シミュレーション演習中間レポート

杉浦航 (262201296)

第 1 問 (Buffon の針)

(1)
とりうる領域は

$$\int_0^{\frac{\pi}{2}} d\theta \int_0^{\frac{d}{2}} dy = \frac{\pi d}{4} \tag{1}$$

一方棒が x 軸に交わる条件の領域は、y_s において

$$\frac{2y}{l} \leq \cos\theta$$

であるから、この領域は

$$\int_0^{\frac{\pi}{2}} d\theta \int_0^{\frac{l}{2}\cos\theta} dy = \frac{l}{2} \tag{2}$$

よって eq. 1, eq. 2の比をとって

$$\frac{\frac{l}{2}}{\frac{\pi d}{4}} = \frac{2l}{\pi d}$$

題意は示された。

(2)
長さの次元を a という変数に持たせて、$y = ay_i$, $l = al_i$, $d = ad_i$ として無次元化すると、

$$\frac{2y_i}{l_i} \leq \cos\theta$$

$$p = \frac{2al_i}{\pi a d_i} = \frac{2l_i}{\pi d_i}$$

$l_i = 1,\ d_i = 2$ と置けば y_i が $[0, \frac{1}{2}]$ の

$$y_i \leq \frac{\cos\theta}{2} \tag{3}$$

$$p = \frac{1}{\pi} \tag{4}$$

という無次元化された不等式と確率を示す式に帰着させることができる。eq. 3は y_i を $[0, 1]$ とすることで $y_i \leq \cos\theta$ と変形できる。

この不等式と確率の式について、N 回の試行を行い、N_{accept} だけ成立したときに実際の成立確率 $p = \frac{N_{accept}}{N}$ と eq. 4を比較すれば良いことがわかるので、π は

$$\pi = \frac{N}{N_{accept}}$$

より求めることができる。実際に計算に使用したプログラムを pi.cpp、MT.h、プロットに使用したプログラムを plot_task1.py として添付する。なお乱数生成には講義で提供された MT.h によって提供されるメルセンヌ・ツイスター法を使用した。

円周率が n に対してどのように変化したかは Figure 1 の上側に示す。

(3)
Figure 1 の下側に示す。なお計算結果と理論値との誤差は、与えられた式に $C = 1$ を代入して直線でプロットした。$\frac{1}{\sqrt{n}}$ で誤差が減少することがわかる。10^6 で誤差が収束していく様子がわかる。

第 2 問 (溶媒中におけるばねで繋がれた物体の運動)

