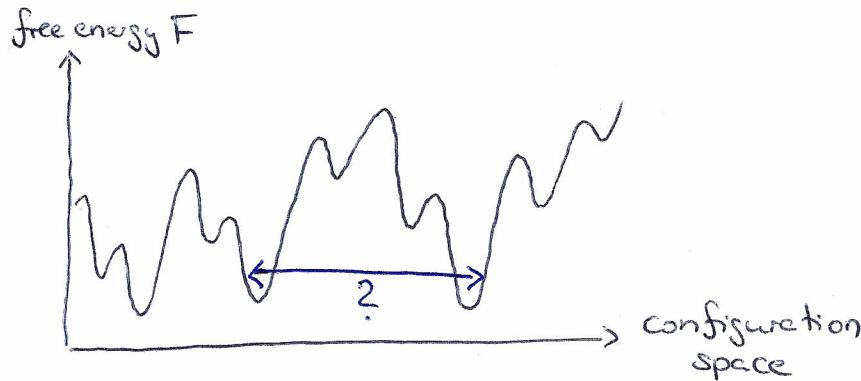


Extended ensemble Simulations

The problem: Thermal equilibrium in systems with rough (free) energy landscape is hugely suppressed.



example systems:

- frustrated magnets
- proteins
- any system close to phase transition

The goal: Improve equilibration in numerical simulations.
How can this be achieved?

Every Monte Carlo step consists of two parts:

- 1) Suggest a configuration update
→ improvements: non-local updates (clusters, worms,...)
- 2) accept/reject this update
→ improvements: extended statistical ensembles

The algorithms:

- 1) Wang-Landau algorithm
- 2) Ensemble optimization
- 3) Parallel tempering

Simulation of Markov chains

Typically, we think of Monte Carlo simulations as sampling configurations in some high-dimensional configuration space

$$C_1 \rightarrow C_2 \rightarrow C_3 \rightarrow \dots \rightarrow C_i \rightarrow C_{i+1} \rightarrow \dots$$

A widely used algorithm to generate such a Markov chain is the Metropolis algorithm:

- propose a (small) change to a configuration

$$\begin{array}{c} \uparrow \uparrow \uparrow \uparrow \uparrow \uparrow \\ C_i \end{array} \longrightarrow \begin{array}{c} \uparrow \uparrow \uparrow \downarrow \uparrow \uparrow \\ C_{i+1} \end{array}$$

- accept/reject the update with probability

$$P(C_i \rightarrow C_{i+1}) = \min \left(1, \frac{\omega(C_{i+1})}{\omega(C_i)} \right)$$

↑
statistical weights

- Common choice for the statistical weights/ensemble is the canonical one:

$$\omega(C_i) = \omega(E_i) = \exp(-\beta E_i)$$

↑
energy $E_i = H(C_i)$.

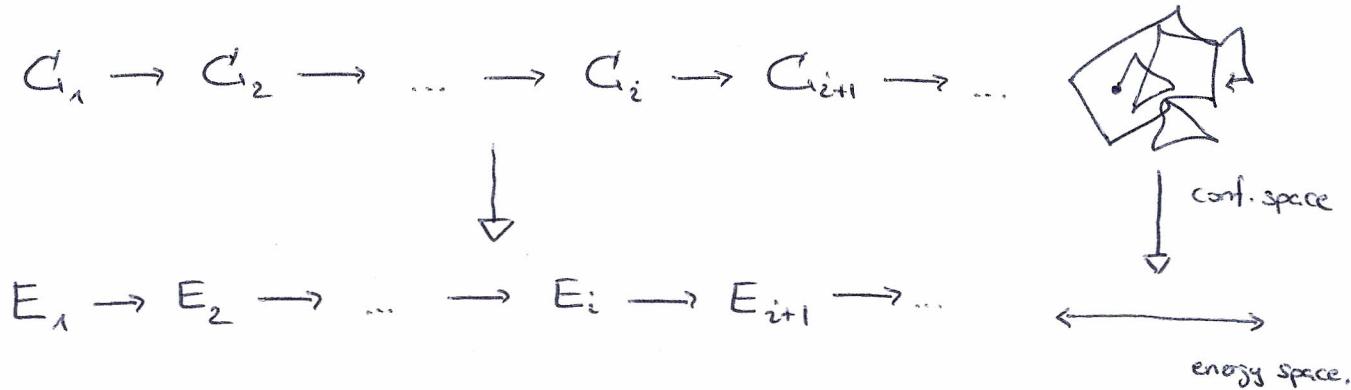
But we could have chosen all sorts of other weights
 - we will see some of these other choices later on.
 Here:

$$P(C_i \rightarrow C_{i+1}) = \min \left(1, \exp(-\beta(E_{i+1} - E_i)) \right).$$

Markov chains (cont'd)

In making our decision about whether or not we accept an update solely based on the energy difference of the two configurations, we neglect a lot of information.

