Lecture 8: Monte Carlo Sampling

@2pm, chemistry seminar,
Bin Zhang, MIT

in 2 weeks @ 2pm
Rob DiStasio, Cornell

Today: Nobel Prize in Physics 2021
Georgio Parisi

Reminder $\langle O \rangle = \int d\vec{X}\, O(\vec{X})\, P(\vec{X})$

$\vec{X} = \{x_1, x_2, \ldots, x_{3N}, p_1, p_2, \ldots, p_{3N}\}$

$\vec{x} = \{x_1 \ldots, x_{3N}\}$ $\quad \vec{p} = \{p_1 \ldots p_{3N}\}$

$\vec{X} = \{\vec{x}, \vec{p}\}$

if $P(\vec{x}, \vec{p}) = P(\vec{x})\, P(\vec{p})$

if $O(\vec{x}, \vec{p}) = O(\vec{x})$

@ constant N, V, T

$$P(\vec{x}, \vec{p}) = e^{-\beta \mathcal{H}(\vec{x}, \vec{p})} / Z$$

$$Z = \int dx \int dp \; e^{-\beta \mathcal{H}(\vec{x}, \vec{p})}$$

$$\mathcal{H}(\vec{x}, \vec{p}) = \sum p_i^2 / 2m_i + U(\vec{x})$$

$$P(\vec{x}, \vec{p}) = \frac{e^{-\beta(KE)} \cdot e^{-\beta U(x)}}{Z_{KE} \;\; Z_{PE}}$$

$$P(\vec{x}) = \int d\vec{p}\; P(\vec{x},\vec{p}) = \int d\vec{p}\, P(\vec{x}) P(\vec{p})$$

$$\int d\vec{p}\, P(\vec{p}) = 1$$

$$= e^{-\beta U(x)} \Big/ Z_{[\text{position}]}$$

Generate a set of configurations $\{\vec{x}_t\}$ where these are distributed according to $P(\vec{x})$

$\{x_t\}$ want

$$P(x_j) / P(x_i) = \frac{e^{-\beta u(x_j)}}{e^{-\beta u(x_i)}}$$

$$= e^{-\beta \Delta u_{ij}}$$

$$\Delta u_{ij} = u(x_j) - u(x_i)$$

Algorithm: (cont)

$x_t \to x_{t+1}$ st

$$P(x_{t+1})/P(x_t) = e^{-\beta \Delta U_{t+1,t}}$$

• Generate a Markov Chain

$$\vec{x}_1 \to \vec{x}_2 \to \vec{x}_3 \to \vec{x}_4 \cdots \vec{x}_T$$

"Markovian": no memory

$P(\vec{x}_t \to \vec{x}_{t+1})$ only depends on $\vec{x}_t$

[not $\vec{x}_1 \ldots \vec{x}_{t-1}$]

Eg Molecular Dynamics

$$\vec{X}_{t+1} = \vec{X}_t + \vec{v}\Delta t \qquad [\vec{X}, \vec{p}] \quad \vec{v} = \vec{p}/m$$

$$\vec{p}_{t+1} = \vec{p}_t + \vec{a}\Delta t$$

Want:

$$P(x_{t+1})/P(x_t) \longrightarrow e^{-\beta \Delta U_{t+1,t}}$$

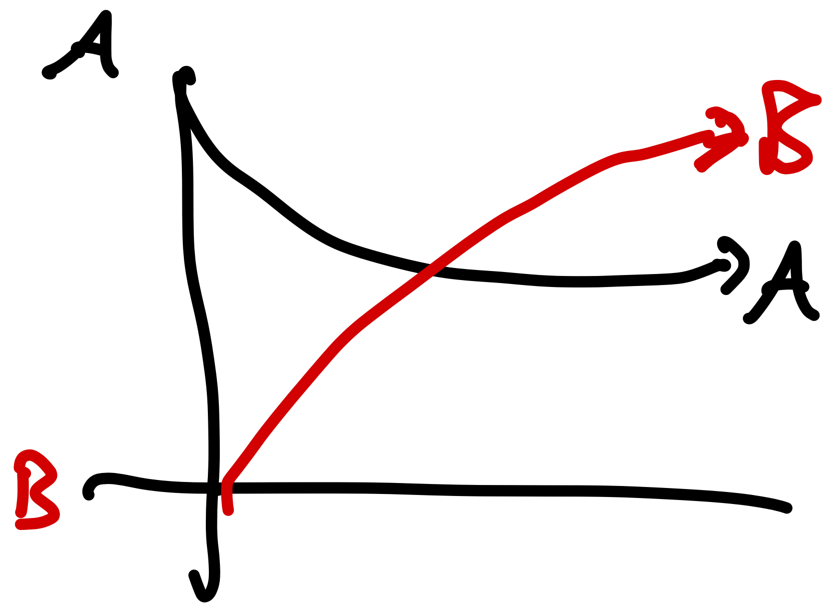
converges? as $t \to \infty$

(1) Markovian

(2) Detailed Balance ☆

Detailed Balance

$$A \rightleftarrows B$$



DB:

