

hybrid Monte Carlo (hamiltonian)

の紹介

@プログラマのための数学勉強会 Nov, 2015

... 数学というより計算方法の話です

Kenji Ogawa topazS50@github

総研大 -> ポスドク (素粒子物理学, LQCD) -> AI

New One-Flavor Hybrid Monte Carlo Simulation Method for Lattice Fermions with gamma-five Hermiticity, PLB 2011

楕円曲線、数学オリンピックに興味があります

コード [topazS50@github intro_hmc](https://github.com/topazS50/intro_hmc)

Outline

Hybrid Monte Carloの目的(2)

Base of HMC: Detailed Balance (6)

Realisation: Metropolis Method(3)

Example - Metropolis

Building Block of HMC: Molecular Dynamics(3)

Realisation: LeapFrog

Example - HMC

HybridMonteCarloの目的

あるタイプの積分

xの次元が大きい時
単純なグリッド分割
の積分は不可能

$$I = \frac{1}{Z} \int d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x})$$

$$Z = \int d\mathbf{x} e^{-f(\mathbf{x})}$$

$$\mathbf{x} = (x_0, x_1, \dots, x_{N-1}) \quad N \gg 1$$

サンプリング

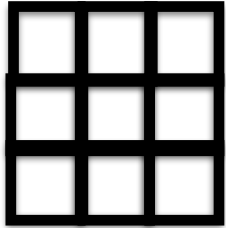
$$I \sim \frac{1}{N_{\text{sample}}} \sum_{i=0}^{N_{\text{sample}}-1} g(\mathbf{x}_i)$$

$$P(\mathbf{x}) \propto e^{-f(\mathbf{x})}$$
$$N_{\text{sample}} \rightarrow \infty$$

MonteCarloやMetropolisとはどう違うの？

=>効率がいい(これに関して後の方で)

$$I = \frac{1}{Z} \int d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x})$$



の用途

$f(x)$, $g(x)$ ともに複雑

格子場の理論

自発的対称性の破れ, Higgs, QCD(クォーク閉じ込め)

ランダム行列理論

~強相関系の低エネルギー極限

(制約)ボルツマンマシン

Deep-learning

分布が $P(x)$ になるような
 $x_0, x_1, \dots, x_{N\text{sample}}$
を生成するには？

十分条件

detailed balance

実現方法の一つ

Metropolis法

Detailed Balance

$$P_i P(i \rightarrow j) = P_j P(j \rightarrow i)$$

欲しいもの P_i, P_j : 状態 i, j になる確率

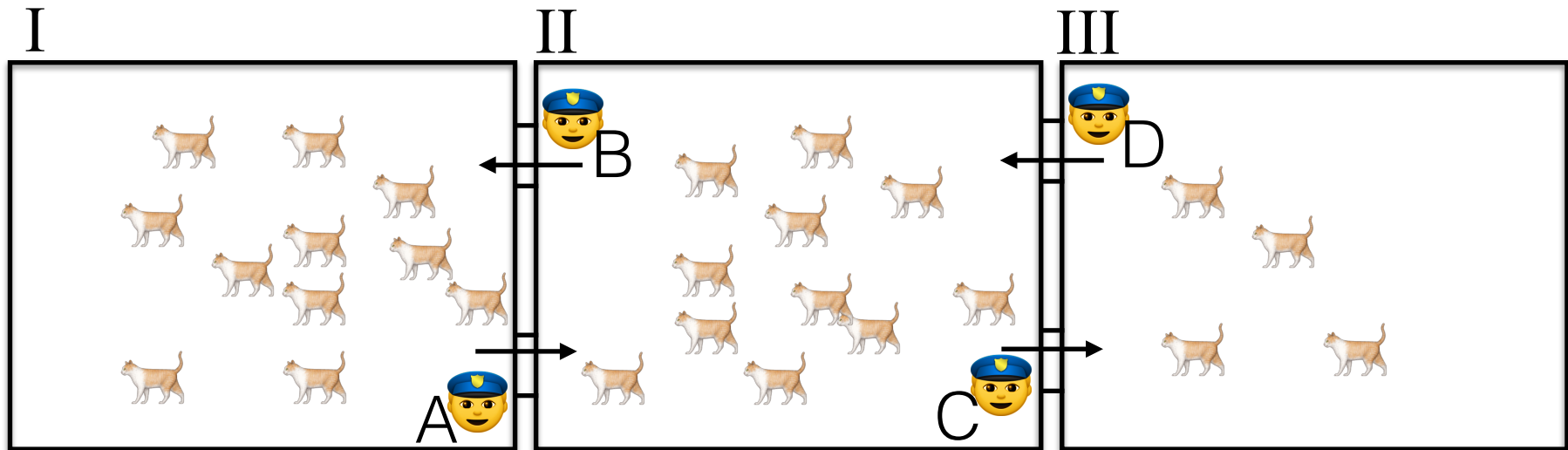
エンジニアリング
で作る $P(i \rightarrow j)$: 状態が i から j に遷移する確率

$P(i \rightarrow j)$ を調節すれば自然と P_i は調節される
(前提条件はあります)

