

# データサイエンス系科目群（理・博）：シミュレーション実習 第4回 講義資料

担当：川崎猛史

名古屋大学大学院理学研究科理学専攻物理科学系・非平衡物理研究室 (R 研)

Last update: May 2, 2022

# 目次

- 1 講義のスケジュール
  - シラバス
- 2 第3回自主課題解説
- 3 Langevin 方程式で駆動される粒子の数値計算（1粒子）
  - Langevin 方程式と揺動散逸定理
  - Langevin 方程式の離散化
  - 無次元化
  - 正規乱数の発生方法：Box Muller 法 [4]
  - Box Muller method の高速化（Marsaglia polar 法）
- 4 様々な平均量
  - 時間平均
  - アンサンブル平均
- 5 第4回自主課題
- 6 参考文献

# 目次

- 1 講義のスケジュール
  - シラバス
- 2 第3回自主課題解説
- 3 Langevin 方程式で駆動される粒子の数値計算（1粒子）
  - Langevin 方程式と揺動散逸定理
  - Langevin 方程式の離散化
  - 無次元化
  - 正規乱数の発生方法：Box Muller 法 [4]
  - Box Muller method の高速化（Marsaglia polar 法）
- 4 様々な平均量
  - 時間平均
  - アンサンブル平均
- 5 第4回自主課題
- 6 参考文献

# 1. 講義のスケジュール

- 当実習は春 1 期にて実施する.
- 講義資料は各講義**予定日当日朝 11 時迄**にアップロードする.
- スケジュール
  - 1 4/11 : 第 1 回
  - 2 4/18 : 第 2 回
  - 3 4/25 : 第 3 回
  - 4 **5/02 : 第 4 回 (中間レポート課題公開)**
  - 5 5/09 : 第 5 回
  - 6 5/16 : 第 6 回 (中間レポート課題提出期限予定)
  - 7 5/23 : **休講**
  - 8 5/30 : 第 7 回
  - 9 6/06 : 第 8 回 (期末レポート課題公開)
  - 10 6/13
  - 11 6/20 : (期末レポート課題提出期限予定)

# 1.1. シラバス

当実習では以下の内容を扱う予定である（進捗に合わせ変更する可能性がある）。

## 1 導入

- C(C++) の使い方 (主に数値計算)
- Python の使い方 (データ解析と作図)
- 数値計算の理念
- 桁落ち
- 科学計算における無次元化

## 2 常微分方程式の数値解法：減衰振動や調和振動子を例に

- 微分方程式の数値積分
- オイラー法
- 蛙飛び差分法（速度ベルレ法）
- 軌道の安定性と保存則

## 3 1粒子系のブラウン運動

- ランジュバン方程式（確率微分方程式）
- 正規乱数の生成法
- オイラー・丸山法
- 時間平均とアンサンブル平均
- 拡散係数の計算

## 4 多粒子系のブラウン運動

- 相互作用力の計算方法
- 非平衡系のシミュレーション：相分離現象を例に

## 5 多粒子系の分子動力学シミュレーション

- 位置ベルレ法と速度ベルレ法
- 多粒子系における保存則（運動量・エネルギー・角運動量）

## 6 モンテカルロ法

- 統計力学の復習
- マルコフ連鎖モンテカルロ法
- メトロポリス判定法

# 目次

- 1 講義のスケジュール
  - シラバス
- 2 第3回自主課題解説
- 3 Langevin 方程式で駆動される粒子の数値計算（1粒子）
  - Langevin 方程式と揺動散逸定理
  - Langevin 方程式の離散化
  - 無次元化
  - 正規乱数の発生方法：Box Muller 法 [4]
  - Box Muller method の高速化（Marsaglia polar 法）
- 4 様々な平均量
  - 時間平均
  - アンサンブル平均
- 5 第4回自主課題
- 6 参考文献

## 2. 第 3 回自主課題解説

### 第 3 回自主課題 減衰運動の数値計算と解析計算の比較（続き）

- (1) 第 2 回自主課題の主計算から作図まで全て python でコーディングしてみよ．python の文法については，検索すればすぐに情報を取得できるが，例えば以下のサイトなどを参照せよ [リンク](#) [1]．
- (2) (発展課題) 次回は Langevin 方程式によって駆動される粒子の運動を数値的に解く．その際に，正規乱数を発生させる必要がある．**余力があれば**正規乱数を発生させるアルゴリズム (Box Muller 法) を以下に示すのでこれを解読せよ [次回解説する]．
- (3) (発展課題) リスト 3 に示したサンプルプログラム "BM.h" は，分散 1 平均 0 の正規分布を満たす乱数を生成する．今，この乱数を  $10^5$  回生成したとき，確率変数の度数分布（ヒストグラム）を描け [次回解説する]．

