

マルコフ連鎖モンテカルロと *HMC*

金森 逸作 (理研 R-CCS)

2024 年 9 月 11 日
格子場の理論夏の学校 2024 @ 筑波大学東京キャンパス



ちょっとだけ自己紹介 (+ 宣伝)

- 名前：金森 逸作
- 所属：理化学研究所 計算科学研究センター (R-CCS)
神戸、スパコン「富岳」のあるところ
- 普段やっていること
有限温度 QCD シミュレーション、コード・アルゴリズム開発 (Bridge++)、
「富岳」向けコードチューニング etc.
ときどき、勉強会: HPC-Phys 勉強会 <https://hpc-phys.kek.jp/>
- 学生の時にやっていたこと
超対称理論の格子正則化 (理論的な話、紙と鉛筆) 数値計算より理論だった
- メッセージ
格子は数値計算だけではないし、数値計算も面白い。いま理論をやっている人・
やりたいと思っている人も、数値計算やコード開発をやることになるかもしれ
ない (全部おもしろい)



今回の講義は主にこの2つをもとに用意した

- 大川正典・石川健一「格子場の理論入門」サイエンス社 (2018) SGC ライブラリ 140 ISBN:4910054700480
- 花田政範・松浦壮「ゼロからできる MCMC」講談社 (2020) 講談社サイエンティフィック ISBN:978-4-06-520174-9

以下も参照した

- 青木慎也「格子場の理論」シュプリンガー (2005)、丸善出版 (2012)
日本語でかかれた本格的な格子 QCD の教科書
- Istvan Montvay and Gernot Münster 「Quantum Fields on a Lattice」
Cambridge University Press (1994)
30 年以上前の古い教科書だが、初期の理論的な基礎づけが詳しい
- Martin Lüscher 「Computational Strategies in Lattice QCD」 arXiv:1002.4232
Les Houches Summer School: Session 93: Modern perspectives in lattice QCD:
Quantum field theory and high performance computing (2010) 331-399
格子 QCD の理論的指導者の一人による、講義ノート。発展的なアルゴリズムの入口

こちら読み応えがある

- A.D. Kennedy 「Algorithms for dynamical fermions」 arXiv:hep-lat/0607038
World Scientific in proceedings of the ILFTN workshop 'Perspectives in Lattice QCD'
奈良で 2005 年にあったワークショップの講演録、HMC の発明者による理論的な考察

Outline

1. 導入編

- 1.1 イントロ
- 1.2 モンテカルロ法 マルコフ連鎖
- 1.3 メトロポリス法
- 1.4 ハイブリッド・モンテカルロ法 (HMC)

2. 発展編

- 2.1 ゲージ場に対する HMC
- 2.2 フェルミオンに対する HMC
- 2.3 種々の工夫：RHMC, Hasenbusch 法

導入編

何を計算したいのか

- 経路積分を用いた期待値

$$\langle O \rangle = \frac{\int dU O[U] \exp(-S[U])}{Z} \quad \text{ただし } Z = \int dU \exp(-S[U])$$

ここで、 $U = U(\mu, x)$ が物理自由度（リンク変数）

- リンク変数 $U(\mu, x)$ の積分：ものすごい高次元積分 次元の呪い
SU(3) を素朴に 3×3 行列で 9 自由度としても（正しくは独立なのは 8 自由度）

例： $3 \times 3 \times 4 \times 32 \times 32 \times 32 \times 32 \approx 3.8 \times 10^7$ 次元の積分

3×3 行列 4 -dim 格子サイズ

ちなみに国内で最大の計算は 256^4 格子: 1.5×10^{11} 次元

- 積分 \approx 和
1 自由度あたり 8 点の和で近似しても、 $8^{3.8 \times 10^7} \approx 10^{3.4 \times 10^7}$ 点の和 (3,400 万桁)
1 秒に 10^9 点の計算ができるとして $\frac{10^{3.4 \times 10^7}}{10^9} = 10^{3.4 \times 10^7 - 9} \approx 10^{3.4 \times 10^7}$ 秒かかる
計算機のタイムスケール：GHz = 10^9 /sec.
宇宙年齢： 4×10^{17} 秒

次元の呪いに打ち勝って、どうにかして期待値（高次元積分）を計算したい

積分からアンサンブル平均へ

自由度が多すぎて直接扱えない \Rightarrow 統計的計算手法 cf. 統計力学

話を簡単にするために、1次元積分で考える

(必須ではないが) 1次元積分でも統計的手法は使える

アンサンブル平均

確率密度関数 $\rho(x)$ の下での積分をアンサンブル平均で近似する

$$\langle f \rangle = \int dx f(x) \rho(x) = \frac{1}{N} \sum_{i=1}^N f(x_i) + O(1/\sqrt{N})$$