i から j となる粒子の総量と j から i になる粒子の総量が同じ
なら巨視的には状態は同じ

Detailed Balance

$$P_i P(i \rightarrow j) = P_j P(j \rightarrow i)$$



👮はそれぞれ決まった割合で🐈を通過させる。

e.g (A:1, B:1/3, C:1, D:1/2) \rightarrow I:II:III=1:6:6

=> 🐈の分布は👮によってコントロールできる。

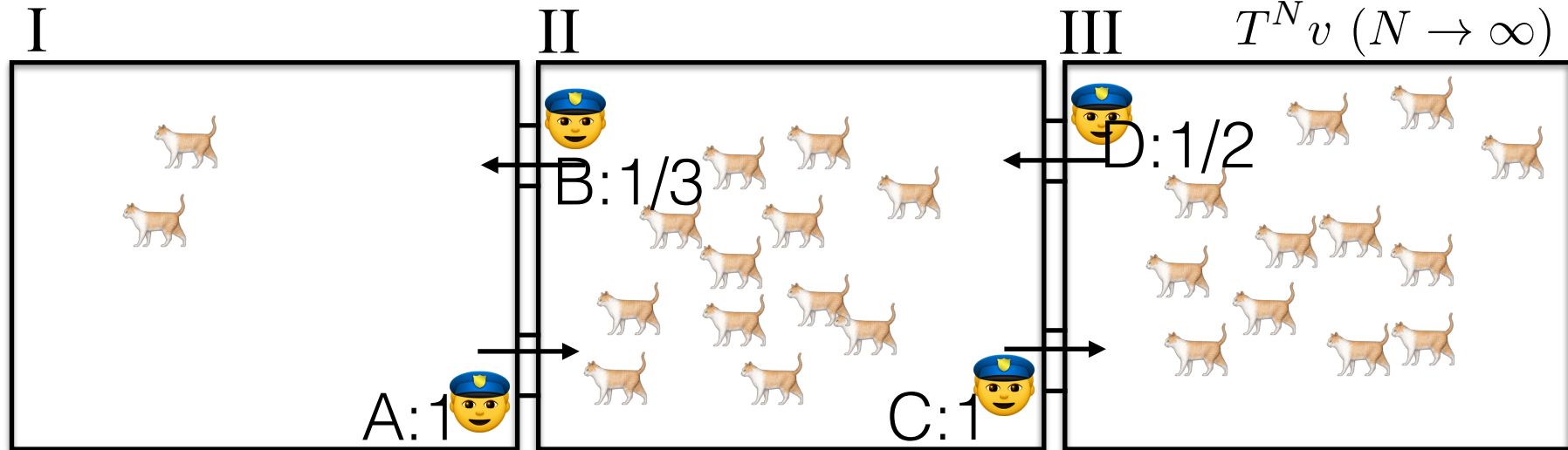
Detailed Balance

遷移行列	I	II	III
I	0	1/6	0
II	1	1/3	1/2
III	0	1/2	1/2

$$P_i P(i \rightarrow j) = P_j P(j \rightarrow i)$$

分布はこの行列の最大固有値
に対応する固有ベクトル

$T^N v (N \rightarrow \infty)$ 冪乗法

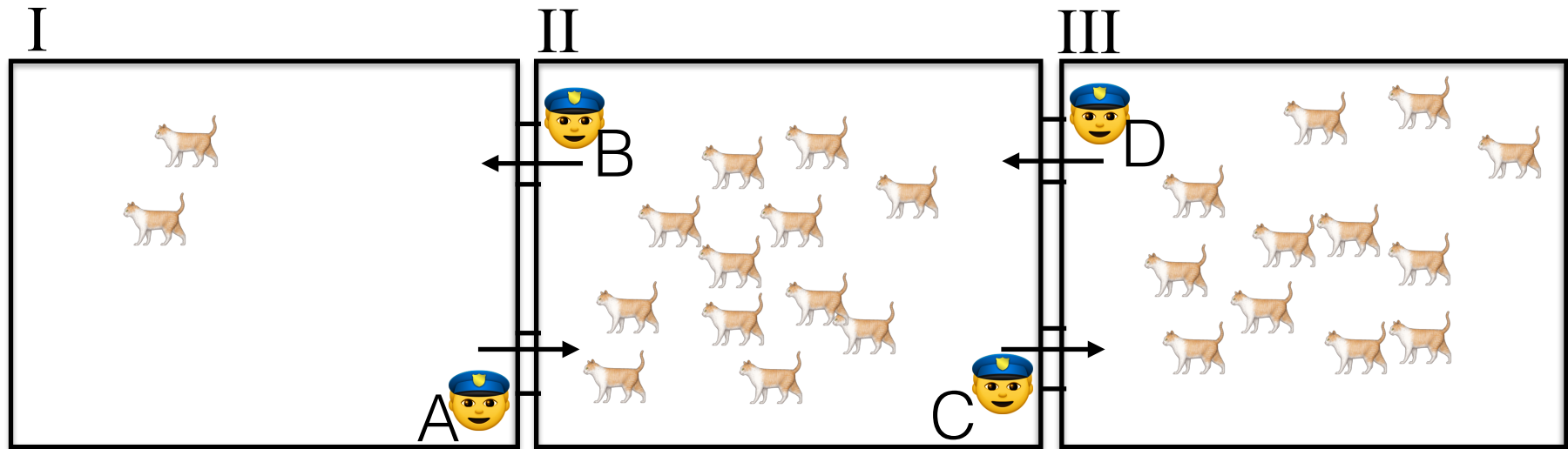


👮はそれぞれ決まった割合で🐈を通過させる。

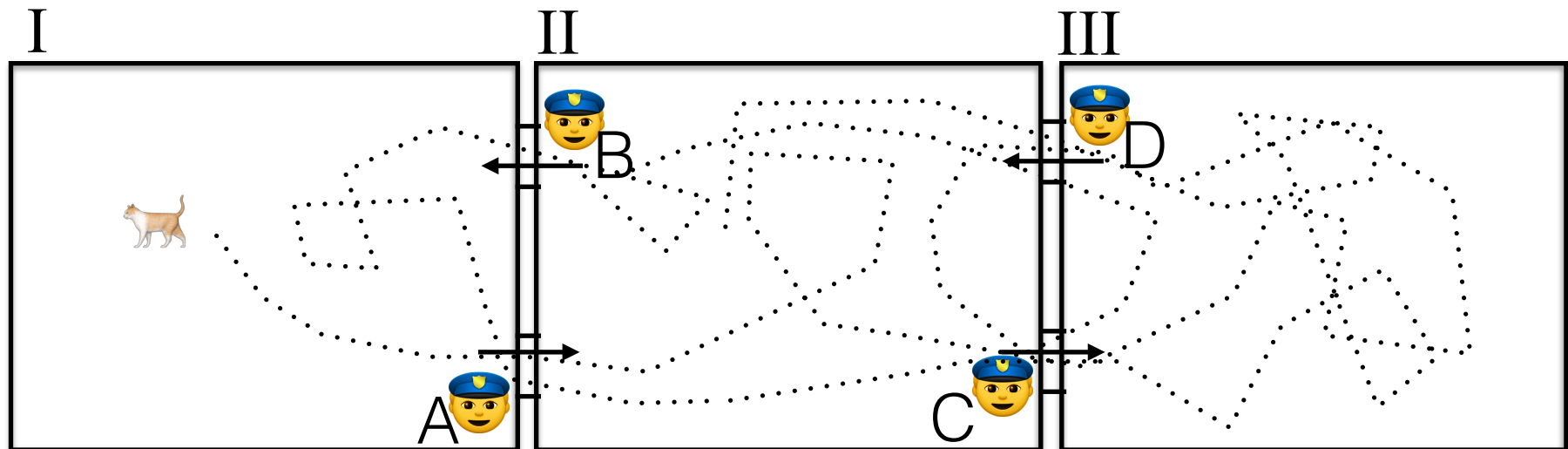
e.g (A:1, B:1/3, C:1, D:1/2) \rightarrow I:II:III=1:6:6