$$P(A) \cdot P(A \to B) = P(B) \cdot P(B \to A)$$

$$x_t \to x_{t+1} \to x_{t+2}$$

$$P(x_t)\,\underbrace{P(x_t \to x_{t+1})} = P(x_{t+1})\,P(x_{t+1} \to x_t)$$

Choosing a rule

$P(x \to y)$

$\hookrightarrow P_{gen}(x \to y)\, P_{accept}(x \to y)$

Generation, make new config y

Accept / reject going to y

"rate"

$r(x \to y)$

$$P(x)\, P_{gen}(x \to y)\, P_{acc}(x \to y) = P(y)\, P_{gen}(y \to x)\, P_{acc}(y \to x)$$

$$P_{acc}(x \to y) = \left[\frac{P(y)}{P(x)} \cdot \frac{P_{gen}(y \to x)}{P_{gen}(x \to y)} \right] P_{acc}(y \to x)$$

Metropolis Monte Carlo (1953)

MTTRR → Manhattan project
Los Alamos Computing Center

$$P_{acc}(x \to y) = \min(1, r(x \to y))$$

$$P_{acc}(x \to y) = \left[\frac{P(y)}{P(x)} \cdot \frac{P_{gen}(y \to x)}{P_{gen}(x \to y)} \right] P_{acc}(y \to x)$$

$$P_{acc}(x \to y) = r(x \to y)\, P_{acc}(y \to x)$$

$$P_{acc}(y \to x) = r(y \to x)\, P_{acc}(x \to y)$$

$$r(x \to y) = \frac{1}{r(y \to x)}$$

(a) $r(x \to y) > 1$ st $r(y \to x) < 1$

(b) $r(x \to y) < 1$ st $r(y \to x) > 1$

$$P_{acc} = \min[1, r(x \to y)]$$

$$\frac{P_{acc}(x \to y)}{P_{acc}(y \to x)} = r(x \to y)$$

a) $P_{acc}(x \to y) = 1$

$P_{acc}(y \to x) = 1/r(x \to y)$

b) $P_{acc}(x \to y) = r(x \to y) < 1$

$P_{acc}(y \to x) = 1$

Algorithm: Start at config $\vec{x}_t$

① Propose x_{t+1} w/ prob $P_{gen}(x_t \to x_{t+1})$

generate uniform rand number "a" $[0,1)$

② if $a < \min[1, r(x_t \to x_{t+1})]$

accept, move to x_{t+1}

else:

$x_{t+1} = x_t$

③ go back to 1

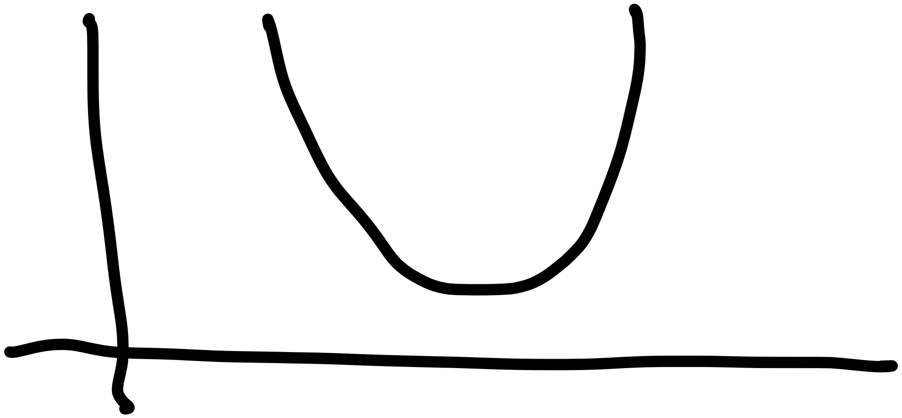
$$r(x\to y) = \underbrace{\cancel{\frac{P_{gen}(y\to x)}{P_{gen}(x\to y)}}}_{\text{gen part}} \; \underbrace{\frac{P(y)}{P(x)}}_{\text{prob part}}$$

