Monte Carlo法

計算機アルゴリズム特論：2017年度

只木進一

Monte Carlo法

- 狭義
  - 乱数を用いた積分（和）計算
- 広義
  - 乱数を用いたアルゴリズム・シミュレーション技法

疑似乱数(Pseudo random numbers)

- コンピュータ内で乱数を発生させる
- なんらかのアルゴリズム(algorithm)が必要
  - つまり、相関(出現した数値と次の数値の関係)がある
- 同じ乱数を何度も発生できる

線形合同法(Linear Congruential Method)

- 漸化式:$i_n = (a i_{n-1} + c) \bmod m$
  - 定数の選び方で性質が大きく異なる。
  - 良い定数は経験的に知られている。
- C/C++は符号なし整数が使える
  - Overflow制御が不要
- FORTRANのような符号なし整数がない言語
  - Overflowが起きないような工夫が必要

- 32ビット符号なし演算の場合のパラメタ例

$$a = 2416,\ c = 374441,\ m = 1771875$$

- 32ビット符号あり演算の場合のパラメタ例

$$a = 9301,\ c = 49297,\ m = 233280$$

Schrageの方法

- 32ビット符号あり演算
  - オーバーフローを避けて$m=2^{31}-1$とする
- 与えられたmに対して

$$(a=17807,\ c=0,\ m=2^{31}-1)$$

$$q=\lfloor m/a \rfloor,\ r=m \bmod a$$

$$m=aq+r$$

- $r < q$の条件が必要

$$ai_n \bmod m = \begin{cases} a(i_n \bmod q) - \lfloor i_n / q \rfloor r & \text{if not negative} \\ a(i_n \bmod q) - \lfloor i_n / q \rfloor r + m & \text{otherwise} \end{cases}$$

- なぜなら$i_n = xq + y$とすると
  - 右辺：$ay - xr$
  - 一方

$$ai_n = a(xq + y) = xaq + ay = x(m - r) + ay$$
$$= xm + ay - xr$$

線形合同法の問題点

- ある数aが発生すると次が一意に決まっている。
- 周期がm
  - ある種のシミュレーションでは不足
- 多次元疎結晶構造
  - 連続するn個の乱数を一つのn次元空間の座標とすると、パラメタによっては、結晶構造が見える

Monte Carlo法の例：πの計算

- 一辺の長さ1の正方形内に2次元乱数を生成：(x, y) $(0 \le x, y < 1)$
- 乱数が半径1の扇形に入る $\left(0 \le (x^2 + y^2)^{\frac{1}{2}} < 1\right)$ 確率
  - 正方形に対する扇形の面積の比$\pi/4$