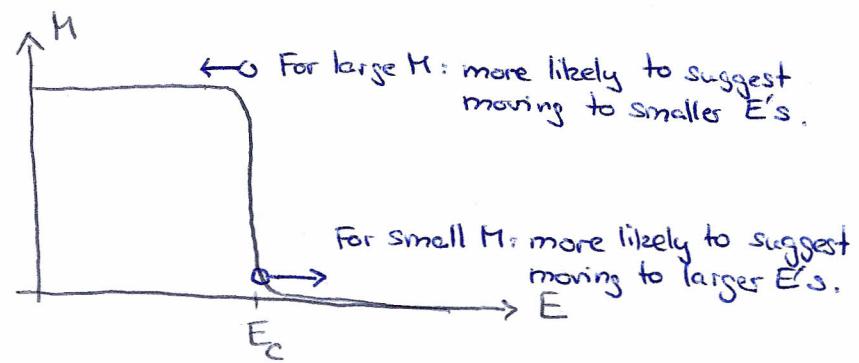
In particular, we project the random walk in configuration space onto one in energy space:



Configuration space: Random walk is biased (by stat. ensemble), but Markovian.

Energy space: Random walk is non-Markovian, 'memory' is stored in configuration space.

Example: Ising model



The Wang-Landau algorithm

Goal: Calculate density of states, $g(E)$, by iteratively approximating weights, $\omega(E) = 1/g(E)$, and sample a flat histogram, $h(E)$.

The algorithm:

- Initialize $h(E) = 0$ and $g(E) = 1$ for all energies E .
- Set 'modification factor' $f = f_0 = e$.
- Start from a random configuration C_1 .

- • Propose configuration update $C_1 \rightarrow C_2$ (e.g. a spin flip), with corresponding energies $E_1 = H(C_1)$, $E_2 = H(C_2)$.

- Accept update with probability

$$p(C_1 \rightarrow C_2) = \min\left(1, \frac{g(E_1)}{g(E_2)}\right),$$

if accepted $E_1 \leftarrow E_2$, $C_1 \leftarrow C_2$.

- loop
main
- Update density of states, $g(E_1) \leftarrow g(E_1) \cdot f$ and histogram, $h(E_1) \leftarrow h(E_1) + 1$.

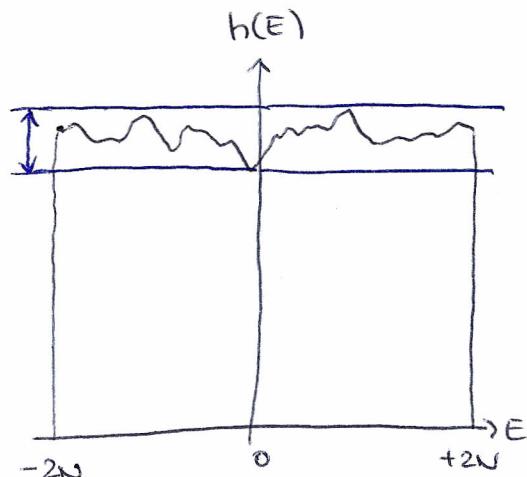
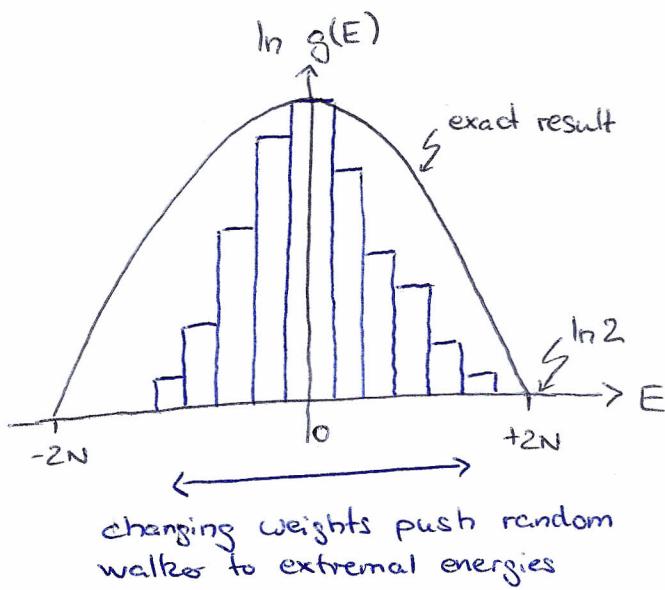
- If histogram is "flat", update $f \leftarrow \sqrt{f}$, and reset $h(E) = 0$ for all energies.