#### 【解説】

- (1) 以下，第 2 回自主課題に関する python でのコーディングである．

## 2. 第3回自主課題解説 (2)

リスト 1: (1) のコーディング例. 第2回自主課題を python でコーディングしたもの. "damping.py". 以下の GitHub リポジトリより取得可能[リンク](#). C でコーディングしたものと比較してみよ.

```
1 import numpy as np
2 import sys
3
4 def damp(dt):
5     v=10.
6     i=0
7     out=1
8     f = open("damp_dt_{:.4f}.dat".format(dt), 'w')
9     while v > sys.float_info.epsilon: # numerical epsilon wherein
        the "float" corresponds to the double in C
10         v -= v*dt
11         i+=1.
12         if(i>=out):
13             print("{:.4f} {:.20f}".format(i*dt, v))
14             f.write("{:.4f} {:.20f}\n".format(i*dt, v))
15             out*=1.5
16     f.close()
17
18 for k in range (0,4):
19     print("dt={:.5f}".format(1.e-4*10**k))
20     damp(1.e-4*10**k)
```



## 2. 第3回自主課題解説 (3)

- (2) 後ほど詳しく解説する.
- (3) 正規乱数の発生およびそのヒストグラムのコーディング例.

リスト 2: (1) のコーディング例. 正規乱数の発生方法 ("BM.h"をインクリュード) およびヒストグラムの典型的な書き方. "hist\_gauss.cpp". 以下の GitHub リポジトリより取得可能[リンク](#)

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <math.h>
4 #include <iostream>
5 #include <fstream>
6 #include <cfloat>
7 #include "BM.h"
8
9 #define asize (int)1e5
10 #define hsize 1200
11 //using namespace std;
12
13 int histogram(double bin, double *array, double *hist){
14     int i;
15     char filename[128];
16     std::ofstream file;
17     sprintf(filename, "hist_gauss.dat");
18     file.open(filename);
```

## 2. 第3回自主課題解説 (4)

```
19     for(i=0;i<hsize;i++)
20         hist[i]=0;
21     for(i=0;i<asize;i++)
22         hist[(int)floor(array[i]/bin)+hsize/2]+=1.0/asize/bin;
23     for(i=0;i<hsize;i++){
24         file<<(i-hsize/2)*bin <<"\t"<< hist[i]<<std::endl;
25         std::cout<<(i-hsize/2)*bin <<"\t"<< hist[i]<<std::endl;
26     }
27     return 0;
28 }
29 int main(){
30     double array[asize],hist[hsize],bin=1.e-2;
31     int i;
32     for(i=0;i<asize;i++)
33         array[i]=gaussian_rand();
34     histogram(bin,array,hist);
35     return 0;
36 }
```

## 2. 第3回自主課題解説 (5)

リスト 3: (3) の作図コーディング例. "hist\_gauss.py". 以下の GitHub リポジトリより取得可能[\[リンク\]](#)

```

1 import matplotlib
2 import matplotlib.pyplot as plt
3 %matplotlib inline
4 %config InlineBackend.figure_format = 'retina'
5 import matplotlib.cm as cm # colormap
6 import numpy as np
7 import math
8 plt.rcParams["text.usetex"] = True
9 fig = plt.figure(figsize=(8,8))
10
11 plt.subplot(111)
12 ax1 = fig.add_subplot(111)
13 x,hist= np.loadtxt("./Documents/GitHub/2022-simulation-training-main
    /Lecture4/hist_gauss.dat", comments='#', unpack=True)
14 plt.plot(x,hist, "-o", markersize=3,color=cm.jet(3/4),label=r"$P(X)$
    ")
15
16 ###Drawing a gaussian curve #####
17 x= np.linspace(-6, 6, 100)
18 f= (2.*3.14)**(-0.5)*np.exp(-x**2./2.)
19 plt.plot(x,f, "--", markersize=3,linewidth = 2.0, color="k",label=r"$
    \frac{1}{\sqrt{2\pi}}e^{-X^2/2}$")
20 #####
  
```

## 2. 第3回自主課題解説 (6)