- N個の2次元乱数のうち、m個が扇形に入る確率$P_N(m)$は二項分布

$$P_N(m) = \binom{N}{m} p^m (1-p)^{N-m}, \; p = \frac{\pi}{4}$$

- 確率母関数を使うと平均等が計算できる

$$G(z) = \sum_{m=0}^{N} P_N(m) z^m = (1-p+pz)^N$$

$$G'(z)=\sum_{m=0}^{N} mP_N(m)\,z^{m-1}=Np(1-p+pz)^{N-1}$$

$$\langle m\rangle = G'(1)=Np$$

$$G''(z)=\sum_{m=0}^{N} m(m-1)P_N(m)\,z^{m-2}$$

$$=N(N-1)\,p^2(1-p+pz)^{N-2}$$

$$\langle m^2\rangle-\langle m\rangle = G''(1)=N(N-1)\,p^2$$

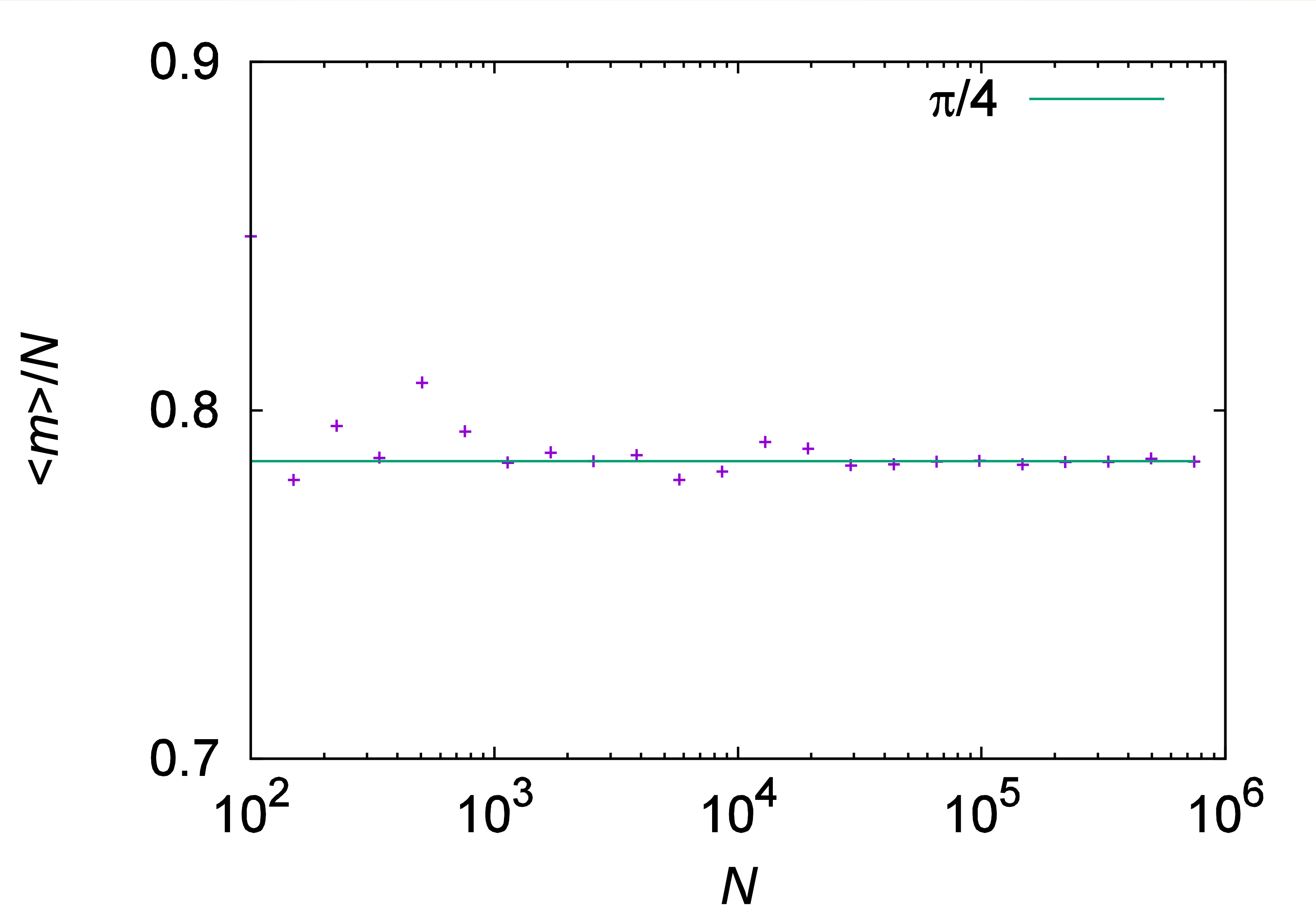
$$\sigma^2=\langle m^2\rangle-\langle m\rangle^2=N(N-1)\,p^2+Np-N^2p^2$$