=> 🐈の分布は👮によってコントロールできる。

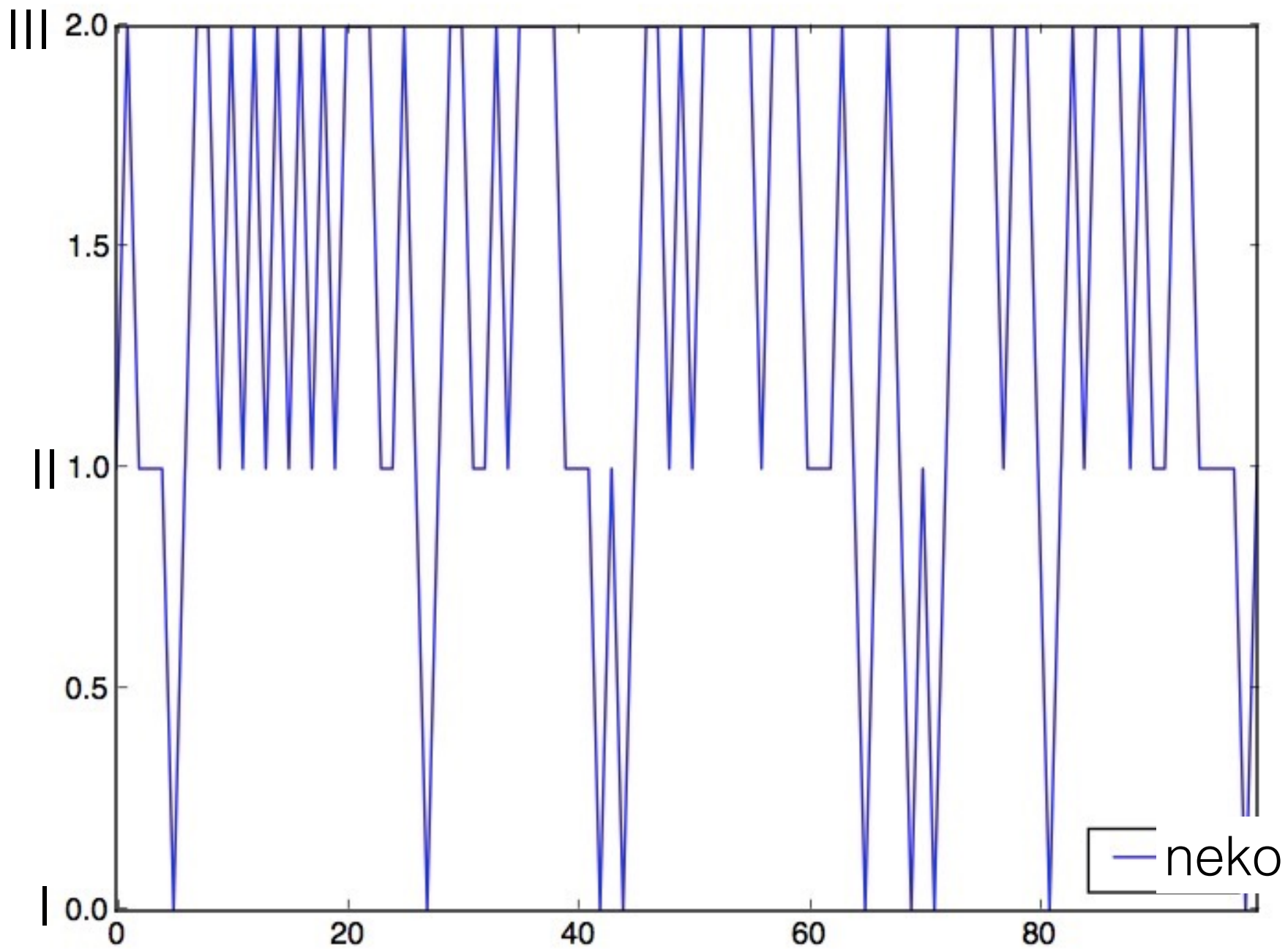
Markov Chain MonteCarlo



たくさんいる の分布 = 一匹の の長い時間の軌跡*



*エルゴード性が必要です



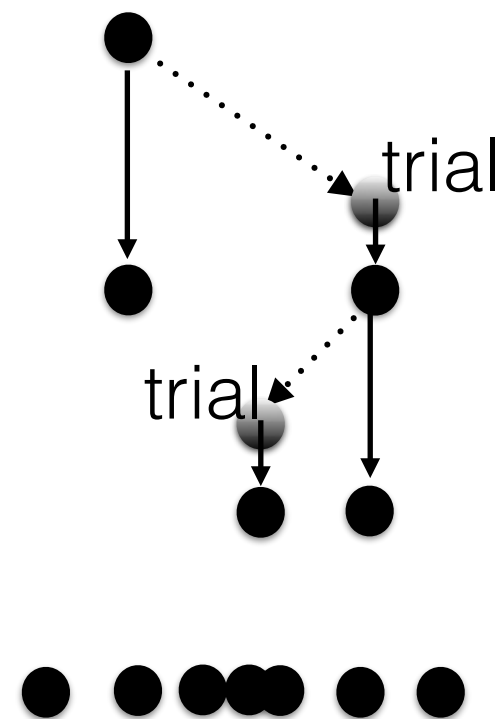
コード topazS50@github intro_hmc

Metropolis Test

$$I = \frac{1}{Z} \int d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x}) \sim \frac{1}{N_{\text{sample}}} \sum_{i=0}^{N_{\text{sample}}-1} g(x)$$