```
21 plt.tick_params(which='major',width = 1, length = 10)
22 plt.tick_params(which='minor',width = 1, length = 5)
23 ax1.spines['top'].set_linewidth(3)
24 ax1.spines['bottom'].set_linewidth(3)
25 ax1.spines['left'].set_linewidth(3)
26 ax1.spines['right'].set_linewidth(3)
27 plt.xlabel(r"$X$",color='k', size=25)
28 plt.ylabel(r"$P(X)$",color='k', size=25)
29 plt.xticks(color='k', size=25)
30 plt.yticks(color='k', size=25)
31
32 plt.legend(ncol=1, loc=1, borderaxespad=0, fontsize=25,frameon=False
33            )
34 #####
35 plt.subplots_adjust(wspace=0.5, hspace=0.25)
36 plt.savefig('./Documents/GitHub/2022-simulation-training-main/
37             Lecture4/hist_gauss.pdf')
38 plt.show()
```

## 2. 第3回自主課題解説 (7)

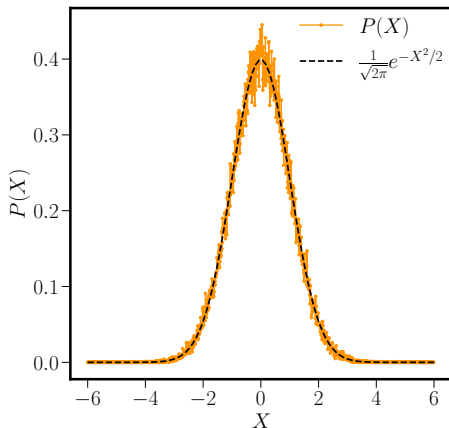


図 1: (3) の結果．正規乱数を  $10^5$  回発生させた際の振る舞い．規格化した度数分布（確率密度関数）．

# 目次

- 1 講義のスケジュール
  - シラバス
- 2 第 3 回自主課題解説
- 3 Langevin 方程式で駆動される粒子の数値計算 (1 粒子)
  - Langevin 方程式と揺動散逸定理
  - Langevin 方程式の離散化
  - 無次元化
  - 正規乱数の発生方法: Box Muller 法 [4]
  - Box Muller method の高速化 (Marsaglia polar 法)
- 4 様々な平均量
  - 時間平均
  - アンサンブル平均
- 5 第 4 回自主課題
- 6 参考文献

### 3.Langevin 方程式で駆動される粒子の数値計算 (1 粒子)

前回 Langevin 熱浴中を運動する粒子の運動方程式やその運動（ブラウン運動）について理論的に議論した [2]．今回はこのような粒子運動の数値解法を導入する．

## 3.1. Langevin 方程式と揺動散逸定理

- 前回の講義では、抵抗係数  $\zeta$  の温度  $T$  の  $d$  次元溶媒 (熱浴) 中を運動する質量  $m$  の球のブラウン運動は、以下の Langevin 方程式

$$m \frac{d\mathbf{v}(t)}{dt} = -\zeta \mathbf{v}(t) + \mathbf{F}_B(t) \quad (1)$$

によってよく表すことができることを扱った。

- ここで、熱揺動力  $\mathbf{F}_B(t)$  は、白色ノイズ (2 時刻時間相関関数が時間に関する delta 関数 = 同時刻以外相関をもたない) であり

$$\langle \mathbf{F}_B(t) \cdot \mathbf{F}_B(t') \rangle = 2d\zeta k_B T \delta(t - t') \quad (2)$$

と表されることを導いた (第 2 種揺動散逸定理) 。

- 次節では、この Langevin 方程式を離散化し、数値的に解く。



## 3.2. Langevin 方程式の離散化

- Langevin 方程式は、毎時刻ランダム力によって叩かれ、その都度エネルギーが散逸されることで熱平衡状態を保つ。
- そのため Langevin 方程式の離散化においては、高次の精度は必要なく Euler 法で十分である。
- このような確率微分方程式に対して適用した、Euler 法による離散化手法を Euler・丸山法 と呼ぶ [3]。
- 次に熱揺動力の相関関数において現れる Dirac のデルタ関数の扱いについて考える。
- 熱揺動力の時間相関関数 (式 (2)) を時間刻みを  $\Delta t$  として積分することで

$$2dk_{\text{B}}T\zeta = \int_{-\infty}^{\infty} dt' \langle \mathbf{F}(t' + t_0) \cdot \mathbf{F}(t_0) \rangle \sim \langle \mathbf{F}(t_0)^2 \rangle \Delta t \quad (3)$$

の関係が得られる。

- 従って、

$$\langle \mathbf{F}(t_0)^2 \rangle \sim \frac{2dk_{\text{B}}T\zeta}{\Delta t} \quad (4)$$

であるから、熱揺動力の各成分の分散値が  $\frac{2k_{\text{B}}T\zeta}{\Delta t}$  であることが分かる。

## 3.2. Langevin 方程式の離散化 (2)

- 然るに、熱揺動力は、

$$\mathbf{F}_B(t) \sim \sqrt{\frac{2k_B T \zeta}{\Delta t}} \mathbf{R}_G \quad (5)$$

と表される．ここで  $\mathbf{R}_G$  は、平均が  $\mathbf{0}$ 、各成分の標準偏差が 1 である正規（ガウス）乱数ベクトルである．