For canonical ensemble

$$P(y)/P(x) = e^{-\beta[U(y)-U(x)]}$$

$$e^{-\beta[H(\vec{x}_2,\vec{p}_2)-H(\vec{x}_1,\vec{p}_1)]}$$

Make generation symmetric



$$\mathcal{H}(x,p) = \frac{p^2}{2m} + \frac{1}{2}kx^2$$

sample

$$P(x) = e^{-\beta U(x)} \Big/ \int e^{-\beta U(x)} = \frac{e^{-\beta \frac{1}{2}kx^2}}{\sqrt{2\pi \frac{k_B T}{k}}}$$

$$\sigma = \sqrt{\frac{k_B T}{x}}$$

Move rule, that is symmetric

propose:

$$X_{t+1} = X_t + a_2 \cdot \xi$$

$r_2 \in (-1, 1)$ uniform random

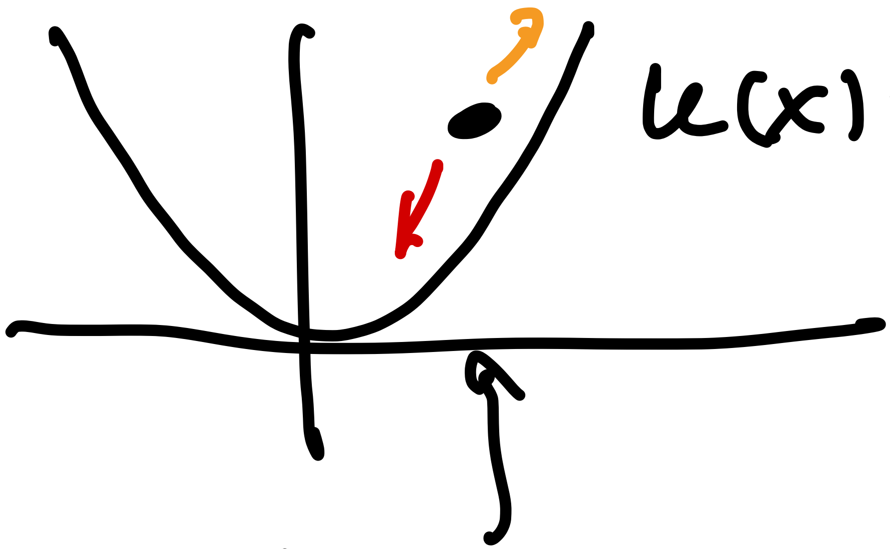
ξ biggest possible move

$$X_{t+2} = X_{t+1} + a_2 \cdot \xi$$

$$\frac{P_{gen}(y \to x)}{P_{gen}(x \to y)} = 1 \qquad r(x \to y) = e^{-\beta \Delta U}$$

$$P_{acc}(X_t \to X_{t+1}) = \min(1, e^{-\beta \Delta U})$$

$$P_{acc}(x_t \to x_{t+1}) = \min(1, e^{-\beta \Delta u})$$



$u(x) = \frac{1}{2} k x^2$

how does Δu compare to $k_B T$

current pos

move 1, energy goes down

$u(x_2) - u(x_1) < 0$, $e^{-\beta \Delta u} > 1$, always accept

Move 2 $u(x_2) - u(x_1) > 0$, $e^{-\beta \Delta u} < 1$, accept prob $e^{-\beta \Delta u}$