MCMCでDetailed Balanceを実現する

0. 初期状態 x を選ぶ
1. Trial State x' を選ぶ*
2. $0 \sim 1$ の一様乱数 r を選ぶ
3. $e^{-f(x')}/e^{-f(x)} > r$ なら $x := x'$, そう
でなければ x はそのまま
4. x の値を記録
5. 1. - 4.を何回もやる
6. 記録した x について $g(x)$ を計算



*ここでは x' は x とは別のものという意味で使います。微分ではありません。

$e^{-f(x')}/e^{-f(x)} > r$ なら $x := x'$, そうでなければ x はそのまま

$f(x) > f(x')$ のとき

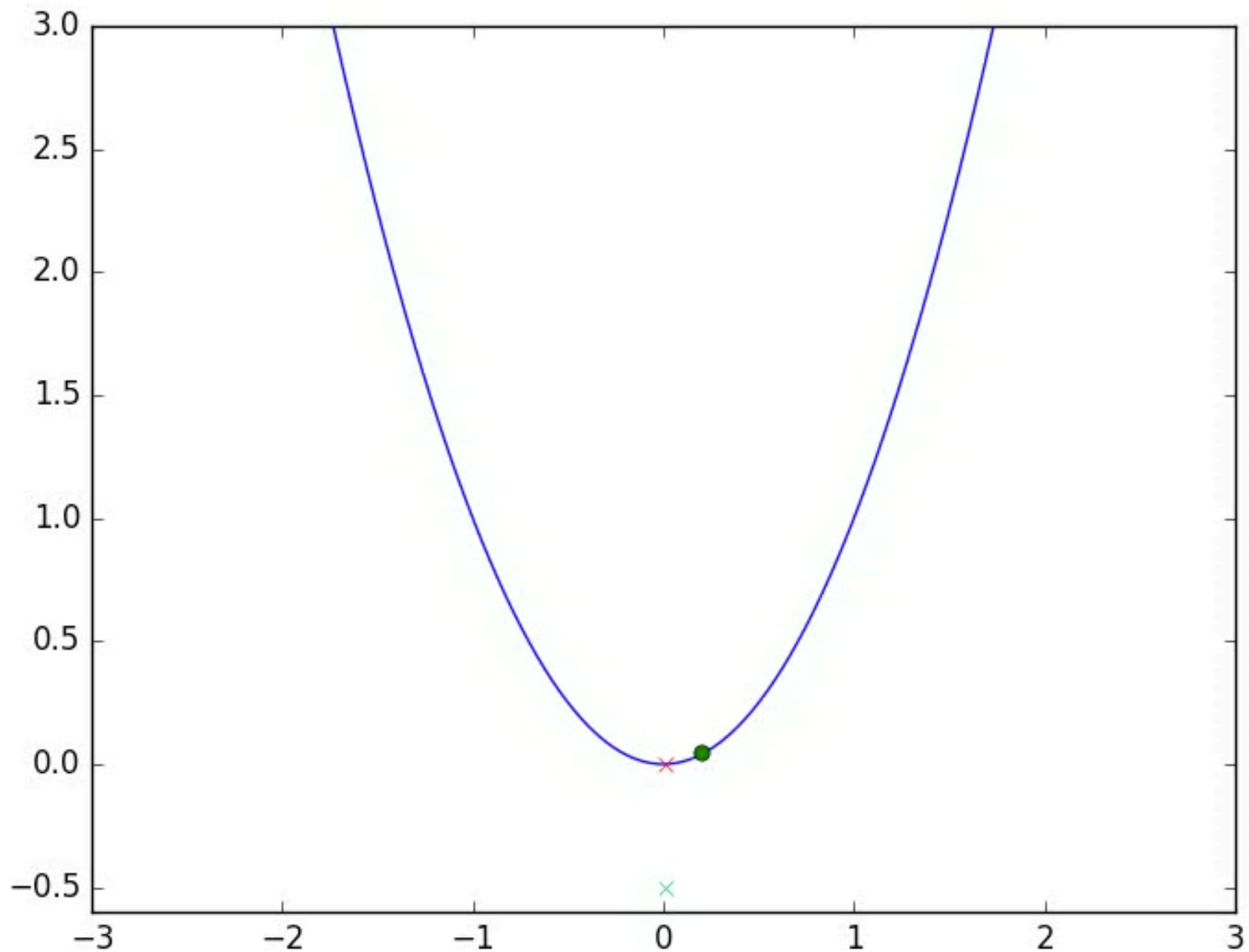
$$P(x \rightarrow x') = 1, P(x' \rightarrow x) = e^{-f(x)}/e^{-f(x')}$$

$f(x) < f(x')$ のとき

$$P(x \rightarrow x') = e^{-f(x')}/e^{-f(x)}, P(x' \rightarrow x) = 1$$

Detailed Balance !

$$e^{-f(x)} P(x \rightarrow x') = e^{-f(x')} P(x' \rightarrow x)$$



demo metropolis
`$python mtpl.py; python anim_mtpl.py`

topazS50@github
intro_hmc

Acceptance Ratio

0. 初期状態 x を選ぶ
1. Trial State x' を選ぶ
2. $0 \sim 1$ の一樣乱数 r を選ぶ
3. $\frac{e^{-f(x')}}{e^{-f(x)}} > r$ なら $x := x'$, そうでなければ x はそのまま
4. x の値を記録
5. 1. - 4.を何回もやる
6. 記録した x について $g(x)$ を計算