- 続いて Langevin 方程式 (式 1) を Euler・丸山法で離散化すると

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t) - \frac{\zeta}{m} \mathbf{v}(t) \Delta t + \sqrt{\frac{2k_B T \zeta \Delta t}{m^2}} \mathbf{R}_G \quad (6)$$

となる．以下、離散化した差分方程式を無次元化していく．

### 3.3. 無次元化

本節では、離散化した **Langevin** 方程式 (式 (6)) を無次元化する。

- 長さの単位を粒子の粒径  $a[\text{m}]$ , 時間の単位を  $t_0[\text{s}]$  とおき,  $\mathbf{v} = \frac{a}{t_0} \tilde{\mathbf{v}}, t = t_0 \tilde{t}$  とし, 物理次元をもつ量と無次元数を分け, 式 (6) に代入すると

$$\frac{a}{t_0} \tilde{\mathbf{v}}(\tilde{t} + \Delta \tilde{t}) = \frac{a}{t_0} (1 - \frac{\zeta}{m} t_0 \Delta \tilde{t}) \tilde{\mathbf{v}}(\tilde{t}) + \sqrt{\frac{2k_B T \zeta t_0 \Delta \tilde{t}}{m^2}} \mathbf{R}_G \quad (7)$$

となる。式 (7) の両辺を  $a/t_0$  で割れば, 無次元数に関する方程式

$$\tilde{\mathbf{v}}(\tilde{t} + \Delta \tilde{t}) = (1 - \frac{\zeta}{m} t_0 \Delta \tilde{t}) \tilde{\mathbf{v}}(\tilde{t}) + \sqrt{\frac{2k_B T \zeta t_0^3 \Delta \tilde{t}}{m^2 a^2}} \mathbf{R}_G \quad (8)$$

となる。ここで時間の単位  $t_0$  を, 第 2 回の減衰運動の際に扱った  $t_0 = \frac{m}{\zeta}$  に選ぶと, 式 (8) の右辺の未定係数が一つ落ちて

$$\tilde{\mathbf{v}}(\tilde{t} + \Delta \tilde{t}) = (1 - \Delta \tilde{t}) \tilde{\mathbf{v}}(\tilde{t}) + \sqrt{\frac{2mk_B T}{a^2 \zeta^2}} \Delta \tilde{t} \mathbf{R}_G \quad (9)$$

と表すことができる。

### 3.3. 無次元化 (2)

- ここで熱揺動力につく係数は無次元数であるため、 $T^* = \frac{mk_B T}{a^2 \zeta^2}$  とおき、これが当該計算における **パラメータ**<sup>1</sup> となる。
- 一般的に、動的方程式において出てくるパラメータは、**異なる物理現象を特徴づける時間スケールの比** で書けることが多い。
- 実際に、今回の時間の単位  $t_0 = \frac{m}{\zeta}$  (減衰運動の緩和時間と、拡散を特徴づける時間スケール  $t_B = \frac{a^2 \zeta}{k_B T} (= a^2/D)$  の比をとると

$$T^* = \frac{mk_B T}{a^2 \zeta^2} = \frac{t_0}{t_B} \quad (10)$$

となる。このことは  $T^*$  が大きくなるほど、減衰の緩和時間に対して拡散時間  $t_B$  が相対的に短くなることを表す。

- この程、無次元化された (速度に関する)Langevin 方程式は

$$\tilde{\mathbf{v}}(\tilde{t} + \Delta \tilde{t}) = (1 - \Delta \tilde{t}) \tilde{\mathbf{v}}(\tilde{t}) + \sqrt{2T^* \Delta \tilde{t}} \mathbf{R}_G \quad (11)$$

となる。ここで  $T^*$  には任意の無次元数を入れることができる。また  $T^*$  の値によって数値計算の結果は系統的に変化する。

### 3.3. 無次元化 (3)

- また、位置に関しては、

$$\tilde{\mathbf{r}}(\tilde{t} + \Delta\tilde{t}) = \tilde{\mathbf{r}}(\tilde{t}) + \tilde{\mathbf{v}}(\tilde{t} + \Delta\tilde{t})\Delta\tilde{t} \quad (12)$$

と離散化する．

- ここで式 (12) は陰解法，式 (11) は陽解法としている．このように，速度と位置で陽解法・陰解法をハイブリッドで使用する方法を **半陰 Euler 法 (Semi-implicit Euler method)** という．

### 3.3. 無次元化 (4)

#### 半陰 Euler 法 (Semi-implicit Euler method)

- 陽的 Euler 法と陰的 Euler 法を連立させる方法.
- 調和振動子系ではシンプレクティック ( $t \rightarrow t + \Delta t$  の時間前進軌道と  $t + \Delta t \rightarrow t$  の時間後退軌道が同一になる) になることが知られている.