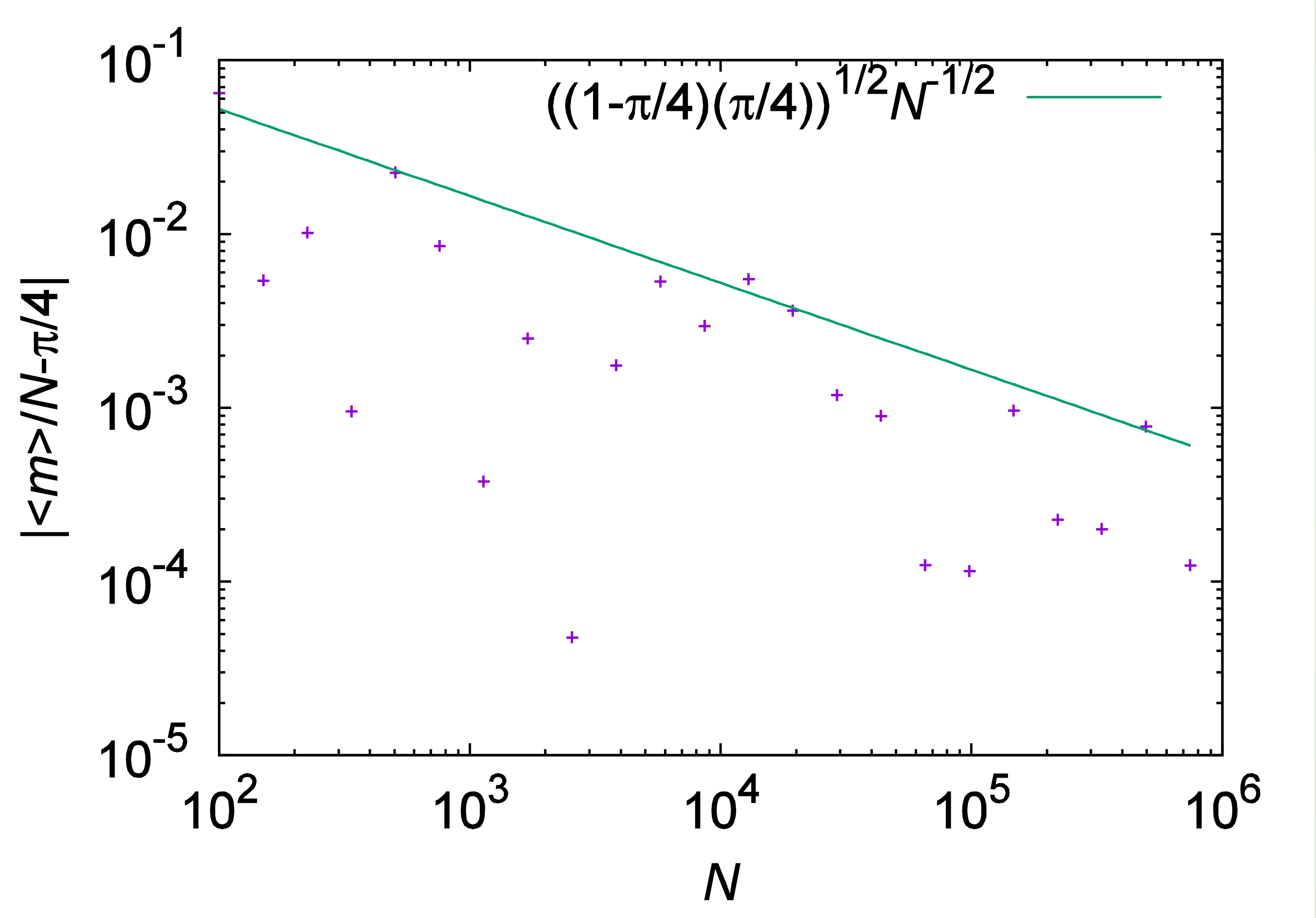
$$=Np(1-p)$$

12

これが正しいことを確かめる ただし、簡単に

- $\langle m \rangle_{\mathrm{exp}}/N \sim p \cong \pi/4$
- $\frac{\sigma}{\langle m \rangle} = \left(\frac{1-p}{p}\right)^{1/2} N^{-1/2}$より
  - $\left|\langle m \rangle_{\mathrm{exp}}/N - \frac{\pi}{4}\right|$が$N^{-1/2}$でゼロに近づく