ここで、 $\underbrace{\{x_i\}}_{\text{アンサンブル}}$ の分布は $\rho(x_i)$ に従う

アンサンブル

問題：アンサンブル $\{x_i\}$ をどのように作るか

モンテカルロ法

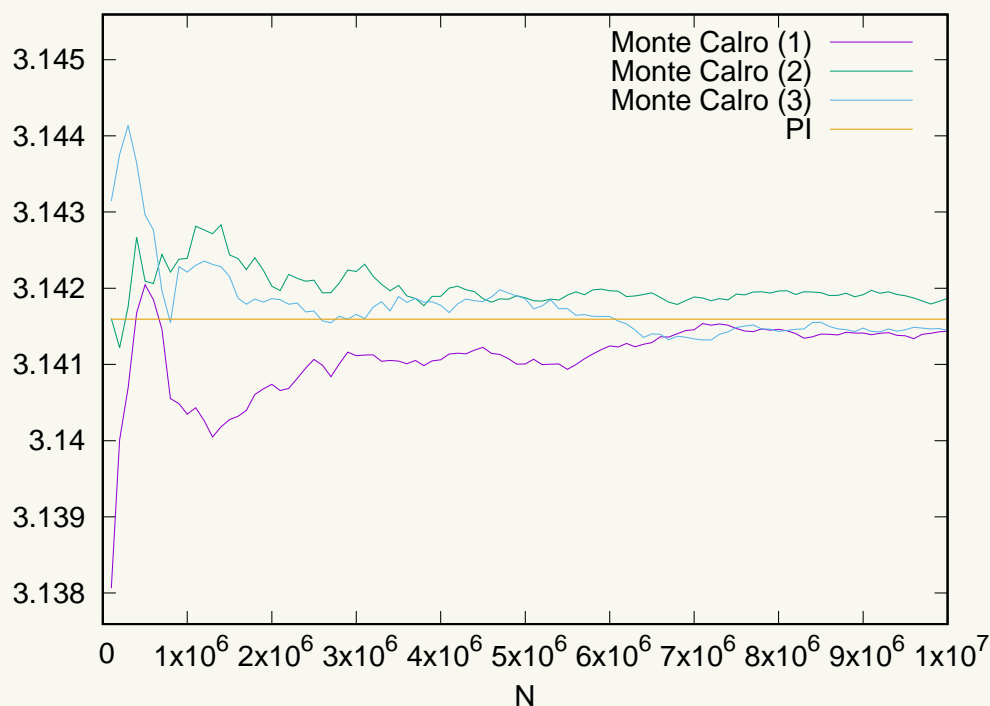
$$\langle f \rangle = \int dx f(x) \rho(x) = \frac{1}{N} \sum_{i=1}^N f(x_i) + O(1/\sqrt{N})$$

例：一様分布

$\rho(x) = \text{constant}$ ($a \leq x \leq b$ とする)

x_i を $a \leq x_i \leq b$ の一様な乱数として $\frac{1}{N} \sum_{i=1}^N f(x_i)$ を計算すれば良い

たとえば $f(x) = 4\sqrt{1-x^2}$ とすると、 $4 \int_0^1 \sqrt{1-x^2} = \pi$ に近づく



サンプルプログラム pi.py

モンテカルロ法：重点サンプリング

例：ガウス分布

$$\int dx f(x) \rho(x) = \frac{1}{N} \sum_{i=1}^N f(x_i) + O(1/\sqrt{N})$$

$$\rho(x) = \frac{1}{\sqrt{2\pi}} \exp(-x^2/2) \quad \langle x^2 \rangle_{\text{ガウス}} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x^2 \exp(-x^2/2)$$

重点サンプリング (importance sampling)

適切な分布に従うアンサンブルを用意して、どの x_i も積分に寄与するようにする
もしガウス分布に従う x_i を用意できれば、

$$\langle x^2 \rangle_{\text{ガウス}} = \frac{1}{N} \sum_{i=1}^N (x_i)^2 + O(1/\sqrt{N})$$

もし一様分布を用いたら (重点サンプリングではない場合)

- $\rho(y) = \text{const.}$, $f(y) = y^2 \exp(-y^2/2)$ として

$$\langle y^2 \exp(-y^2/2) \rangle_{\text{一様}} = \frac{1}{N} \sum_{i=1}^N (y_i)^2 \exp(-y_i^2/2) + O(1/\sqrt{N})$$

- 無限範囲は困るので、適当なカットオフを入れる: $-1000 \leq y_i \leq 1000$
- $y^2 \exp(-y^2/2)$ は $y \simeq 0$ 付近以外はほとんどゼロ: $y = 2$ で 0.54, $y = 3$ で 0.01
- $-3 < y_i < 3$ しか積分に寄与しない
⇒ 1000 個の y_i のうち、997 個は無駄

ガウス分布の作り方: ボックス・ミュラー法

$x = r \cos \theta$, $y = r \sin \theta$, $\theta' = \theta/(2\pi)$, $z = \exp(-r^2/2)$ として、

$$\begin{aligned} \frac{1}{2\pi} \int_{-\infty}^{\infty} dx dy \exp\left[-\frac{x^2}{2} - \frac{y^2}{2}\right] &= \frac{1}{2\pi} \int_0^{2\pi} d\theta \int_0^{\infty} dr r \exp\left[-\frac{r^2}{2}\right] \\ &= \int_0^1 d\theta' \int_0^1 dz \end{aligned}$$

つまり、 θ' と z を $[0, 1]$ の一様乱数として、

$$x = \sqrt{-2 \ln z} \cos(2\pi\theta'), \quad y = \sqrt{-2 \ln z} \sin(2\pi\theta')$$

とすると、 x, y はガウス分布になる

数値的には、 $z = 0$ の時に $\ln z$ が発散して困るので、 $(0, 1]$ の乱数を用いる

再現性に関するマニアックな Tips

一様乱数 2 個から、ガウス乱数 2 個を返している。処理系によって x, y の順番で返すか、 y, x の順番で返すか、異なることがある。C++ にはガウス分布に従う乱数を返す関数 `std::normal_distribution` があるが、同じ乱数列を使っても、処理系によって結果が異なる（そもそもボックス・ミュラー法でないかもしれない）。

QCD のコードで問題になった例 <https://github.com/paboyle/Grid/issues/353>

マルコフ連鎖 (Markov Chain)

マルコフ連鎖

状態 (配位) を

$$x_1 \rightarrow x_2 \rightarrow \cdots \rightarrow x_k \rightarrow \dots$$

と次々と chain 状に作るとき、 x_k が **一つ前の状態 x_{k-1} にしか依存しない** 作り方
マルコフ過程 (Markov process) ともいう