- 調和振動子系での適用例 ( $v$  は陽解法,  $x$  は陰解法):

$$v(t + \Delta t) = v(t) - \omega^2 x(t) \Delta t \quad (13)$$

$$\begin{aligned} x(t + \Delta t) &= x(t) + v(t + \Delta t) \Delta t \\ &= x(t) + v(t) \Delta t - \omega^2 x(t) (\Delta t)^2 \end{aligned} \quad (14)$$

- これを位相空間に関する時間発展行列で書くと

$$\begin{pmatrix} v(t + \Delta t) \\ x(t + \Delta t) \end{pmatrix} = \begin{pmatrix} 1 & -\omega^2 \Delta t \\ \Delta t & 1 - \omega^2 (\Delta t)^2 \end{pmatrix} \begin{pmatrix} v(t) \\ x(t) \end{pmatrix} = \mathbf{A}(t + \Delta t | t) \begin{pmatrix} v(t) \\ x(t) \end{pmatrix} \quad (15)$$

- 一方, 時間後退方向に関して同様の式から得られる  $\mathbf{A}(t | t + \Delta t)$  は  $\mathbf{A}(t + \Delta t | t)$  の逆行列となる. このような性質をシンプレクティック (symplectic) といい, 軌道の収束性がよくなる.
- シンプレクティックな場合  $\det \mathbf{A}(t | t + \Delta t) = 1$  (位相空間が膨張・収縮しない).
- 陽解法のみを用いた場合はシンプレクティックにならず軌道が閉じなくなる.

- 今回の課題では, パラメータ  $T^*$  を変化させた際に粒子の平均二乗変位がどのように変化するかについて考察しよう.
- 次に, 当数値計算を実行する際に必要である正規乱数 (ガウス乱数) の発生方法を紹介する (第3回課題参照).

<sup>1</sup> 任意の数を当てることができ, その値により解の振る舞いが定性的に異なる

### 3.4. 正規乱数の発生方法 : Box Muller 法 [4]

本節では、正規乱数を数値的に発生させる方法について紹介する。

- 分散 1 の正規乱数  $R_G$  を計算で発生させる方法として Box Muller 法 [4] が有名である。

#### Box Muller 法 [4]

$U_1$  と  $U_2$  を  $[0,1]$  の範囲で分布する一様乱数とすると、以下の  $X_1$  と  $X_2$  は独立な標準 (分散 1 の) 正規乱数となる。

$$X_1 = \sqrt{-2 \log U_1} \cos 2\pi U_2 \quad (16)$$

$$X_2 = \sqrt{-2 \log U_1} \sin 2\pi U_2 \quad (17)$$

#### (証明)

- 式 (16) and (17) を連立することで  $U_1, U_2$  について解く。
- まず  $X_1$  と  $X_2$  の 2 乗を取ることで以下の関係式

$$\log U_1 = -\frac{X_1^2 + X_2^2}{2} \quad (18)$$

$$U_1 = e^{-\frac{X_1^2 + X_2^2}{2}} \quad (19)$$

を得る。

### 3.4. 正規乱数の発生方法: Box Muller 法 [4] (2)

- 式 (16) を式 (17) で割ることで

$$\begin{aligned}\frac{X_2}{X_1} &= \tan 2\pi U_2 \\ U_2 &= \frac{1}{2\pi} \arctan \frac{X_2}{X_1}\end{aligned}\quad (20)$$

を得る.

- これらを用いることで  $X_1$  と  $X_2$  が **独立なガウス過程** であることを示す.
- いま確率密度関数  $P(X_1, X_2)$  と  $\tilde{P}(U_1, U_2)$  を導入する.
- $P(X_1, X_2)$  と  $\tilde{P}(U_1, U_2)$  の関係は,

$$P(X_1, X_2) dX_1 dX_2 = \tilde{P}(U_1, U_2) dU_1 dU_2 = \tilde{P}(U_1, U_2) \left| \frac{\partial(U_1, U_2)}{\partial(X_1, X_2)} \right| dX_1 dX_2 \quad (21)$$

となる.

- $U_1$  と  $U_2$  は独立な  $[0, 1]$  を閾値にもつ一様乱数であることから

$$\int_0^1 dU_1 \int_0^1 dU_2 \tilde{P}(U_1, U_2) = 1 \quad (22)$$

となり  $\tilde{P}(U_1, U_2) = 1$  を得る.



### 3.4. 正規乱数の発生方法: Box Muller 法 [4] (3)

- いま, これらのヤコビアン  $\left| \frac{\partial(U_1, U_2)}{\partial(X_1, X_2)} \right|$  は