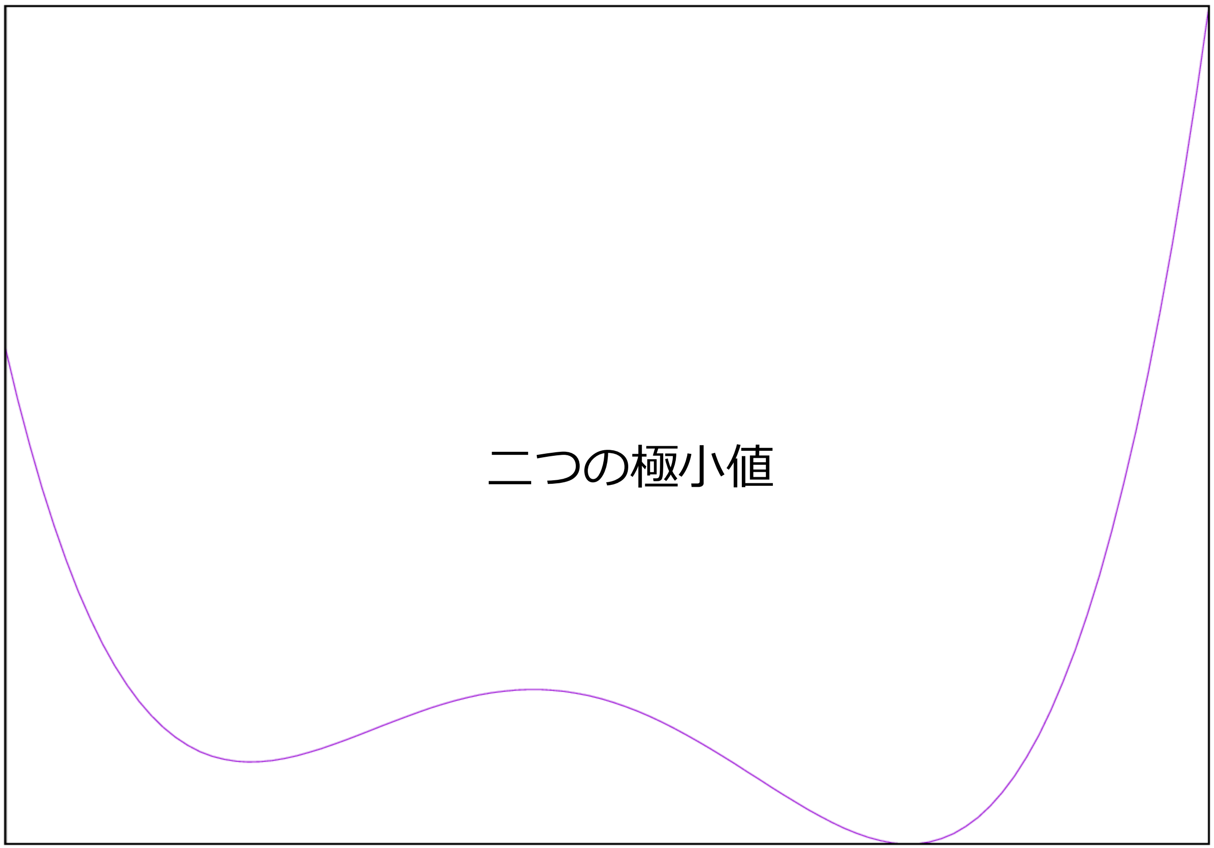

0.9
0.8
0.7
<m>/N
π/4
10^2
10^3
10^4
10^5
10^6
N

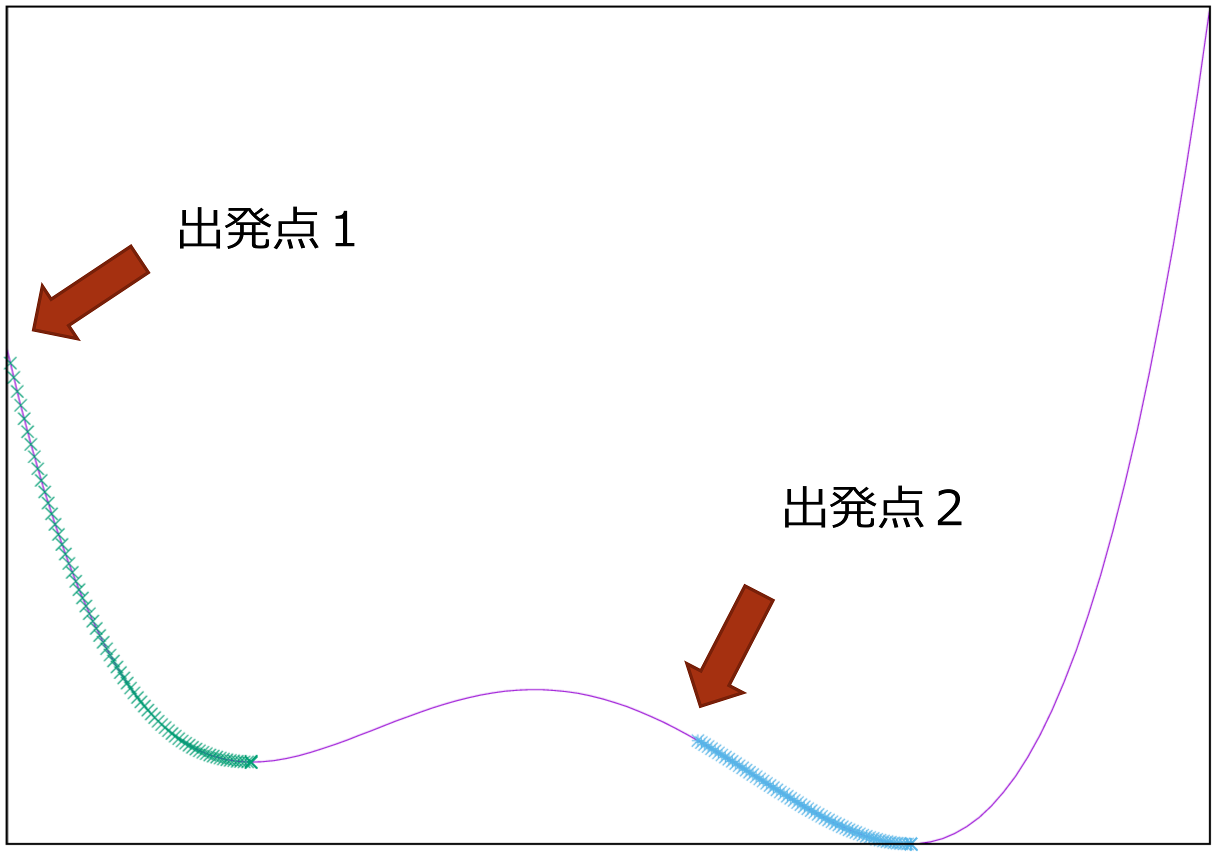


$((1-\pi/4)(\pi/4))^{1/2}N^{-1/2}$
$|\langle m\rangle/N-\pi/4|$
N
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{2}
10^{3}
10^{4}
10^{5}
10^{6}

例：複数の極値を有する場合

- 関数$f(x)$の極小値を求める
  - 極小値が一つならば、適当な出発点xから$f(x)$の値が小さくなるように、xを少しずつ変化させる
  - 極小値が複数ならば
    - 出発点をランダムに選んで、複数回試行する