- Iterate until f becomes smaller than some threshold, e.g. $f_{\text{final}} = 1.000\ 001$.

The Wang-Landau algorithm (cont'd)

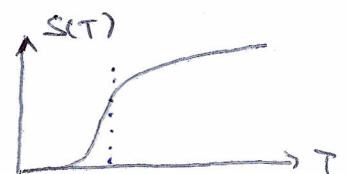
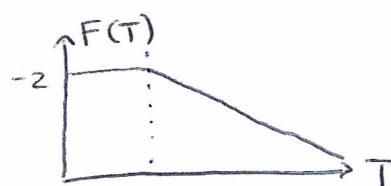
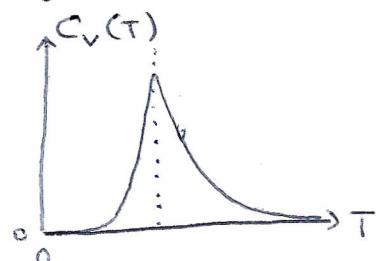
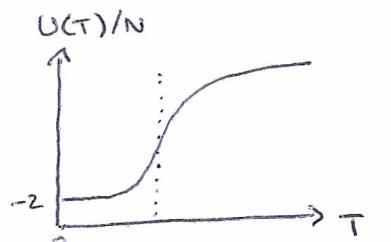
Example: 2D Ising model on a square lattice with N sites.

Wang-Landau iterations



With estimated density of states calculate:

- $Z = \sum_E g(E) \cdot e^{-\beta E}$
- $U(T) = \frac{1}{Z} \cdot \sum_E E \cdot g(E) e^{-\beta E} = \langle E \rangle_T$
- $C_V(T) = \frac{\partial U}{\partial T} = \frac{\langle E^2 \rangle_T - \langle E \rangle_T^2}{T^2}$
- $F(T) = -kT \cdot \ln Z$
- $S(T) = \frac{U(T) - F(T)}{T}$



Limitations of flat-histogram sampling

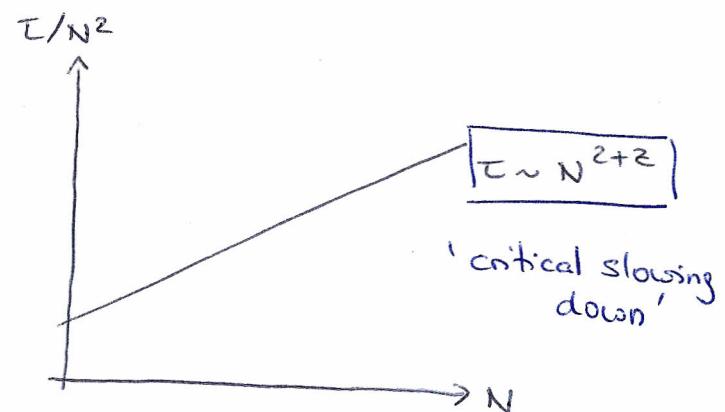
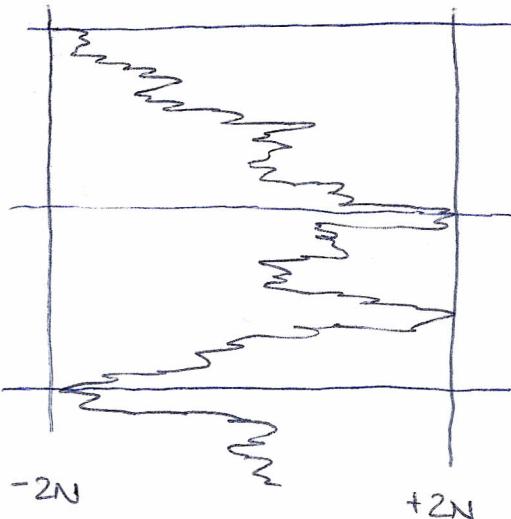
Is sampling a flat histogram really the best we can do to explore a system with a rough energy landscape?