$$\left| \frac{\partial(U_1, U_2)}{\partial(X_1, X_2)} \right| = \left| \begin{array}{cc} \frac{\partial U_1}{\partial X_1} & \frac{\partial U_1}{\partial X_2} \\ \frac{\partial U_2}{\partial X_1} & \frac{\partial U_2}{\partial X_2} \end{array} \right| = \left| \frac{\partial U_1}{\partial X_1} \frac{\partial U_2}{\partial X_2} - \frac{\partial U_2}{\partial X_1} \frac{\partial U_1}{\partial X_2} \right| \quad (23)$$

- となるので, 適宜微分を実行すれば

$$\frac{\partial(U_1, U_2)}{\partial(X_1, X_2)} = \left| \frac{1}{2\pi} \frac{e^{-\frac{X_1^2 + X_2^2}{2}}}{1 + (\frac{X_2}{X_1})^2} - \frac{1}{2\pi} \frac{-X_2^2}{X_1^2} \frac{e^{-\frac{X_1^2 + X_2^2}{2}}}{1 + (\frac{X_2}{X_1})^2} \right| = \frac{1}{2\pi} e^{-\frac{X_1^2 + X_2^2}{2}} = \frac{1}{\sqrt{2\pi}} e^{-\frac{X_1^2}{2}} \frac{1}{\sqrt{2\pi}} e^{-\frac{X_2^2}{2}} \quad (24)$$

を得る. 従って,  $P(X_1, X_2) = p(X_1)p(X_2)$  であるので  $X_1$  と  $X_2$  は,

**独立なガウス過程**であることが示された.

$y = \arctan x$  の微分

$y = \arctan x$  は,  $x = \tan y$  である. これを  $y$  で微分すると  $\frac{dx}{dy} = \frac{1}{\cos^2 y}$  である. つまり

$$\frac{dy}{dx} = \cos^2 y = \frac{1}{1 + \tan^2 y} = \frac{1}{1 + x^2} \quad (25)$$

を得る.

## 3.5. Box Muller method の高速化 (Marsaglia polar 法)

### 改良 Box Muller 法 (Marsaglia polar 法) [5]

- 1  $[-1, 1]$  を閾値にもつ一様乱数  $u_1, u_2$  を用いることで, 2次元ベクトル

$$\mathbf{R} = (u_1, u_2) \quad (26)$$

を導入する. ここでは,  $\mathbf{R}$  を適当にふり, 半径 1 の円の中に入った乱数のペアのみ抜き出す (外に出たものは棄却) .

- 2 すると, ここで得た  $R^2 = u_1^2 + u_2^2$  は  $[0, 1]$  を閾値にもつ一様乱数となる (証明を下に載せた). そのため  $U_1 = R^2$  とおくことができ,  $\sqrt{-2 \log U_1}$  を得る.

- 3 次に  $\mathbf{R}$  の偏角は  $[0, 2\pi]$  の範囲を一様に分布することから  $2\pi U_2$  と等価である. このことから,  $2\pi U_2$  を引数にもつ三角関数は, 間接的に

$$\frac{u_1}{R} = \cos 2\pi U_2 \quad (27)$$

$$\frac{u_2}{R} = \sin 2\pi U_2 \quad (28)$$

のようにして得られ, 直接三角関数を計算する必要がなくなる. そのため計算スピードが上がる.

- Box Muller 法における, **三角関数の計算は比較的重い**計算であり, 大量に乱数を発生させる際は大きなコストとなる.
- ここでは, 計算方法を工夫することで, **三角関数を直接計算せずに Box Muller 法と同等の結果を得る方法**を紹介する (Marsaglia polar 法) [5].

## 3.5. Box Muller method の高速化 (Marsaglia polar 法) (2)

(2) の証明)

$X = R^2$  とすると確率密度関数  $f(X)$  を  $R$  に関する確率密度関数  $g(R)$  との関係は

$$f(X)dX = g(R)dR = g(R)\frac{\partial R}{\partial X}dX \quad (29)$$

となる. いま,  $g(R)$  は  $R$  に比例<sup>2</sup> するので  $g(R) = CR$  である. 従って以下の積分を実行することで

$$\int_0^1 g(R)dR = \left[ \frac{C}{2} R^2 \right]_0^1 = \frac{C}{2} = 1 \quad (30)$$

より  $C = 2$  となる. このことから,  $\frac{\partial R}{\partial X} = \frac{1}{2R}$  となり

$$f(X)dX = \frac{2R}{2R}dX = 1dX \quad (31)$$

を得る. いま,  $f(X) = 1$  より  $R^2$  は  $[0, 1]$  を閾値とする一様乱数であることが示された.

## 3.5. Box Muller method の高速化 (Marsaglia polar 法) (3)

リスト 4: 正規乱数の発生アルゴリズム (Box-Muller 法) 以下のプログラム"BM.h"は以下の GitHub リポジトリより取得可能[リンク](#). サブルーチンとして移植するかヘッダとして include して用いるとよい.