Trial State x' はどう選んでもいいが

$e^{-f(x')}/e^{-f(x)}$ が小さいとずっと
 x は同じまま

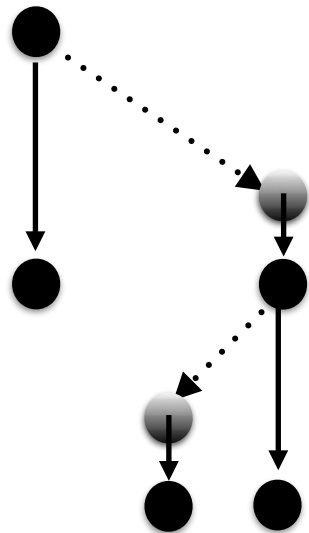
$x' := x + \varepsilon$ とするのが定石

少しずつしか進めない

ε は $+/-$ がありうるので戻ってくる

効率よくできないか

=>HMC



HMC

$$\frac{1}{Z} \int d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x}) = \frac{1}{Z_{\text{HMC}}} \int d\mathbf{p} e^{-\frac{1}{2}\mathbf{p}^2} \int d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x})$$

pとxを同時にアップデートして

$\frac{1}{2} \mathbf{p}^2 + f(\mathbf{x})$ が変わらないようにする

Molecular Dynamics

Detailed Balanceは保証しないといけない

Metropolis Test

Molecular Dynamics

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

$$\frac{d}{dt} H(x(t), p(t)) = \frac{dx}{dt} \frac{\partial H}{\partial x} + \frac{dp}{dt} \frac{\partial H}{\partial p} = \frac{dx}{dt} \frac{df(x)}{dx} + \frac{dp}{dt} p$$

$$\frac{dx}{dt} = p, \quad \frac{dp}{dt} = -\frac{df(x)}{dx} \quad \text{ならば} \quad H \text{は変化しない} \quad dH/dt = 0$$

(ハミルトニアンの保存)

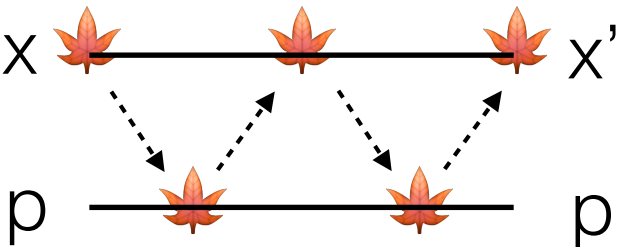
シミュレーションでやるときは

Leap Flogを使う*

reversibilityがDetailed
Balance には必要

$$\mathbf{x} := \mathbf{x} + \epsilon_x \mathbf{p}$$

$$\mathbf{p} := \mathbf{p} - \epsilon_p \partial f(\mathbf{x}) / \partial \mathbf{x}$$



シミュレーションのパラメータ

$$\epsilon_x = \lambda_t \quad \lambda_t \quad N_{\text{step}}$$

$$\epsilon_p = \frac{\lambda_t}{2}, \lambda_t, \dots, \lambda_t, \frac{\lambda_t}{2}$$