例：ランダムウォーク

サイコロを振って、

- 偶数の目がでたら $x_k = x_{k-1} + 1$
- 奇数の目がでたら $x_k = x_{k-1} - 1$

もし +1 のあとに +1 が出やすいなら、マルコフ連鎖ではない: x_{k-2} の情報も使っている

既約性

有限回のステップで、任意の状態に移れること
既約ではない例

- 2つの島に別れたランダムウォーク
- 積分領域が複数に別れていて、領域間をつなぐステップがない (e.g. 1ステップの変化が少なすぎて、領域間の隙間を飛び越えられない)
注意：完全に別れていなくても分布がほとんどゼロの領域で分割されていると、有限の計算時間では既約としては振る舞わない

非周期性

任意の状態 x_a から、同じ状態 x_a に戻るまでの可能なステップ数の最大公約数をマルコフ連鎖の周期という。周期が1のとき、周期性がないという。

- 一歩が ± 1 のランダムウォーク：2ステップ (+1, -1 or -1, +1)、4ステップ、..、でもとに戻ることが可能⇒周期は2
- 一歩が $[-1, 1]$ の乱数によるランダムウォーク：歩幅が0のときに、1ステップでもとに戻れる⇒周期性がない (非周期性)

補足：任意の状態 x_s から出発して有限ステップ数で必ず x_s に戻ってくるとき、「再帰的」であるという。一般に、マルコフ連鎖で生成した状態の分布がある分布に一意に収束するためには、既約性・非周期性を満たす他に再帰的であることが必要。ただし計算機でのシミュレーションのように状態空間が有限なときは自動的に満たされるため、気にする必要はない。

詳細均衡 (Detailed Balance)

これまで：分布関数 $\rho(x)$ についての情報は使っていない

欲しい分布をどうやって作るか： **詳細均衡 (detailed balance) を満たせば良い**

- 分布を熱平衡状態の分布と捉えたと、ミクロには変化してもマクロな分布は平衡であれという条件
- 欲しい分布を得るための十分条件：詳細均衡を満たさないアルゴリズムもある

以下、確率密度関数 $\rho(x)$ を $P(x)$ と書くことにする： $P(x) = \rho(x)$

ある x から x' への遷移確率を $T(x \rightarrow x')$ とすると、

$$\text{詳細均衡条件： } P(x)T(x \rightarrow x') = P(x')T(x' \rightarrow x)$$

を満たすとき、系の分布は $P(x)$ になる

詳細均衡 (Detailed Balance) : 直感的な説明

ある x から x' への遷移確率を $T(x \rightarrow x')$ とすると、

$$\text{詳細均衡条件 : } P(x)T(x \rightarrow x') = P(x')T(x' \rightarrow x)$$

を満たすとき、系の分布は $P(x)$ になる

具体例

$x = 1, 2, \dots, 10$ で 10000 個の粒子を 10 個の箱に散らばっている x は箱のラベル

- $x = 1$ である確率が 10% : $P(x = 1) = 0.1$, $x = 1$ にある箱に 1000 個の粒子
- $x = 10$ である確率が 20% : $P(x = 10) = 0.2$, $x = 10$ にある箱に 2000 個の粒子

とする。 $T(1 \rightarrow 10) = 0.1$ のとき、詳細均衡を満たすなら、 $T(10 \rightarrow 1) = 0.05$

- $x = 1$ かつ、 $x' = 10$ である確率 : $P(x = 1)T(1 \rightarrow 10) = 0.01$
 $x = 1$ にいた 1000 個の粒子のうち、次のステップで $x = 10$ に移動したのは 100 個
- $x = 10$ かつ、 $x' = 1$ である確率 : $P(x = 10)T(10 \rightarrow 1) = 0.01$
 $x = 10$ にいるのは 2000 個の粒子のうち、次のステップで $x = 1$ に移動したのは 100 個

ちょうど同じ数の粒子 (100 個) が $x = 1$ と $x = 10$ で入れ替わったので、この分布から変わらない

詳細均衡が目的の分布を与えること

ステップ i での確率分布を $P_i(x)$ とする。ステップ $i+1$ での確率分布は、

$$\begin{aligned} P^{(i+1)}(x) &= \sum_y P^{(i)}(y) T(y \rightarrow x) && i \text{ での状態を足し合わせ} \\ &= \sum_y P^{(i)}(x) T(x \rightarrow y) && \text{detailed balance} \\ &= P^{(i)}(x) && \sum_x T(x \rightarrow y) = 1 \quad (\text{必ず行き先がある}) \end{aligned}$$

となつて分布は変わらない。

一方、遷移確率を行列 T_{xy} とみると、初期分布 $P_y^{(0)}$ をベクトル見て N ステップ進んだときの分布は

$$(T)_{xy}^N P_y^{(0)} \xrightarrow{N \gg 1} T_{xy} \text{ の最大固有値に属する固有ベクトル}$$

この「 T_{xy} の最大固有値に属する固有ベクトル」が詳細均衡を満たす状態だということが示せる（ペロン・フロベニウスの定理からの帰結）

メトロポリス法

詳細均衡を満たす分布を作る方法の一つ

目標とする確率分布関数を $P(x)$ とする [例: $P(x) = \exp[-H(x)]/Z$].

- 状態 x_i から、新しい状態の候補 y を作る
- 係数 $u = P(y)/P(x_i)$ を計算する [例: $u = \exp[-\Delta H]$, $\Delta H = H(x_i) - H(y)$]
- 一様乱数 $r \in [0, 1]$ を生成し、
 - $r \leq u$ なら $x_{i+1} = y$ accept
 - $r > u$ なら $x_{i+1} = x_i$ reject やり直すのではなく、次に進む

とする (メトロポリス・テストによる、accept/reject)

$\Delta H \leq 0$ なら無条件に新しい候補 y を採用、 $\Delta H > 0$ なら確率 $\exp(-\Delta H)$ で y を採用

ここでは、状態 x_i から y を作る確率 $Q(y, x_i)$ と、 y から x_i を作る確率 $Q(x_i, y)$ が等しいとした。一般には、 $u = P(y)Q(y, x_i)/\{P(x_i)Q(x_i, y)\}$ となる (メトロポリス・ヘイスティング法)。