No - one reason we have just seen: Not all temperature / energy regimes are the same. So, why should this not be reflected in the histogram / statistical ensemble?

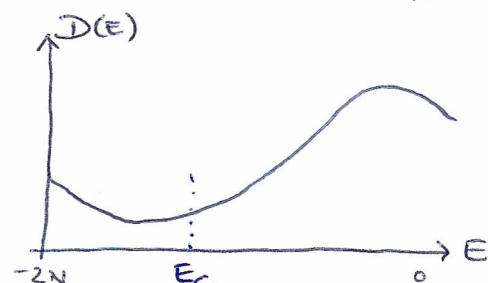
A more complex answer: Carefully study the random walk in energy (temperature) space as opposed to the random walk in configuration space:

Something we have already seen: Markovian vs. non-Markovian behavior.

A further manifestation of the non-Markovian behavior is found in the scaling behavior of round-trip times.



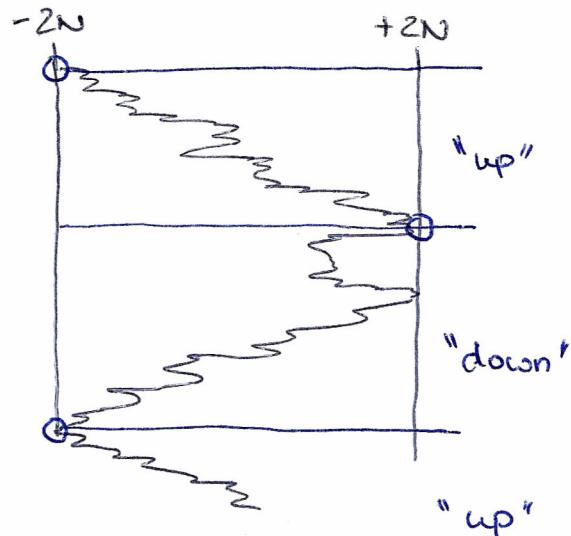
What originates this slowing down? A (local) diffusivity that is modulated in (energy) space:



Ensemble optimization

(7)

Can we further improve the simulated statistical ensemble?
To achieve this we want to minimize the round-trip times between extremal energies.

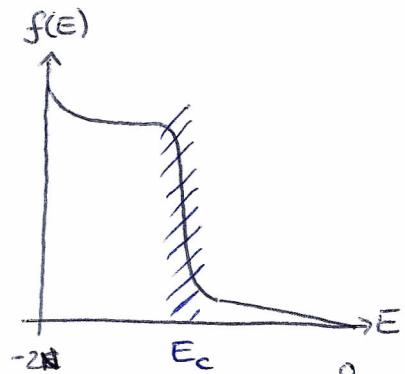


Split random walk into two:
Steady-state currents from $E_{\min} \rightarrow E_{\max}$ and $E_{\max} \rightarrow E_{\min}$
which exactly cancel
(since we are in total equilibrium).

These steady-state currents can be estimated

$$j = D(E) \cdot h_w(E) \cdot \frac{df}{dE}$$

↑ ↑ ↑
 local histogram derivative
 diffusivity of fraction
 $f(E) = \frac{h_w^+(E)}{h_w(E)}$



Minimize round-trip time by maximizing this current:

$$\tau \propto \frac{1}{j} = \int_{E_{\min}}^{E_{\max}} dE \left(\frac{1}{D(E) \cdot h_w(E)} + \lambda h_w(E) \right)$$

Lagrange multiplier
 $h_w(E)$ must remain normalized

The 'optimal' histogram thus becomes:

$$h_w(E) \propto \frac{1}{\sqrt{D(E)}} \propto \frac{df}{dE}$$

Ensemble optimization (cont'd)

Feedback algorithm:

- Start with some trial weights $\omega(E)$, e.g. from Wang-Landau algorithm.