*Omelyanなど発展形あり

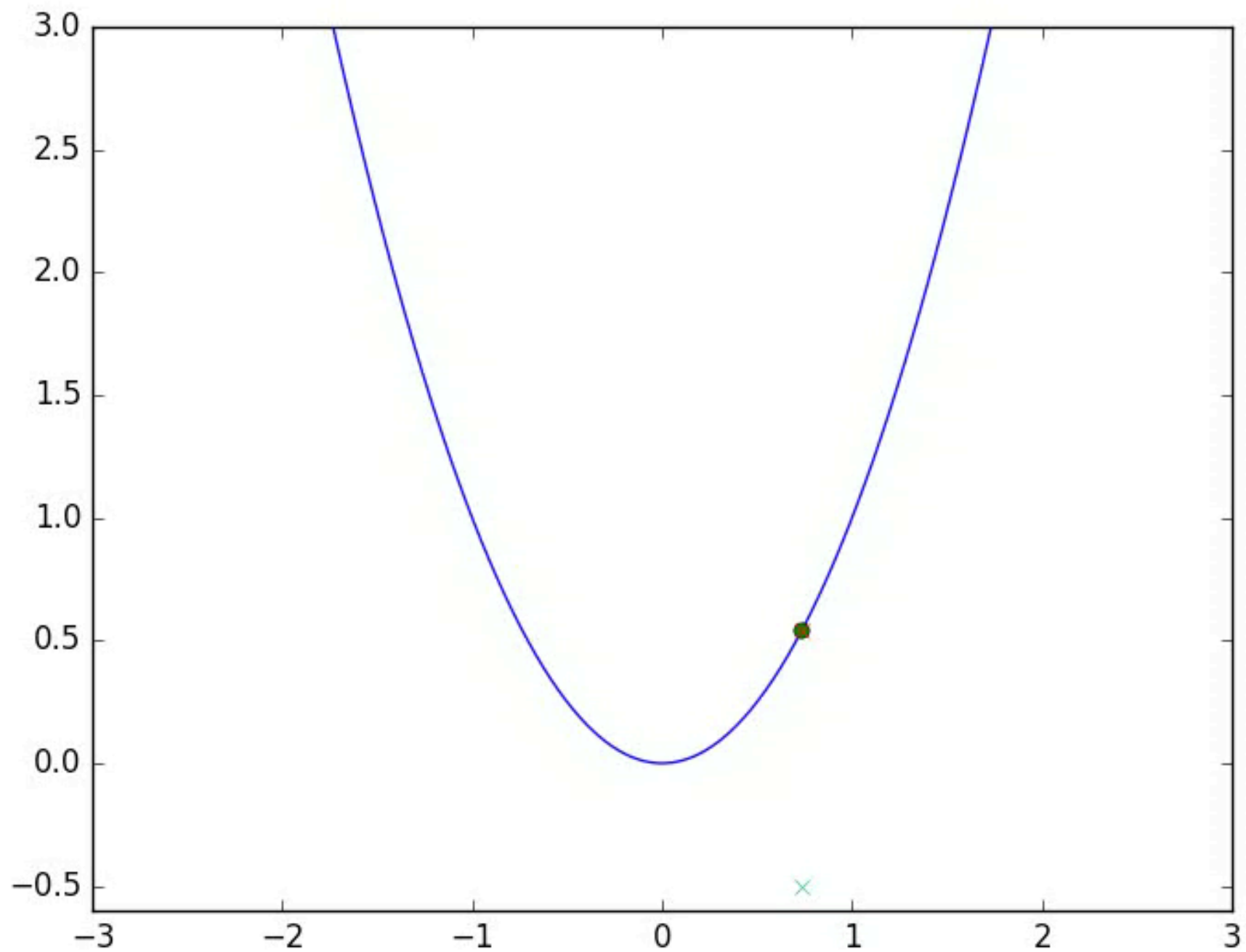
HMC

$$\frac{1}{Z} \int d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x}) = \frac{1}{Z_{\text{HMC}}} \int d\mathbf{p} e^{-\frac{1}{2}\mathbf{p}^2} d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x})$$

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

0. 初期状態 x を選ぶ

1. p を $\exp(-1/2 p^2)$ の確率で生成
2. leap frog を使って $(x, p) \rightarrow (x', p')$ を得る
4. $0 \sim 1$ の一様乱数 r を選ぶ
5. $e^{-H(x', p')} / e^{-H(x, p)} > r$ なら $x := x'$, そうでなければ x はそのまま
6. x の値を記録
7. 1. -7.を何回もやる
8. 記録した x について $g(x)$ を計算