Tune our moves, here ξ

st average acceptance rate $\sim$

0.25 → 0.5

Trade off between efficiency &

exploration

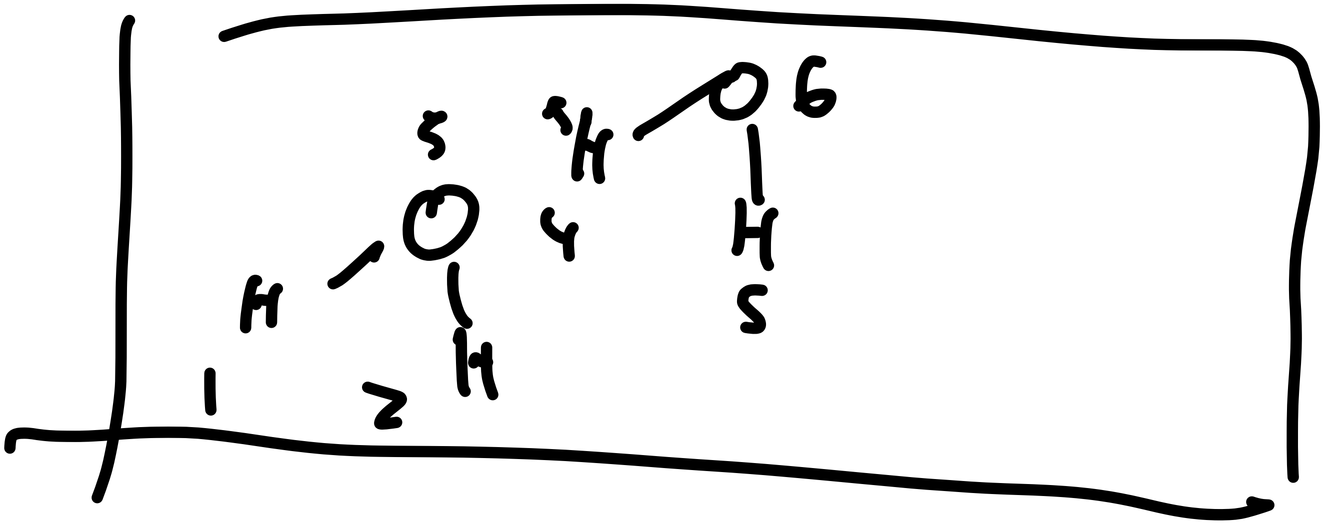
if ξ is large, each accepted

move will go far, but most

moves will be rejected

ξ is really small, almost always accept

but stay close to starting point



Doesn't have to be this simple

(1) pick an atom [atom 4]

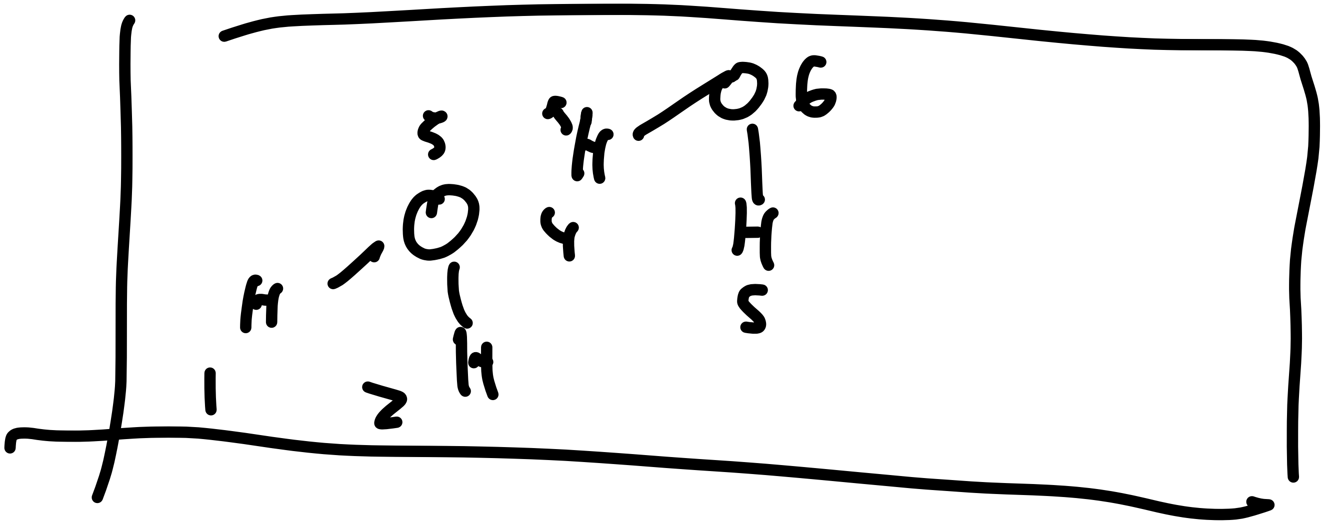
(2) generate a random move

$$\vec{x}_4(t+1) = \vec{x}_4(t) + \text{random}$$

$$\vec{x}_4 = \begin{bmatrix} x_4 \\ y_4 \\ z_4 \end{bmatrix} + \begin{bmatrix} \text{random} \cdot \xi \\ \text{random} \cdot \xi \\ \text{random} \cdot \xi \end{bmatrix}$$

$r \in (-1, 1)$

(3) then calc $e^{-\beta \Delta U}$ for whole system



Random move

(1) move CoM of molecule

$$X_{com} = X_{com} + \vec{a}\,\xi$$

(2) rotate molecule by a random angle

Why MC & why not:

1. easy
2. can choose very smart types of moves, jump over energy barriers s.t. exploration is very fast

Why not:

1. not real dynamics [gives static properties]
2. usually only tiny changes accepted