メトロポリス法が詳細均衡を満たすこと

一般の $u = P(y)Q(x_i, y)/\{P(x_i)Q(y, x_i)\}$ で考える。accept されるとき遷移確率 $T(x_i \rightarrow y)$ は、

- $u \geq 1$ のとき、 $T(x_i \rightarrow y) = Q(y, x_i)$
 $P(x_i)T(x_i \rightarrow y) = P(x_i)Q(y, x_i) = \frac{1}{u}P(y)Q(x_i, y)$ (このとき、 $1/u \leq 1$)
- $u < 1$ のとき、確率 u で y が採用されるので、 $T(x_i \rightarrow y) = uQ(y, x_i)$
 $P(x_i)T(x_i \rightarrow y) = uP(x_i)Q(y, x_i) = P(y)Q(x_i, y)$ (このとき、 $1/u > 1$)

まとめて

$$P(x_i)T(x_i \rightarrow y) = \min(1, u)P(x_i)Q(y, x_i) = \min(1, 1/u)P(y)Q(x_i, y)$$

一方、

$$P(y)T(y \rightarrow x_i) = \min(1, 1/u)P(y)Q(x_i, y)$$

このことから、

$$P(x_i)T(x_i \rightarrow y) = P(y)T(y \rightarrow x_i) \text{ が成立}$$

reject されたときは、 $x_{i+1} = x_i$ で自明に成り立つ場合に含まれる。

$x_i \rightarrow x_i$ の場合を丁寧に示すと、

- y 経由で reject されて $x_{i+1} = x_i$ になった
- たまたま $y = x_i$ (これは常に採用される) で $x_{i+1} = x_i$ になった

のそれぞれについて、遷移確率を求めればよい

分布を統計力学での分布関数とみなして、ちょっと飛躍してみる

分子動力学 (Molecular Dynamics, MD)

多自由度の力学系（例：多数の分子からなる系）の運動方程式を解いて時間発展を追いかける

- ある程度時間が経てば、平衡状態になる（エネルギーは一定：ミクロカノニカル）
- 配位空間 (粒子の座標が張っている空間) の中を一本の線に乗って動いていく
- 亜種：運動量をときどきランダムに変える
配位空間の中でランダム・ウォーク、エネルギーも揺らぐ（カノニカル）

欠点：数値的に時間発展を追う際に、離散化誤差がある

このままでは有限な時間刻み δt を 0 に外装する必要がある

有限 δt からくるズレを、メトロポリス法で補正できないか？

⇒ハイブリッド・モンテカルロ法 (HMC)

ハイブリッド・モンテカルロ法 (Hybrid Monte Carlo, HMC)

ハミルトニアン・モンテカルロとも呼ばれる

準備 1変数のように書いているが、 x や π は複数の変数を束ねたベクトル

- 欲しい分布 $\exp(-S[x])/Z \propto \exp(-S[x])$
- ガウス積分 $\int d\pi \exp[-\frac{1}{2}\pi^2] = \text{const}$ を分子分母にかけても、 x の分布は変わらない
 $\Rightarrow \propto \exp(-\frac{1}{2}\pi^2 - S), H \equiv \frac{1}{2}\pi^2 + S$

ハイブリッド・モンテカルロ法

1. π をガウス分布で生成する。始状態は $x = x_i$
2. $\{\pi, x\}$ を、 H のもとで、有限時間 Δt 時間発展させる

$$\text{運動方程式: } \dot{\pi} = -\frac{\partial H}{\partial x}, \dot{x} = \frac{\partial H}{\partial \pi}$$

時間発展の離散化は後で

3. メトロポリス・テスト: 一様乱数 $r \in [0, 1]$ と $\Delta H \equiv H[\pi', x'] - H[\pi, x]$ で
 - $\exp[-\Delta H] \leq r$ なら、 x' を新しい x として採用: $x_{i+1} = x'$
 - $\exp[-\Delta H] > r$ なら、 x' は棄却: $x_{i+1} = x_i$
- ここでいう「時間」は仮想的な時間 (fictitious time)
力学変数を $x = \phi(\vec{x}, t)$ としたときの場の理論の時間とは別物
- Δt は任意だが、 $\Delta t = 1$ ととることが多い
- 採用率が 0.8 ± 0.1 ぐらいになるように時間発展の離散化のパラメータを選ぶ
(具体的には離散化の詳細による)

x についての詳細均衡

$H[\pi', x']$ は π' について対称なので、メトロポリス・テストの直前に終状態の運動量を反転する操作 $R(\pi') = -\pi'$ を加えて考える

遷移確率は

$$T(x \rightarrow x') = C \int d\pi d\pi' T(\{\pi, x\} \rightarrow \{-\pi', x'\}) \exp(-\pi^2/2) \quad (C : \text{定数})$$

$$\begin{aligned} T(x \rightarrow x') \exp(-S[x]) &= C \int d\pi d\pi' T(\{\pi, x\} \rightarrow \{-\pi', x'\}) \underbrace{\exp(-\pi^2/2) \exp(-S[x])}_{=\exp(-H[\pi, x])} \\ &= C \int d\pi d\pi' T(\{-\pi', x'\} \rightarrow \{\pi, x\}) \exp(-H[\pi', x']) \\ &= C \int d\pi d\pi' T(\{-\pi', x'\} \rightarrow \{\pi, x\}) \exp(-\pi'^2/2) \exp(-S[x']) \\ &= T(x' \rightarrow x) \exp(-S[x']) \end{aligned}$$

途中で $\{\pi, x\}$ についての詳細均衡

$T(\{\pi, x\} \rightarrow \{-\pi', x'\}) \exp(-H[x]) = T(\{-\pi', x'\} \rightarrow \{\pi, x\}) \exp(-H[x'])$
を仮定した (次のスライド)