demo hmc

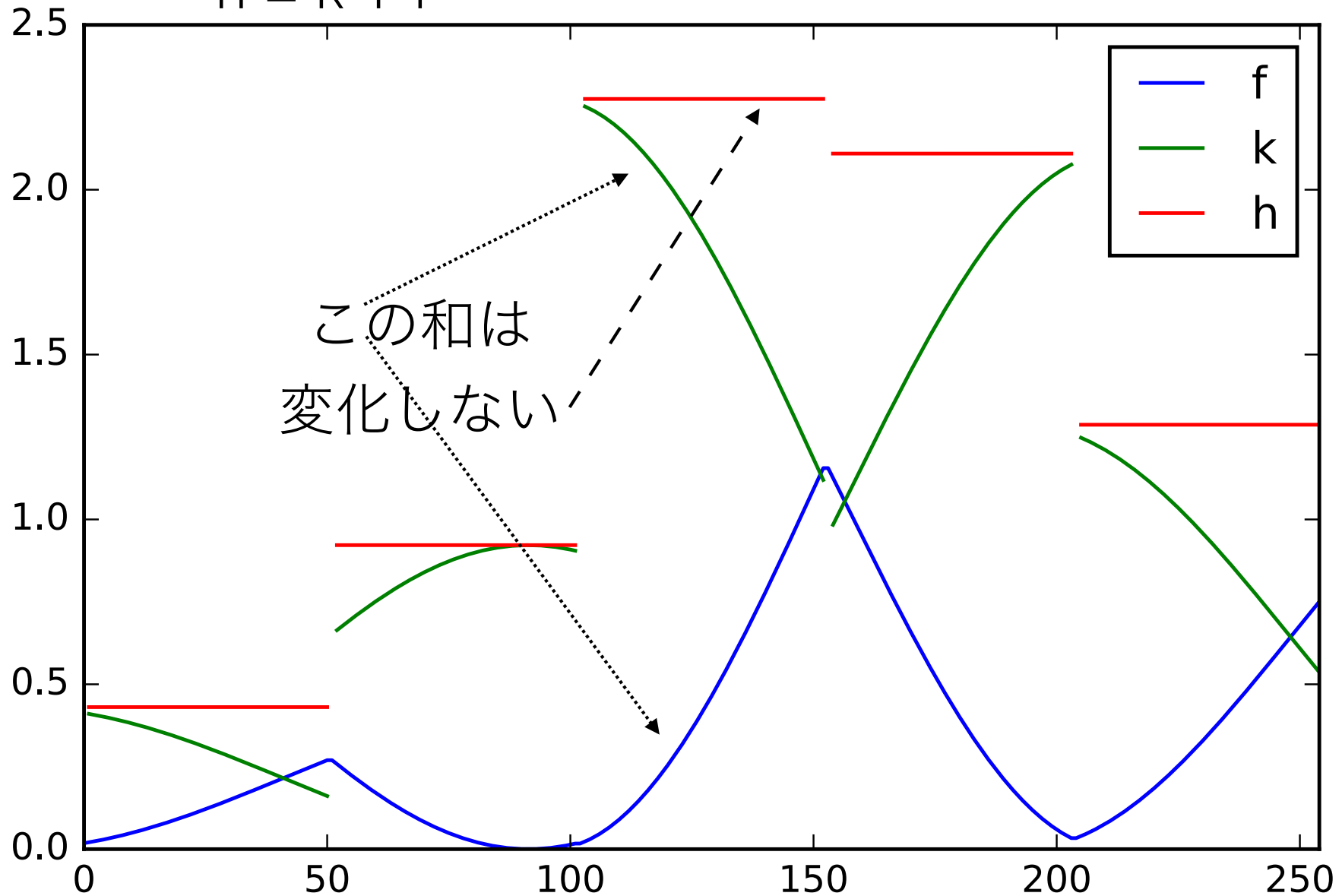
コード topazS50@github intro_hmc

Conservation of Hamiltonian in HMC

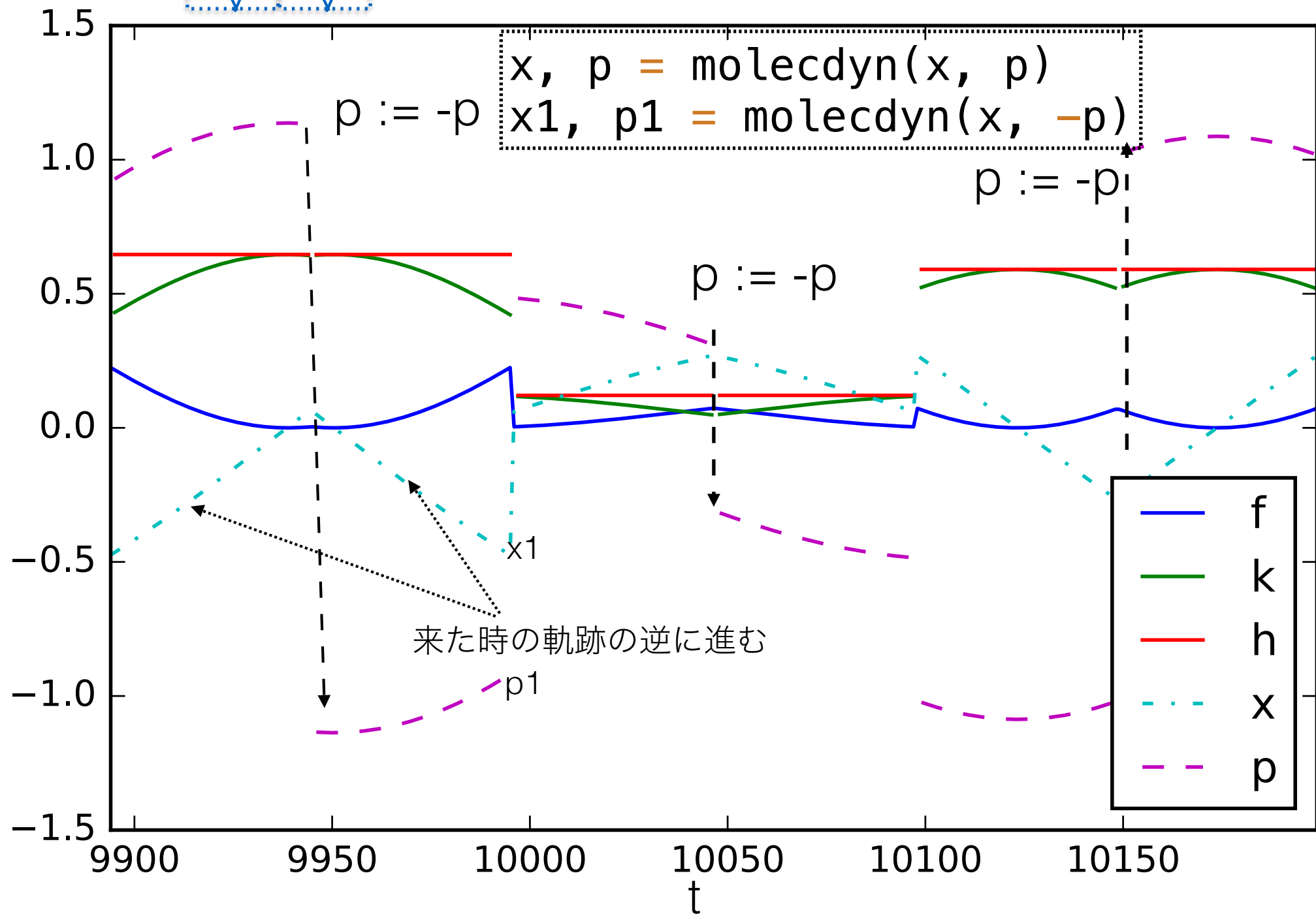
$$k = 0.5 * p^2$$

$$f = f(x)$$

$$h = k + f$$



Leap Frog. Reversibility のデモンストレーション



まとめ

目的 $I = \frac{1}{Z} \int d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x}) \sim \frac{1}{N_{\text{sample}}} \sum_{i=0}^{N_{\text{sample}}-1} g(x)$
with $P(\mathbf{x}) = e^{-f(\mathbf{x})}$

Detailed Balance

$$P(x) P_{\text{transition}}(x \rightarrow x') = P(x') P_{\text{transition}}(x' \rightarrow x)$$

Metropolis Test

$$x_{\text{next}} := x_{\text{trial}} \quad \text{with probability} \quad \min \left(1, \frac{P(x_{\text{trial}})}{P(x)} \right) \quad \text{else} \quad x_{\text{next}} := x$$

HMC $\int d\mathbf{x} e^{-f(\mathbf{x})} g(\mathbf{x}) \rightarrow \int d\mathbf{p} d\mathbf{x} e^{-p^2} e^{-f(\mathbf{x})} g(\mathbf{x})$

Molecular Dynamics $\frac{dx}{dt} = p, \frac{dp}{dt} = -\frac{df(x)}{dx}, \frac{dH}{dt} = 0$

今回話せなかったこと

パラメータについて

AutoCorrelation

Lattice QCDに関すること

Fermion Determinant and Pseudo Fermion Action

Schur Decomposition

MultiTimeStep

観測できる現象、物理学


Spontaneous Symmetry Breaking

結合定数の繰り込み

Appendix

Power Method を知っていると MCMCの収束についてより 理解できるというお話

初期状態を $v_0 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$ として

一回の遷移の後
がそれぞれの
部屋にいる確率は*

$$v := Tv_0 = \begin{pmatrix} 1/6 \\ 1/3 \\ 1/2 \end{pmatrix}$$

N回繰り返すと $v \propto T^N v_0 \sim T$ の最大固有ベクトル



この事実は v_0 がどんなベクトルでも変わらない**

*今、 v のnormではなく v のsumが 1.0 となるように規格化している

** T の最大固有値が縮退してたらどうするんだ！ とは聞かないでください... (サンプル数を増やして確認するしか方法がない)

$T =$

	I	II	III
I	0	1/6	0
II	1	1/3	1/2
III	0	1/2	1/2

とのモデルに
おける遷移行列

$$e^{-f(x)} P(x \rightarrow x') = e^{-f(x')} P(x' \rightarrow x)$$

HMCの手順を紐解いていく

$$P(x \rightarrow x') = \int dp \int dp' e^{-\frac{1}{2}p^2} P(x, p \rightarrow x', p')$$

p は $\exp(-1/2 p^2)$ の確率で生成、 $P(x, p \rightarrow x', p')$ はMolc.Dyn.とMetropolis Test