Do we have to use Metropolis rule

Glauber Rule

$$P_{acc}(x \to y) = \frac{e^{-\beta \Delta U/2}}{e^{-\beta \Delta U/2} + e^{+\beta \Delta U/2}}$$

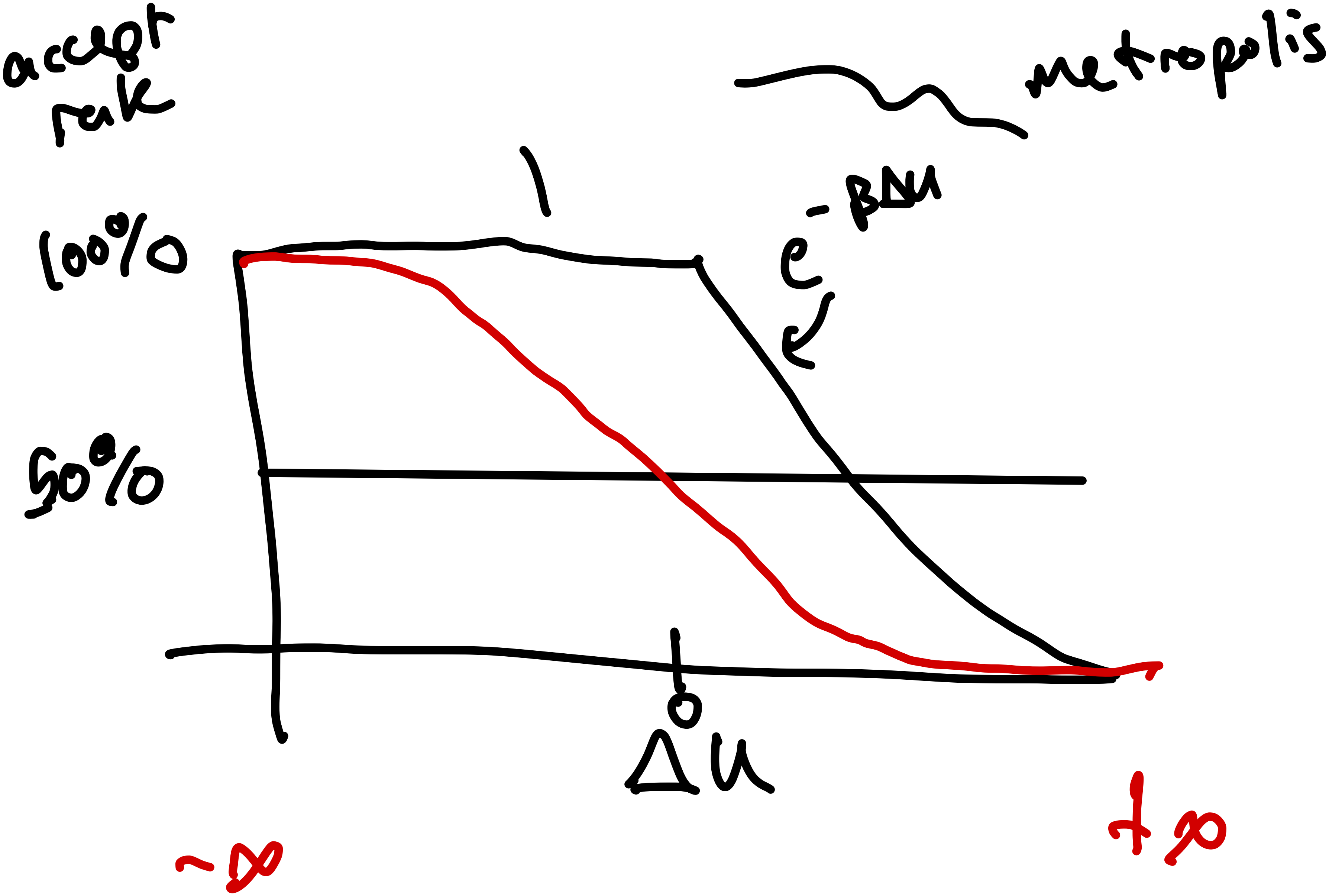
why?

$$\Delta U_{xy} = U(y) - U(x)$$

$$\Delta U_{yx} = U(x) - U(y) = -\Delta U_{xy}$$

$$\frac{P_{acc}(x \to y)}{P_{acc}(y \to x)} = e^{-\beta \Delta U_{xy}}$$

✓



accept rate
metropolis
1
$e^{-\beta \Delta U}$
100%
50%
0
ΔU
$-\infty$
$+\infty$