{ π, x } についての詳細均衡

$$T(\{\pi, x\} \rightarrow \{-\pi', x'\}) \exp(-H[x]) = T(\{-\pi', x'\} \rightarrow \{\pi, x\}) \exp(-H[x'])$$

分布 $\exp[-H[\pi, x]]$ のみを用いてメトロポリス・テストをしているので、さっき示した一般論を適用するにはテスト配位 $\{\pi', x'\}$ を作る確率が対称

$$Q(\{\pi, x\} \rightarrow \{-\pi', x'\}) = Q(\{-\pi', x'\} \rightarrow \{\pi, x\})$$

であることが必要。これを示す。

$U(\Delta t)$ を時間発展、 R を運動量反転として、

- $RU(\Delta t)\{\pi, x\} = R\{\pi', x'\} = \{-\pi', x'\}$
- 時間反転対称性があれば、 $RU(\Delta t)\{-\pi', x'\} = R\{-\pi, x\} = \{\pi, x\}$
- 上記の操作で $\{\pi, x\} \rightarrow \{-\pi', x'\}$ の確率と $\{-\pi', x'\} \rightarrow \{\pi, x\}$ は同じようにみえるが、 Q としては欲しい関係式は

$$\delta(\{-\pi', x'\} - RU(\Delta t)\{\pi, x\}) = \delta(\{\pi, x\} - RU(\Delta t)\{-\pi', x'\})$$

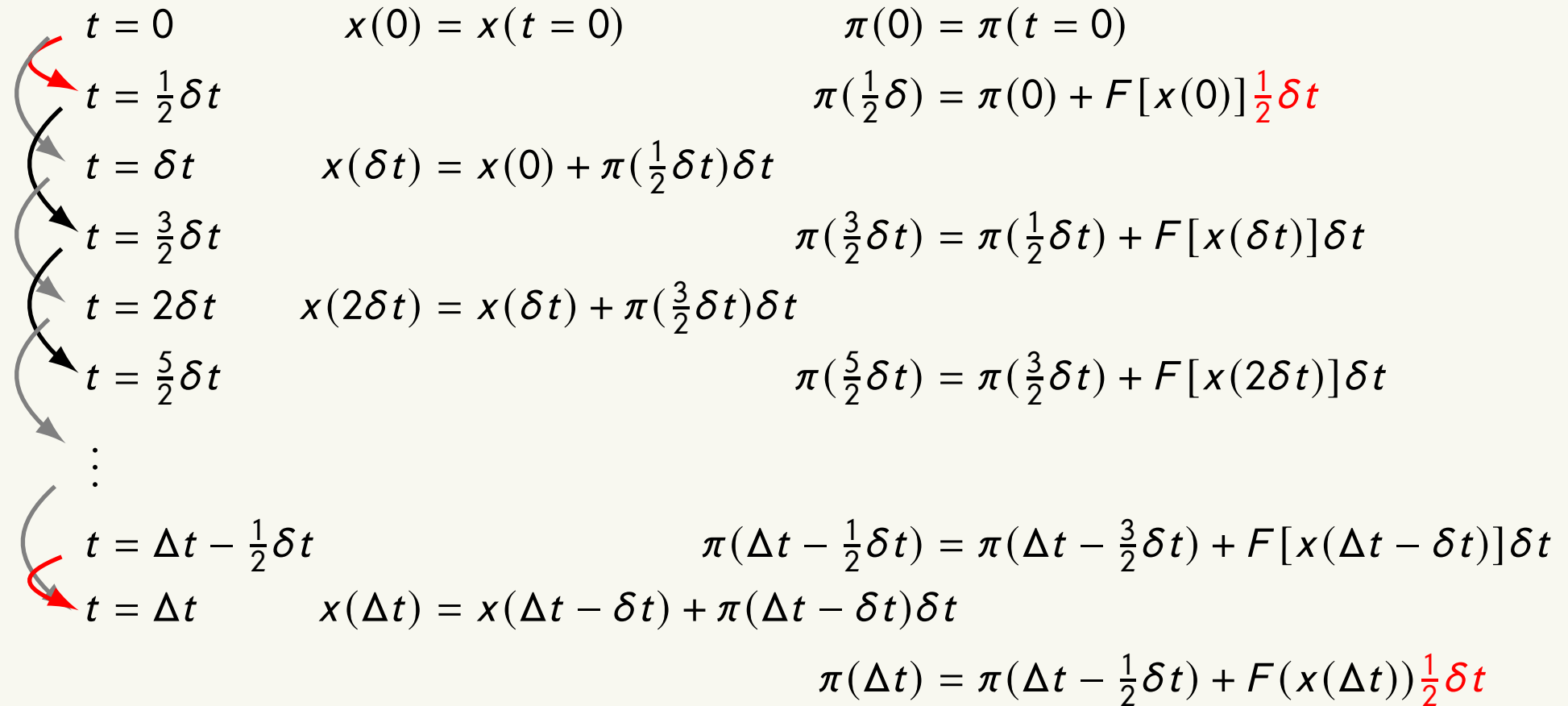
δ -関数の変数変換になっているので、変換のヤコビアン ($RU(\Delta t)$ の行列式) が 1 であれば OK

ここで使ったもの

- 時間反転対称性
- 時間発展演算子の行列式が 1 (area preserving; cf. リウヴィルの定理)
area preserving から $\langle \exp(-\Delta H) \rangle = 1$ が導かれ、シミュレーションのチェックによく使われる

時間発展：リープ・フロッグ

$\dot{\pi} = -\frac{\partial H}{\partial x} \equiv F[x]$, $\dot{x} = \frac{\partial H}{\partial \pi}$ のもとで、 $\{\pi, x\}$ を $t = 0$ から $t = \Delta t$ まで時間発展
 $\delta t = \Delta t/N$ とする（最初と最後の π の時間発展は $\frac{1}{2}\delta t$ ）