(1)
与式

$$m\ddot{x(t)} = -\zeta \dot{x(t)} - kx(t)$$

の特性方程式は

$$m\lambda^2 + \zeta\lambda + k = 0$$

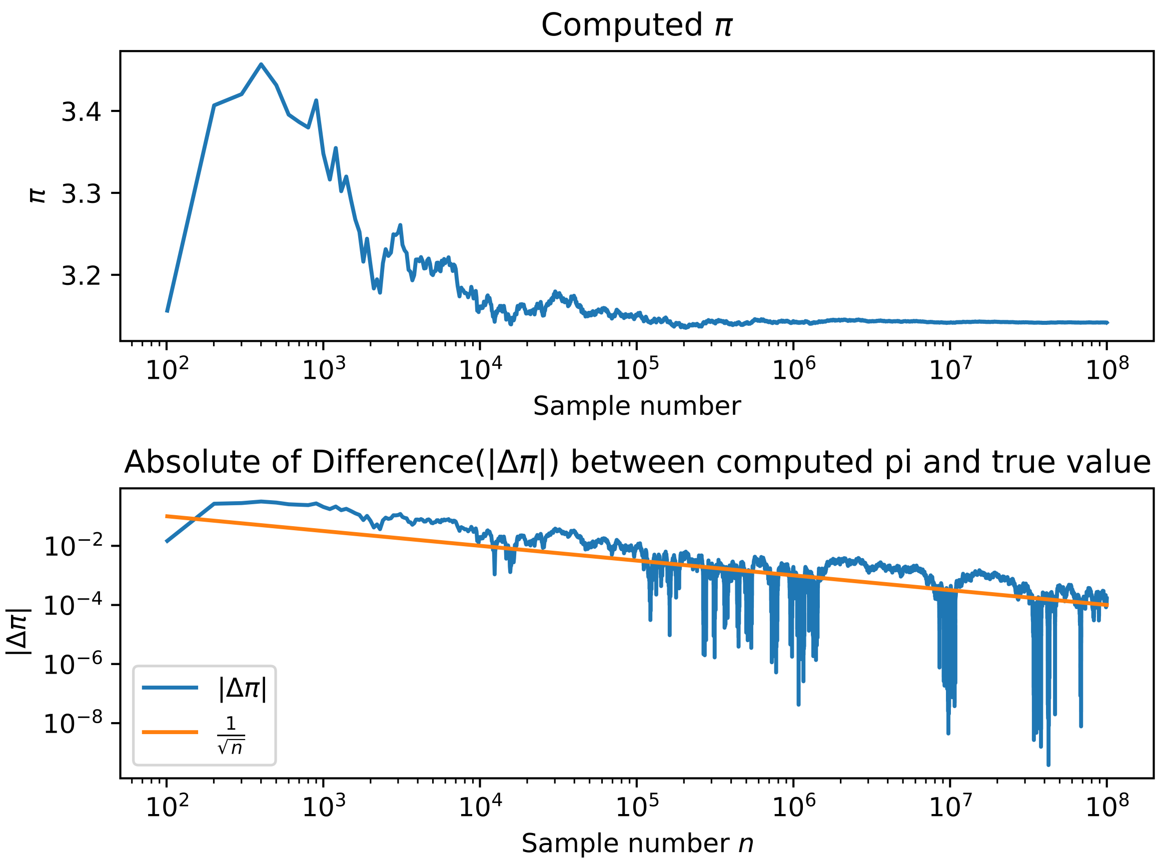

Figure1: サンプル数と計算された π の関係

で、λ について求めると

$$\lambda = \frac{-\zeta \pm \sqrt{\zeta^2 - 4mk}}{2m}$$

となる。よって

- 減衰振動となるのは $\zeta^2 - 4mk < 0$ のとき
- 過減衰となるのは $\zeta^2 - 4mk > 0$ のとき
- 臨界減衰となるのは $\zeta^2 - 4mk = 0$ のとき

であることがわかる。

(2)

与式を無次元化する。$x = a\tilde{x}$, $t = t_0\tilde{t}$, $\dot{x} = v = \frac{a}{t_0}\tilde{v}$ より

$$m\frac{dv}{dt} = \frac{ma}{t_0^2}\frac{d\tilde{v}}{d\tilde{t}}$$

$$-\zeta\dot{x}(t) = -\zeta v = -\zeta\frac{a}{t_0}\tilde{v}$$

$$-kx(t) = -ka\tilde{x}$$

なお $\frac{d\tilde{t}}{dt} = \frac{1}{t_0}$ を使った。それぞれ与式に代入して

$$\frac{ma}{t_0^2}\frac{d\tilde{v}}{d\tilde{t}} = -\zeta\frac{a}{t_0}\tilde{v} - ka\tilde{x}$$

$$\frac{d\tilde{v}}{d\tilde{t}} = -\frac{t_0^2}{ma}\left(\zeta\frac{a}{t_0}\tilde{v} + ka\tilde{x}\right)$$

$$\frac{d\tilde{v}}{d\tilde{t}} = -\left(\zeta\frac{t_0}{m}\tilde{v} + \frac{kt_0^2}{m}\tilde{x}\right)$$

を得る。

半陰的 Euler 法で離散化すると以下の 2 つの漸化式が得られる。

$$\tilde{v}_{n+1} = \tilde{v}_n - \left(\zeta\frac{t_0}{m}\tilde{v}_n + \frac{kt_0^2}{m}\tilde{x}_n\right)\Delta t \tag{5}$$

$$\tilde{x}_{n+1} = \tilde{x}_n + \tilde{v}_{n+1}\Delta t \tag{6}$$

(3)

$t_d = \frac{m}{\zeta}$, $t_s = \sqrt{\frac{m}{k}}$ より (2) で得た漸化式 eq. 5と eq. 6は

$$\tilde{v}_{n+1} = \tilde{v}_n - \left(\frac{t_0}{t_d}\tilde{v}_n + \frac{t_0^2}{t_s^2}\tilde{x}_n\right)\Delta t$$

$$\tilde{x}_{n+1} = \tilde{x}_n + \tilde{v}_{n+1}\Delta t$$

と変形される。

(4)