→ Initialize $h(E) = 0$, $h^+(E) = 0$, $h^-(E) = 0$

- Run simulations with acceptance rates

$$P(C_1 \rightarrow C_2) = \min\left(1, \frac{\omega(C_2)}{\omega(C_1)}\right) = \min\left(1, \frac{\omega(E_2)}{\omega(E_1)}\right).$$

and record the histograms $h(E)$, $h^+(E)$ and $h^-(E)$.

- Estimate local diffusivity

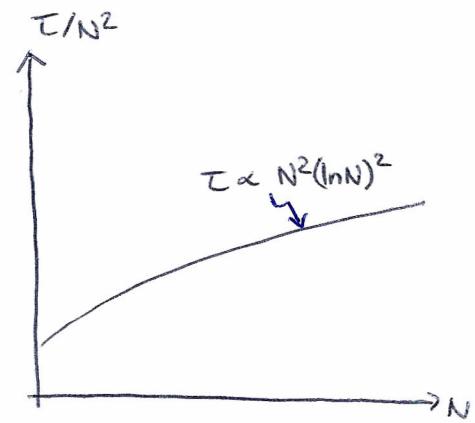
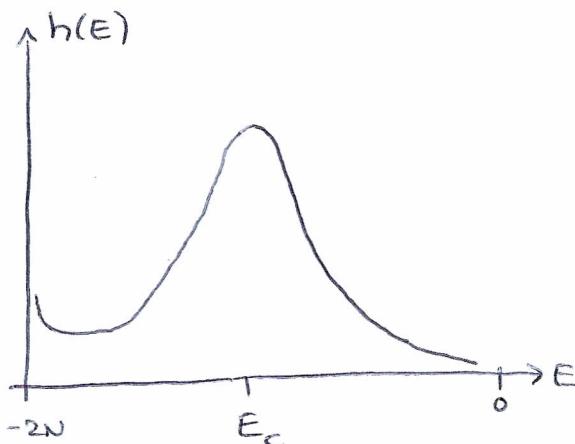
$$D(E) \propto \frac{1}{h(E) \frac{df}{dE}}, \text{ where } f(E) = \frac{h^+(E)}{h(E)}$$

- Optimize weights as

$$\omega(E) \leftarrow \omega(E) \cdot \sqrt{\frac{1}{h(E)} \cdot \frac{df}{dE}}$$

- Continue feedback until weights converge

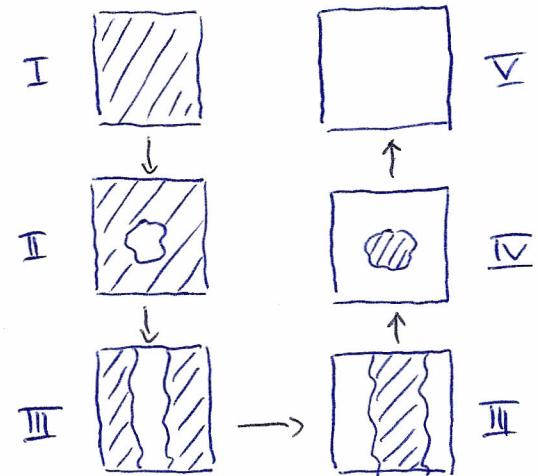
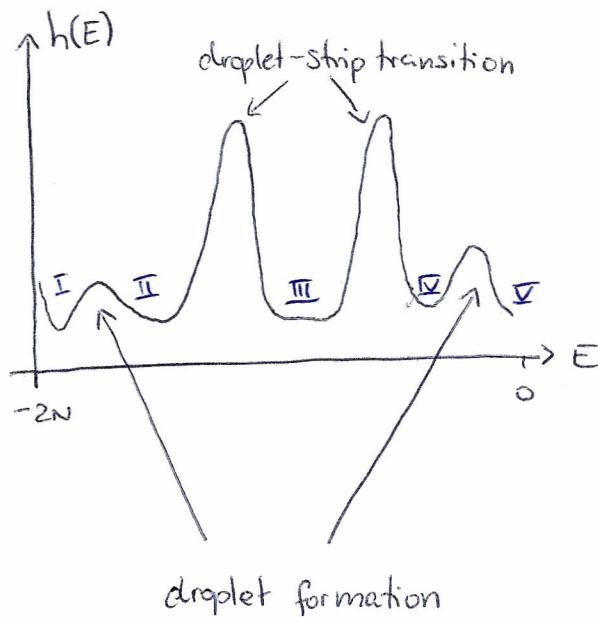
Example: 2D Ising model