$t = 0$	$x(0) = x(t = 0)$	$\pi(0) = \pi(t = 0)$
$t = \frac{1}{2}\delta t$		$\pi(\frac{1}{2}\delta t) = \pi(0) + F[x(0)]\frac{1}{2}\delta t$
$t = \delta t$	$x(\delta t) = x(0) + \pi(\frac{1}{2}\delta t)\delta t$	
$t = \frac{3}{2}\delta t$		$\pi(\frac{3}{2}\delta t) = \pi(\frac{1}{2}\delta t) + F[x(\delta t)]\delta t$
$t = 2\delta t$	$x(2\delta t) = x(\delta t) + \pi(\frac{3}{2}\delta t)\delta t$	
$t = \frac{5}{2}\delta t$		$\pi(\frac{5}{2}\delta t) = \pi(\frac{3}{2}\delta t) + F[x(2\delta t)]\delta t$
\vdots		
$t = \Delta t - \frac{1}{2}\delta t$		$\pi(\Delta t - \frac{1}{2}\delta t) = \pi(\Delta t - \frac{3}{2}\delta t) + F[x(\Delta t - \delta t)]\delta t$
$t = \Delta t$	$x(\Delta t) = x(\Delta t - \delta t) + \pi(\Delta t - \frac{1}{2}\delta t)\delta t$	
		$\pi(\Delta t) = \pi(\Delta t - \frac{1}{2}\delta t) + F(x(\Delta t))\frac{1}{2}\delta t$

時間反転に対して対称 (reversibility)

cf. ルンゲ・クッタは時間反転に対して対称ではない

area preserving も成り立つ

x の更新と、 π の更新は、

$$\begin{pmatrix} x' \\ \pi' \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & \delta t \\ 0 & 1 \end{pmatrix}}_{\det=1} \begin{pmatrix} x \\ \pi \end{pmatrix} \qquad \begin{pmatrix} x' \\ \pi' \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 \\ \delta t F \frac{\partial}{\partial x} & 1 \end{pmatrix}}_{\det=1} \begin{pmatrix} x \\ \pi \end{pmatrix}$$

ヤコビアンが 1 なので、area preserving も成り立つ

- リープ・フロッグは Hamiltonian の保存に対して $(\delta t)^2$ の誤差がある。より誤差が少ない積分が使われることもある
- 一般に、ヤコビアンが 1 の時間発展はシンプレクティック積分と呼ばれ、もとの Hamiltonian から少しずれた Shadow Hamiltonian が（有限の δt でも）厳密に保存している。数値的には、有限桁の浮動小数点演算の精度の範囲で保存する

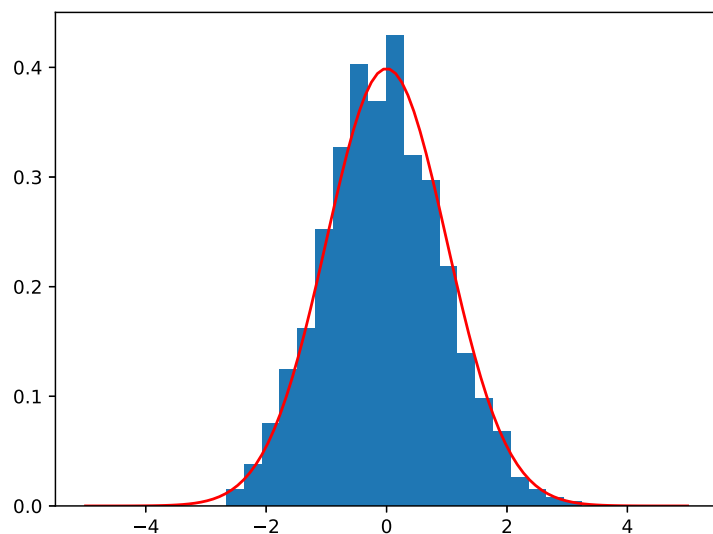
ガウス積分での例

$\frac{1}{\sqrt{2\pi}} \int dx x^2 \exp(-x^2/2)$ を HMC で計算する

分布 $P(x) = \exp(-x^2/2)/Z$ のもとで $\langle x^2 \rangle$

- $S = \frac{1}{2}x^2, H = \frac{1}{2}\pi^2 + \frac{1}{2}x^2$
- 運動方程式: $\dot{\pi} = -x, \dot{x} = \pi$
- MD のパラメータは、 $\Delta t = 1, \delta t = \Delta t/2$ とした
- 初期配位は $x = 0$, 最初の 100 ステップは thermalization として捨てて、その後の 900 ステップで $\frac{1}{N} \sum_i (x_i)^2$ を計算
 $\langle x^2 \rangle = 0.954(43)$ (厳密値は 1.0)

注意: 自己相関を考慮していないので、誤差は過小評価の可能性あり



HMC で作った分布

曲線は $\frac{1}{\sqrt{2\pi}} \exp(-x^2/2)$

サンプルプログラム `hmc_gauss.py`

発展編

ゲージ場

- 力学変数：SU(3) 群に値をもつ 3×3 複素行列 $U(x, \mu)$
- 共役運動量 $\pi(x, \mu)$ を以下で定義する

$$\dot{U}(x, \mu) = i\pi(x, \mu)U(x, \mu), \quad \pi(x, \mu)^\dagger = \pi(x, \mu), \quad \text{tr}(\pi(x, \mu)) = 0$$

左端のドット付き U は、MD time (fictitious time) についての時間微分

- Hamiltonian は

$$H = \frac{1}{2} \sum_{x, \mu} \text{tr} [\pi(x, \mu)^2] + S_G(U)$$

- 運動方程式は、 H ドットがゼロ ($\dot{H} = 0$) で定める
- ゲージ場の時間発展: $U(x, \mu)$ がユニタリになるように定める

$$U(x, \mu)' = \exp[i\delta t \pi(x, \mu)]U(x, \mu) \quad (= GU(x, \mu), G \in \text{SU}(3))$$