問題文では $t_0 = t_s$ とあり、あたかも t_0 に t_s を代入するような雰囲気だが、その後の文章で $\frac{t_0}{t_d}$ を使っているので、t_s に t_0 を代入し、パラメータ $T = \frac{t_0}{t_d}$ を置くと

$$\tilde{v}_{n+1} = (1 - T\Delta t)\tilde{v}_n - \tilde{x}_n \Delta t$$

$$\tilde{x}_{n+1} = \tilde{x}_n + \tilde{v}_{n+1}\Delta t$$

が得られる。

ところで、(1) で求めた条件を今の問題で扱いやすくする。左辺を変形すると

$$1 - \frac{4mk}{\zeta^2} = 1 - \frac{4}{T^2}$$

となるため、求めた条件は

- 減衰振動となるのは $T < 2$ のとき
- 過減衰となるのは $T > 2$ のとき
- 臨界減衰となるのは $T = 2$ のとき

と置き換えられる。

シミュレーションでは Table 1 に示すパラメータをそれぞれ使用した。

Table1: 使用したパラメータ

variable	value
a	1
t_0	1
Δt	0.001

実際に計算に使用したプログラムを md.cpp、プロットに使用したプログラムを plot_task2.py として添付する。

30000 ステップまで計算してプロットしたものを Figure 2に示す。

臨海減衰となる $T = 2$ の時点について、出力ファイルを確認したところ 0 を下回らず振動していないことが確認できた。

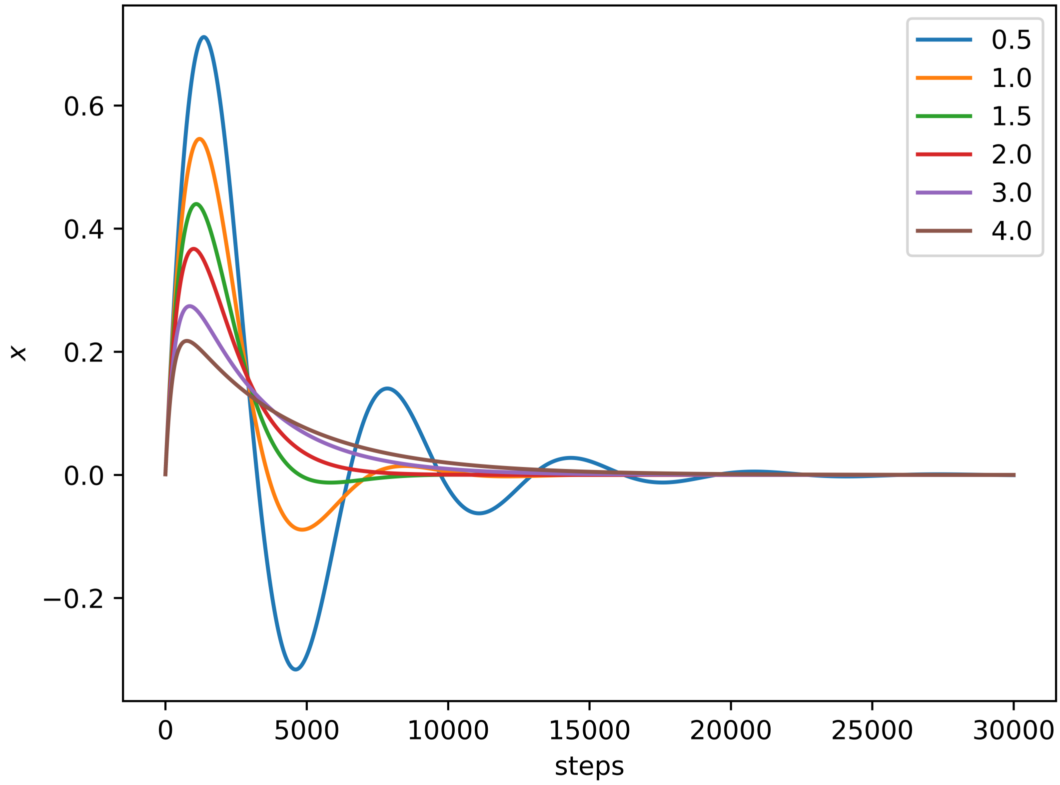

Figure2: 運動の様子

付録

課題 1 のプログラムの使用方法

提出ファイルはすべて同じディレクトリに保存されることを前提としている。

```
g++ pi.cpp
./a.out > pi.txt
python plot_task1.py pi.txt
```

標準出力でデータを書き出しているため、適当なファイル `pi.txt` に書き出した後、`plot_task1.py` でそのファイルを第一引数で指定している。

課題 2 のプログラムの使用方法

提出ファイルはすべて同じディレクトリに保存されることを前提としている。

```
g++ md.cpp -fopenmp -std=c++11 -O3
./a.out
python plot_task2.py
```

C++11 の機能と OpenMP を使っているので指定している。python のプログラムに C++ の書き出すファイルがハードコーディングされているため、指定は必要ない。

実行環境

本課題はすべて名大が提供するスーパーコンピュータ不老のログインノードで行った。コンパイラには GCC 4.8.5 を利用した。