Ensemble optimization (cont'd)

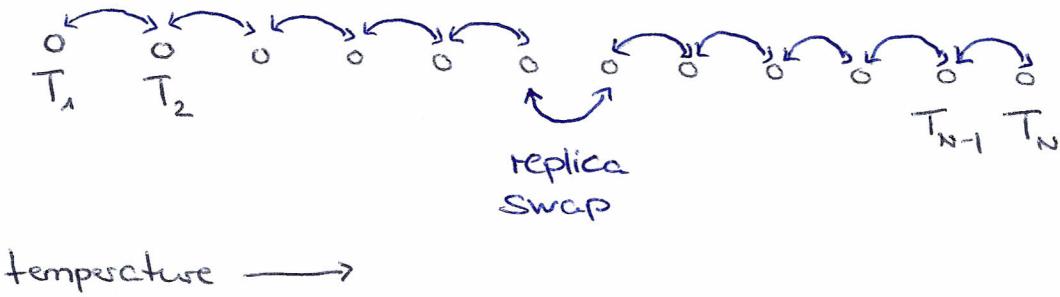
Example: 2D Potts model (first-order transition)

$$\mathcal{H} = - \sum_{\langle ij \rangle} \delta(\beta_i, \beta_j) \quad \beta_i \in \{1, 2, \dots, Q\}$$



The parallel tempering algorithm

Goal: Improve thermal equilibration by simulating multiple replicas of the system at various temperatures and swapping replicas between adjacent temperatures.

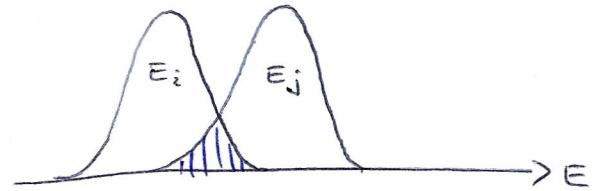


Accept replica swap with probability

$$p(E_i, T_i \rightarrow E_j, T_j) = \min(1, \exp(\Delta\beta \Delta E))$$

$$\Delta\beta = \beta_j - \beta_i = \frac{1}{T_j} - \frac{1}{T_i}$$

$$\Delta E = E_j - E_i$$



From perspective of given temperature:

Replica swaps (may) result in global configuration update.

From perspective of given replica:

Replica swaps result in random walk in T-space.

Parallel tempering (cont'd)

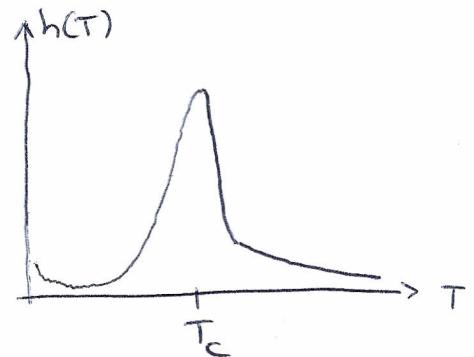
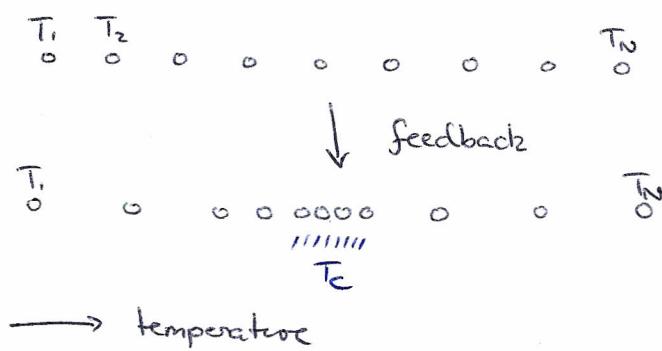
(11)

How do you choose the set of temperature points?

Again, we want to shift (numerical) resources towards the bottleneck of the simulation, i.e. those temperature regimes where the (local) diffusivity (in temperature space) is suppressed.

$$h(T) \propto \frac{1}{\sqrt{D(T)}} \quad , \text{ where } h(T) \propto \frac{1}{\Delta T}$$

Example: 2D Ising model



Note: This is not the same as a uniform acceptance rates for Swap moves.