \cdots, \infty$

$$p_n = \frac{\lambda^n e^{-\lambda}}{n!} \tag{2.1}$$

with $\lambda = \Delta t \Gamma$. https://en.wikipedia.org/wiki/Poisson_distribution This is equivalent to a decay rate $P(t) = \Gamma \exp[-t\Gamma]$, normalized to $\int_0^\infty P(t) = 1$ with mean lifetime:

$$\int_0^\infty dt\; t\; \Gamma e^{-t\Gamma} = 1/\Gamma \tag{2.2}$$

The probability of no decay in interval $[0, \Delta t]$ is

$$p_0 = \Gamma \int_{\Delta t}^\infty e^{-t\Gamma} = e^{-\Gamma \Delta t} \tag{2.3}$$

Theorem: The distribution of n decay time in an interval Δt for n -decay is a random series of times: $t_0 = 0 < t_1 < t_2 < \cdots < t_n < \Delta t$. The Poisson distribution is give by both by the time ordered sequence of decays in intervals $0 < \Delta t_i = t_i - t_{i-1} < \Delta t$ **and** the uniform distribution random times in $[0, \Delta t]$:

$$\begin{aligned} p_n &= \int_{t_{n-1}}^{\Delta t} dt_n \cdots \int_0^{\Delta t_2} dt_1 (\Gamma e^{-\Delta t_1 \Gamma}) \times \cdots \times (\Gamma e^{-\Delta t_n \Gamma}) \times (e^{-(\Delta t - [\Delta t_1 + \cdots + \Delta t_n])\Gamma}) \\ &= \int_{t_{n-1}}^{\Delta t} dt_n \cdots \int_0^{\Delta t_2} dt_1 \Gamma^n e^{-\Delta t \Gamma} = \frac{1}{n!} (\prod_i \int_0^{\Delta t} dt_i) \; \Gamma^n e^{-\Delta t \Gamma} = \frac{(\Delta t \Gamma)^n e^{-\Delta t \Gamma}}{n!} \end{aligned} \tag{2.4}$$

This allows us to introduce cut links as sequence of decays in any interval until we run out of space! Very efficient algorithm going for stage A to B.

The step form B to C constructs new cluster with percolation in to neighboring links with probability

$$p_{\langle i,j \rangle} = e^{-2 J_{ij} \Delta t_{ij}} \tag{2.5}$$

where Δt_{ij} is the overlap time interval between equal spin segments on the $\langle i, j \rangle$ edges.

3 Data Structures

The state is keep in a basic Data struture, called Rails.The 'Langrangian" phase space for the partion funtion $Z = Tr[e^{-\beta H}]$ are strips of Euclidean time length $DetlaT = \beta$ with cut-bonds at each spacial site $x = 0 \cdots L - 1$. The state is keep in a basic Data struture, called Rails.

```
struct param{
  double  DeltaT;
  double Gamma;
  int N;
  vector<double>  time[L];
  vector<int>  spin[L];
  vector<int> clusterNumber[L];
};
```

Rail for each $x = 0, \cdots, L - 1$ is a list of times and corresponding spins

$$t_i = time[x][i] \quad , \quad s_i = spin[x][i] \in \{t_{I+1}, t_i\} \tag{3.1}$$

The spins are assigne to each interval:

$$t_0 = 0 < t_1 < t_2 < \cdot < t_n < t_{n+1} = T \tag{3.2}$$

between the cuts at t_i for $i = 0, .., n$. **Perodic B.C are enforced** "removing the virtual cut at $t = 0$ and $t = T$ are glued together by joining the first and last strip with single spin

$$s_0 = s_n \in \{T, t_n\} \cup \{t_1, 0\} \tag{3.3}$$

If you don't do this you can have fixed Dirichlet boundary condition by fixing s_0, s_n or Neumann by letting theme be independent variable.

All the steps A,B,C and D update in place this data structure!