Metropolis Testが終わった後は p' について気にする事はないので p' はintegrate outするだけ

x, p が与えられた時、Molc Dynは決まった終点を与える。その後、Metropolis Testをする

$$P(x, p \rightarrow x', p') = \delta(x'_{MD}(x, p) - x') \delta(p'_{MD}(x, p) - p') P_{\text{Metropolis}}(H(x, p), H(x'_{MD}, p'_{MD}))$$

$x'_{MD}(x, p)$, $p'_{MD}(x, p)$ は x, p からMolc.Dyn.を行った時に終点で得られる値*

これらをまとめると証明したい式の左辺は

$$\begin{aligned} & e^{-f(x)} P(x, p \rightarrow x', p') \\ &= \int dp \int dp' e^{-f(x) - \frac{1}{2}p^2} P_{\text{Metropolis}}(H(x, p), H(x'_{MD}(x, p), p'_{MD}(x, p))) \delta(x'_{MD} - x') \delta(p'_{MD} - p') \end{aligned}$$

* $x'_{MD}(x, p)$ を x'_{MD} と引数を省略して書く事があります。

前ページの式、証明したい式の左辺

$$\begin{aligned} & e^{-f(x)} P(x, p \rightarrow x', p') \\ &= \int dp \int dp' e^{-f(x) - \frac{1}{2}p^2} P_{\text{Metropolis}}(H(x, p), H(x'_{MD}(x, p), p'_{MD}(x, p))) \delta(x'_{MD} - x') \delta(p'_{MD} - p') \end{aligned}$$

ちょっと手直し

$$= \int dp \int dp' e^{-H(x, p)} P_{\text{Metropolis}}(H(x, p), H(x', p')) \delta(x'_{MD} - x') \delta(p'_{MD} - p')$$

ここで次の事実を使います

S1. P_Metropolisは単体でDetailed Balanceを与える

$$e^{-H(x, p)} P_{\text{Metropolis}}(H(x, p), H(x', p')) = e^{-H(x', p')} P_{\text{Metropolis}}(H(x', p'), H(x, p))$$

S2. Molc.Dyn.(LeapFlopSteps)のreversibility

$$\begin{aligned} x'_{MD}(x'_{MD}(x, p), -p'_{MD}(x, p)) &= x, \\ p'_{MD}(x'_{MD}(x, p), -p'_{MD}(x, p)) &= -p \end{aligned}$$

S3. ハミルトニアンはpを-pとしても同じ*

$$H(x, p) = H(x, -p)$$

これらを使うと右辺が出るぞ！(次ページで)

*(念のため)

integral dpも

p → -pで変わらない

左辺

$$= \int dp \int dp' e^{-H(x,p)} P_{\text{Metropolis}} (H(x,p), H(x',p')) \delta(x'_{MD} - x') \delta(p'_{MD} - p')$$

S1 使う

$$= \int dp \int dp' \underline{e^{-H(x',p')}} P_{\text{Metropolis}} (H(x',p'), H(x,p)) \delta(x'_{MD} - x') \delta(p'_{MD} - p')$$

S2から次の式を得る (x', p'のデルタ関数を x,pのデルタ関数として読み替える)

$$\delta(x'_{MD}(x,p) - x') \delta(p'_{MD}(x,p) - p') = \delta(x'_{MD}(x', -p') - x) \delta(p'_{MD}(x', -p') - p) \dots (*)$$

代入

$$\int dp \int dp' e^{-H(x',p')} P_{\text{Metropolis}} (H(x',p'), H(x,p)) \underline{\delta(x'_{MD}(x', -p') - x) \delta(p'_{MD}(x', -p') - p)}$$

S3を使う。pを-pにする

$$\int dp \int \underline{dp'} e^{-H(x',p')} P_{\text{Metropolis}} (H(x',p'), H(x,p)) \underline{\delta(x'_{MD}(x', p') - x) \delta(p'_{MD}(x', p') - p)}$$

こっちはそのままだよ

積分変数 (p,p')はこっちの都合で(p',p)と書き換えても良い

左辺を(x,x') -> (x',x)
にしたもの

$$\int dp \int dp' e^{-H(x',p)} P_{\text{Metropolis}} (H(x',p), H(x,p')) \delta(x'_{MD}(x',p) - x) \delta(p'_{MD}(x',p) - p')$$

というわけで、
左辺 $\exp(-f(x))P(x \rightarrow x')$ は x, x' を交換しても同じ値になるので
$$e^{-f(x)} P(x \rightarrow x') = e^{-f(x')} P(x' \rightarrow x)$$

の証明ができました。

... が、
私は一つずるをしてしまいました。
前ページの 式(*) です。

$$\delta(x'_{MD}(x, p) - x') \delta(p'_{MD}(x, p) - p') = \delta(x'_{MD}(x', -p') - x) \delta(p'_{MD}(x', -p') - p)$$

デルタ関数の性質（ヤコビアン）を考慮して、

式(*)が成り立つには Area Conserving

$$\begin{aligned} & \text{Area}((x, p), (x + \Delta x, p), (x + \Delta x, p + \Delta p), (x, p + \Delta p)) \\ &= \text{Area}(f_{MD}(x, p), f_{MD}(x + \Delta x, p), f_{MD}(x + \Delta x, p + \Delta p), f_{MD}(x, p + \Delta p)) \end{aligned}$$

$$\Delta x, \Delta p \text{ は微小量} \quad f_{MD}(x, p) := (x'_{MD}(x, p), p'_{MD}(x, p))$$

という関係式が必要です。

私の力不足により、ここでは Leap Flog ならば成り立っていると言うにとどめます。
とりあえず 証明終わり