```
1 double unif_rand(double left, double right)
2 {
3     return left + (right - left)*rand()/RAND_MAX;
4 }
5 double gaussian_rand(void)
6 {
7     static double iset = 0;
8     static double gset;
9     double fac, rsq, v1, v2;
10
11     if (iset == 0) {
12         do {
13             v1 = unif_rand(-1, 1);
14             v2 = unif_rand(-1, 1);
15             rsq = v1*v1 + v2*v2;
16         } while (rsq >= 1.0 || rsq == 0.0);
17         fac = sqrt(-2.0*log(rsq)/rsq);
18
19         gset = v1*fac;
20         iset = 0.50;
21         return v2*fac;
22     } else {
```

## 3.5. Box Muller method の高速化 (Marsaglia polar 法) (4)

```
23     iset = 0;  
24     return gset;  
25 }  
26 }
```

---

<sup>2</sup>円環に分割した際、微小区間の面積は  $R$  に比例する

# 目次

- 1 講義のスケジュール
  - シラバス
- 2 第3回自主課題解説
- 3 Langevin 方程式で駆動される粒子の数値計算（1粒子）
  - Langevin 方程式と揺動散逸定理
  - Langevin 方程式の離散化
  - 無次元化
  - 正規乱数の発生方法：Box Muller 法 [4]
  - Box Muller method の高速化（Marsaglia polar 法）
- 4 様々な平均量
  - 時間平均
  - アンサンブル平均
- 5 第4回自主課題
- 6 参考文献

## 4. 様々な平均量

数値計算結果は、統計誤差を含むため、解析の際、各種平均操作を行い、物理的に重要な要素を抜き出す必要がある。よく用いられる 2 つの平均手法を紹介する。

## 4.1. 時間平均

### 時間平均

定常状態の物理量について、各時刻の値に対して平均化する操作を時間平均という。

- この操作は定常状態における物理量  $X(t)$  に対して行われ、実際に

$$\langle X \rangle_{t_0} = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T dt_0 X(t_0) \quad (32)$$

と計算される。

- 同様に、この操作を 2 時刻相関関数  $C(t, t_0) = X(t + t_0)X(t_0)$  に対して施す場合、

$$C(t) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T dt_0 C(t, t_0) \quad (33)$$

となる。この時、 $C(t) = \langle X(t + t_0)X(t_0) \rangle_{t_0}$ 、あるいは、より単純に  $C(t) = \langle X(t)X(0) \rangle$  と書く。



## 4.2. アンサンブル平均

### アンサンブル平均

独立かつ同等な試行（実験）をいくつも行いを加算平均することをアンサンブル平均という．

- 熱平衡状態にある  $\alpha$  番目のサンプルの物理量を  $A_i$  とするとき，アンサンブル平均は

$$\langle A \rangle_{\text{ens}} = \frac{1}{N_{\text{ens}}} \sum_{\alpha=1}^{N_{\text{ens}}} A_{\alpha} \quad (34)$$

となる．

- (参考) このことは以下の統計力学平均と同等である．

$$\langle A \rangle_{\text{ens}} = \frac{\text{Tr} A(\mathbf{q}, \mathbf{p}) e^{-\beta \hat{H}(\mathbf{q}, \mathbf{p})}}{Z} \quad (35)$$

ここで， $Z$  は分配関数:  $Z = \text{Tr} e^{-\beta \hat{H}(\mathbf{q}, \mathbf{p})}$  である．