二つの極小値

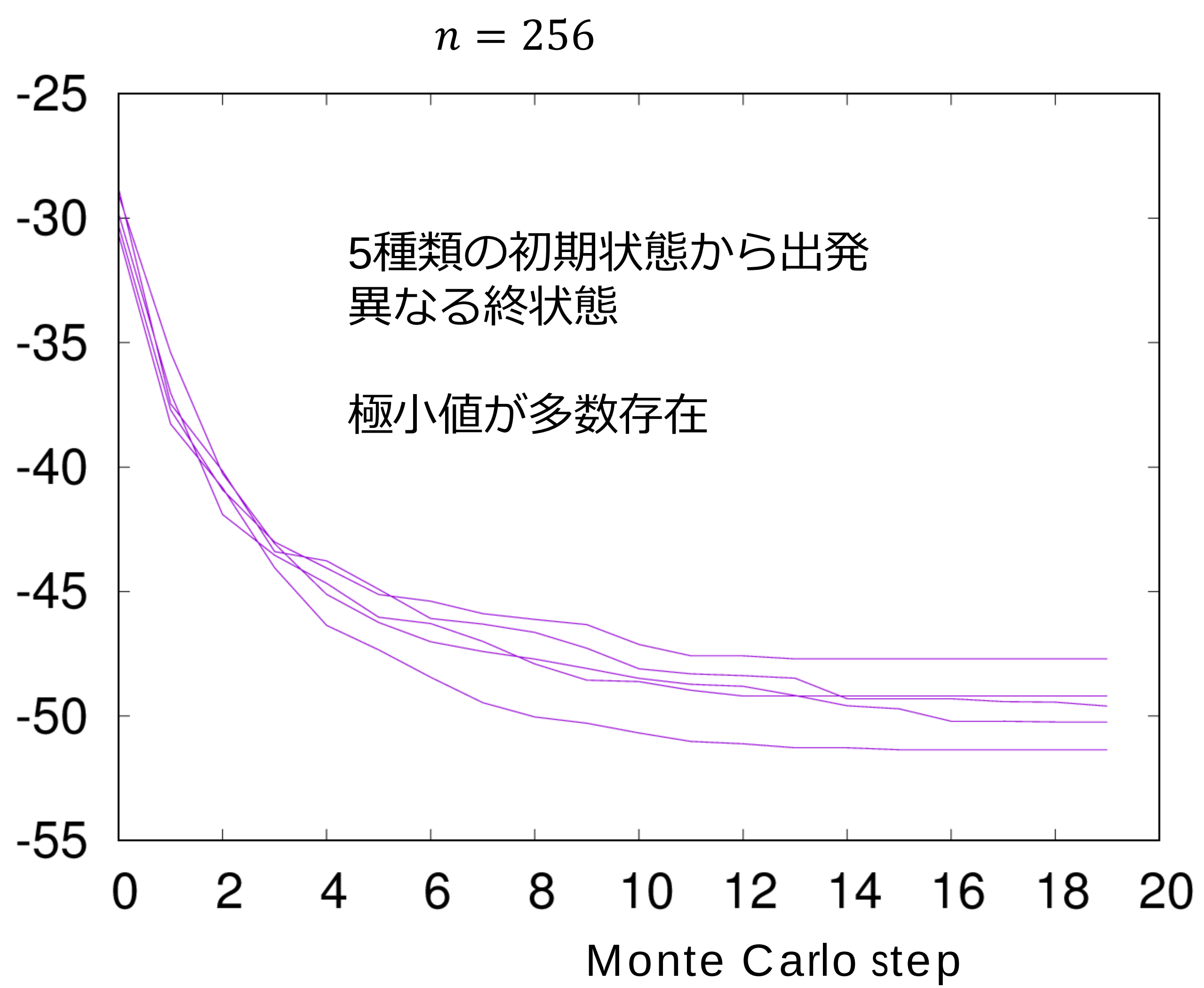

出発点 1
出発点 2

random spin系

- n個の±1を取る変数s_i
- 相互作用J_{ij} $(J_{ij} = J_{ji}, J_{ii} = 0)$
  - 正負の値がランダム
- $E = -\sum_{ij} J_{ij} s_i s_j$を最小にする
  - s_iを二つのグループに分ける

spin系のMonte Carlo法

- 毎回、ランダムにs_iを選び、変化
  - $s_i \to s_i + \Delta \mathrm{s_i}$
- $\Delta E = -2\sum_j J_{ij} s_j \Delta s_i < 0$ならば$s_i$を変更する（符号を反転する）
- 1 Monte Carlo step : n回の更新



$n = 256$
-25
-30
-35
-40
-45
-50
-55
0
2
4
6
8
10
12
14
16
18
20
Monte Carlo step
5種類の初期状態から出発
異なる終状態
極小値が多数存在