- ゲージ場の更新は、左から SU(3) 行列をかける操作になっている。これは Haar 測度を不変に保つ
もし measure が不変でなければ、メトロポリス・テストにその効果を入れる必要がある

フェルミオン場の積分: ボソン場 $\phi, \bar{\phi}$ (擬フェルミオンと呼ばれる) で書き換える

$$\begin{aligned} \int d\psi d\bar{\psi} dU \exp(-\bar{\psi} D[U] \psi - S_{\text{gauge}}[U]) &= \det D[U] \int dU \exp -S_{\text{gauge}}[U] \\ &= \int d\bar{\phi} d\phi dU \exp(-\bar{\phi} D[U]^{-1} \phi - S_{\text{gauge}}[U]) \end{aligned}$$

ここでは擬フェルミオンでの積分は収束している仮定 (一般には正しくない)

- 質量が同じフェルミオンが n_f 個あると、 $(\det D[U])^{n_f}$ になる
- $n_f = 2$ のとき (isospin 対称性がある u -quark と d -quark が縮退)
 $(\det D)^2 = (\det D)(\det D) = (\det \underbrace{\gamma_5 D \gamma_5}_{=D^\dagger})(\det D) = \det(D^\dagger D)$
- $\gamma_5^2 = 1$, $\det(AB) = \det(A) \det(B) = \det(B) \det(A) = \det(BA)$
- Dirac 演算子の性質: $\gamma_5 D \gamma_5 = D^\dagger$

とくに 2 flavor のとき、

$$\int d\psi_f d\bar{\psi}_f \exp(-\sum_{f=1,2} \bar{\psi}_f D[U] \psi_f) = \int d\bar{\phi} d\phi \exp(-\bar{\phi} (D^\dagger D)^{-1} \phi)$$

$$\int d\bar{\phi}d\phi \exp(\underbrace{-\bar{\phi}(D^\dagger D)^{-1}\phi}_{S_{\text{pf}}}) \exp(-S_{\text{gauge}}[U])$$

2-flavor で考える

1. ガウス分布の複素場 η を作り、擬フェルミオンを $\phi = D^\dagger \eta$ とする
2. 実数場 π^a をガウス分布で作ри、共役運動量を $\sum_a \pi^a T_a$ とする
(T_a は SU(3) の生成子)
3. 初期状態の Hamiltonian $H(U = U_i, \pi)$ を計算する
4. $S_{\text{pf}} + S_{\text{gauge}}$ で時間発展させる (MD) : $\{U, \pi\} \rightarrow \{U', \pi'\}$
5. 終状態の Hamiltonian $H(U', \pi')$ を計算する
6. Hamiltonian の変化 ΔH を計算し、メトロポリス・テストを行う
採用なら、 $U_{i+1} = U'$, 棄却なら $U_{i+1} = U_i$

フェルミオンがある時の HMC: 特徴

フェルミオンがあるとき：擬フェルミオン (pseudo fermion) を導入

$$S = S_{\text{pf}} + S_{\text{gauge}}, \quad S_{\text{pf}} = \bar{\phi} (D^\dagger D)^{-1} \phi \quad (n_f = 2)$$

MD force の計算や作用の計算で、 $(D^\dagger D)^{-1} \phi$ が必要

- 「フェルミオンが入ると 100 倍遅くなる」
- Conjugate Gradient (CG) 法などの反復法で計算 (HMC で最も時間がかかる部分)
- 一般に am が小さいほど (= クォークの質量 m や格子間隔 a が小さいほど) 時間がかかる
- Dirac 演算子が複雑だと時間がかかる
staggered (速い) \ll Wilson(Colver) \ll Domainwall \ll overlap
- $(D^\dagger D)^{-1} \phi$ を求めるソルバーは石川さんの講義で

s-quark の扱い: 奇数 flavor, Majorana (SUSY)

- $\det D$ に必要な D^{-1} を $D^\dagger D$ の有理式近似 (Rational approximation) で表す

$$D^{-1} = (\det D^\dagger D)^{-1/2} \simeq a_0 + \sum_{i=1}^n \frac{a_i}{D^\dagger D + b_i}$$

- n 本の方程式 $(D^\dagger D + b_i)\chi_i = \phi$ を同時に解くアルゴリズムが知られている (multi-shift solver)
- 係数 a_i, b_i を求めるプログラムも公開されている

M.A. Clark and A.D. Kennedy, <https://github.com/mikeaclark/AlgRemez>, 2005

Multi time step

作用の各項ごとに、MD の時間発展の δt を変えても良い

$$H = \frac{p^2}{2} + S_1 + S_2 \text{ とする}$$

- $I_U(\delta t)$: p を用いて U を δt 進める
- $I_1(\delta t)$: $S_1[U]$ を用いてフォース F_1 を計算し、 $p \rightarrow p - F_1 \delta t$ とする
- $I_2(\delta t)$: $S_2[U]$ を用いてフォース F_2 を計算し、 $p \rightarrow p - F_2 \delta t$ とする

アルゴリズム (leap frog with multi time step)

- S_1 由来の Leap Frog の微小ステップを N_1 回繰り返す:
 $[I_1(\delta t/2)I_U(\delta t)I_1(\delta t/2)]^{N_1} \quad N_1 \delta t$ 進む
- S_2 由来の効果を足す:
 $J(N_1, \delta t) \equiv I_2(N_1 \delta t/2) [I_1(\delta t/2)I_U(\delta t)I_1(\delta t/2)]^{N_1} I_2(N_1 \delta t/2)$
- これを N_2 回くり返す: $[J(N_1, \delta t)]^{N_2}$

このとき、系は $\Delta t = N_1 N_2 \delta t$ 進む

- F_1 を評価する回数: $N_2(N_1 + 1)$ 、 p への寄与は $F_1 \delta t$
 - F_2 を評価する回数: $N_2 + 1$ 、 p への寄与は $N_1 F_2 \delta t$
- ⇒
- S_1 (内側): 計算が楽でフォース F_1 が大きいもの (ゲージ場)
 - S_2 (外側): 計算が大変でフォース F_2 が小さいもの (擬フェルミオン)