# 目次

- 1 講義のスケジュール
  - シラバス
- 2 第 3 回自主課題解説
- 3 Langevin 方程式で駆動される粒子の数値計算（1 粒子）
  - Langevin 方程式と揺動散逸定理
  - Langevin 方程式の離散化
  - 無次元化
  - 正規乱数の発生方法：Box Muller 法 [4]
  - Box Muller method の高速化（Marsaglia polar 法）
- 4 様々な平均量
  - 時間平均
  - アンサンブル平均
- 5 第 4 回自主課題
- 6 参考文献

## 5. 第 4 回自主課題

## 第 4 回自主課題 1 粒子ブラウン運動の実装

温度  $T$ ，摩擦係数が  $\zeta$  である 3 次元溶媒中を熱揺動力による駆動される 1 粒子の運動を考える．この粒子の運動は Langevin 方程式  $m\dot{\mathbf{v}}(t) = -\zeta\mathbf{v}(t) + \mathbf{F}_B(t)$  でモデル化できることが広く知られている．この Langevin 方程式を，長さ，時間の単位をそれぞれ  $a$ ， $\frac{m}{\zeta}$  として無次元化すると， $T^* = \frac{mk_B T}{a^2 \zeta^2}$  がパラメータとなることを講義で扱った．そこで，この粒子の運動に関する以下の各問いに答えよ．なお，以下の  $\langle \cdots \rangle$  は，アンサンブル平均や時間平均を十分とった量であることを表す．

- (1) 平均二乗変位に関する解析解  $\langle \mathbf{r}(t)^2 \rangle = \frac{2dk_B T}{\zeta} \left\{ t - \frac{m}{\zeta} e^{-\zeta t/m} - \frac{m}{\zeta} \right\}$  を無次元化しパラメータ  $T^*$  を用いて表せ．
- (2) 任意の  $T^*$  に対して，(1) の理論解と数値解が一致することを確認せよ．数値解は半陰 Euler・丸山法で求めよ．
- (3) 粒子の速度相関関数  $C(t) = \langle \mathbf{v}(t) \cdot \mathbf{v}(0) \rangle$  を計算せよ．
- (4) (3) で求めた  $C(t)$  を時間積分することにより以下のグリーン・久保公式  $D = \frac{1}{d} \int_0^\infty dt C(t)$  が成立することを数値計算で確認せよ．

補足 (4) の数値積分は矩形や台形の短冊の和の計算をすればよい．

# 目次

- 1 講義のスケジュール
  - シラバス
- 2 第 3 回自主課題解説
- 3 Langevin 方程式で駆動される粒子の数値計算（1 粒子）
  - Langevin 方程式と揺動散逸定理
  - Langevin 方程式の離散化
  - 無次元化
  - 正規乱数の発生方法：Box Muller 法 [4]
  - Box Muller method の高速化（Marsaglia polar 法）
- 4 様々な平均量
  - 時間平均
  - アンサンブル平均
- 5 第 4 回自主課題
- 6 参考文献

## 参考文献・ウェブサイト

- [1] AIAcademy (2022) 【Python 入門】無料で Python の基本文法を 5 時間で学ぼう！  
(<https://aiacademy.jp/media/?p=265>).
- [2] Zwanzig R (2001) Nonequilibrium Statistical Mechanics.  
(Oxford University Press, Oxford ; New York).
- [3] Kloeden PE, Platen E (1992) Numerical Solution of Stochastic Differential Equations — SpringerLink.  
(Springer Berlin Heidelberg).
- [4] Box GEP, Muller ME (1958) A Note on the Generation of Random Normal Deviates.  
The Annals of Mathematical Statistics 29(2):610–611.
- [5] Marsaglia G, Bray TA (1964) A Convenient Method for Generating Normal Variables.  
SIAM Review 6(3):260–264.