Integrator の詳細 (またはコードの都合) によっては、 N_i に制約がつくことがある。

ゲージ場と比べてフェルミオンのフォースは揺らぎ大きい
フォースが大きくなった時に合わせて δt を選ぶと、刻み幅が小さくなりすぎる

$$\det(D(m)^\dagger D(m)) = \det \frac{D(m)^\dagger D(m)}{D(m_1)^\dagger D(m_1)} \det(D(m_1)^\dagger D(m_1))$$

として、擬フェルミオン作用を $S_{\text{pf}} = S_1 + S_2$ とする: もとは $\bar{\phi}[D(m)^\dagger D(m)]^{-1}\phi$ だった

$$S_2 = \bar{\phi}_2 \frac{D(m_1)^\dagger D(m_1)}{D(m)^\dagger D(m)} \phi_2 \quad S_1 = \bar{\phi}_1 \frac{1}{D(m_1)^\dagger D(m_1)} \phi_1$$

このとき、

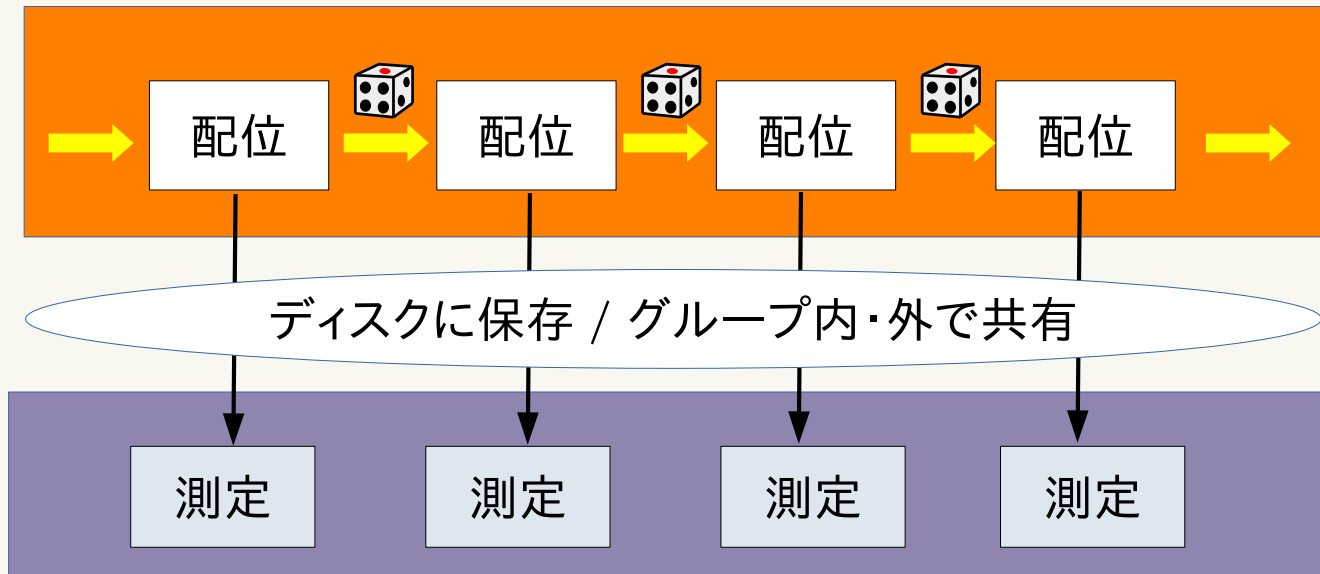
- $m < m_1$ と選ぶと、 S_1 中の $[D(m_1)^\dagger D(m_1)]^{-1}$ の計算は $[D(m)^\dagger D(m)]^{-1}$ より速い
- S_2 は $[D(m)^\dagger D(m)]^{-1}$ があるため計算に時間がかかるが、 $\frac{D(m_1)^\dagger D(m_1)}{D(m)^\dagger D(m)} \simeq 1$ のためフォースは小さい (フォースの揺らぎも小さい)

multi time step と組み合わせて高速化ができる (S_1 : 刻み幅小、 S_2 : 刻み幅大)
さらに $m < m_1 < m_2 < \dots$ なる m_2, \dots を導入して3項以上に分けても良い

まとめ

配位生成のアルゴリズム: サイコロの中身

- 経路積分を、アンサンブル平均に置き換える: $\exp(-S)$ の分布に従った配位を作る
- 配位の作り方: HMC (マルコフ連鎖モンテカルロ法的一种)
- フェルミオンは擬フェルミオン (pseudo fermion) として扱う
cf. Dirac 方程式のソルバー



話せなかったこと、配位生成に関するさらに発展的な話題

- 乱数
 - 第 3 回 HPC-Phys 勉強会 松本さん・斎藤さんの講演スライド
 - https://hpc-phys.kek.jp/workshop/workshop190318/saito_190318.pdf
- 熱浴法 基本事項なのですが、時間の都合で割愛。フェルミオンには使えない
- 自己相関、臨界減速 [ハンズオン?] これも基本事項
- HMC でのパラメータの選び方 ノウハウ的な要素もあります
- MD での高次の積分法
- 作用の書き換え: even-odd 分割, 領域分割
 - 計算機の発展に合わせて最適なアルゴリズムも変化します
- master field, trivialization map 発展途上、臨界減速対策
- 機械学習の利用、self-learning Monte Carlo [富谷さんの講義] 発展途上
- 符号問題がある時の配位生成 発展途上: Complex Langevin, 世界体積 HMC etc.
- ベイズ統計への応用
- 効率の良いアルゴリズムの重要性 研究は競争、電気代、計算機の費用