

## 4. 微分方程式

2016/10/31

### 1 作図環境の整備

数値実験やシミュレーションの結果は、レポートや論文に図として張り込むことができなければなりません。コンピュータを自由に使うことが出来なかった時期には、データを紙に印刷し、それを手でプロットしたものを清書していました。現在では様々なデータプロットツールが利用できます。実験では、多くの OS 上で 30 年近くも利用され続けている gnuplot (<http://www.gnuplot.info/>) を利用することにします。

#### 1.1 gnuplot のインストール

- 配布したインストールファイルをダブルクリックし、インストールを実行する。
- 途中の「追加タスクの選択」において、「実行ファイルのディレクトリを PATH 環境変数に追加する」にチェックをするのを忘れないようにしましょう。

**課題 1** 続いて、簡単なテストをしましょう。インストールした gnuplot を起動します。スタートメニューやアプリケーションの一覧から起動できない場合には

```
C:\Program Files\gnuplot\bin\wgnuplot.exe
```

を起動します。起動後、

```
plot sin(x)
```

と入力すると、正弦関数  $\sin$  の波形が描画されるはずですが。

## 2 共通のライブラリの準備

この実験では、Java を使ってモデルを実装します。その際に、ファイルへの出力や新しいリストの生成など、同じような処理を何度も行います。

**課題 2** そのような共通の処理をライブラリにまとめたものをインストールしておきましょう。

- Netbeans を用いて、新しいプロジェクト MyLib を作成します。デフォルトでは、`$HOME\Documents\NetBeansProjects` の下に作ります。
- <http://aoba.cc.saga-u.ac.jp/lecture/ModelingAndSimulation/javasrc/MyLib/src.zip> をダウンロードして、ダブルクリックして解凍します。これらのファイルを `$HOME\Documents\NetBeansProjects\MyLib\src` の下に置きます。
- プロジェクトをビルドします。  
`$HOME\Documents\NetBeansProjects\MyLib\dist\MyLib.jar` ができていることを確認します。

MyLib には以下のようなクラス、メソッドが含まれています。

`myLib.rungeKutta`

`DifferentialEquation`: 微分方程式を表すインターフェイス。

メソッド `double[] derivatives(double x, double y[])` を実装しなければならない。

`RungeKutta`: 4 次の Runge Kutta 法で微分方程式を解くクラス。二つのメソッド

```
static double[] rk4(double x, double y[], double h,
DifferentialEquation eq)
```

と

```
double[][] rk4dumb(double vstart[], double x1, double x2,
int nstep, DifferentialEquation eq)
```

を有する。

`Dynamics`: 具体的な微分方程式で表される系に対応する抽象クラス。

myLib.utils.FileIO:ファイル入出力用のライブラリ

BufferedWriter openWriter(String filename)

throws IOException:ファイル名を指定して writer を開く

void writeSSV(BufferedWriter out, Object... objects)

throws IOException:out へ向けて、objects で指定された対象物の列をスペース区切りで出力する

myLib.utils.Util:その他の便利なもの

List<T> createList():クラス T の空のリスト生成

int[] createRandomNumberList(int max)0 から max-1 までの整数の  
たらめな並びの生成

### 3 Euler 法による重力中の粒子運動のシミュレーション

Euler 法の例として、重力中の粒子運動のシミュレーションを行います。その準備として

- Netbeans を用いて、新しいプロジェクト DifferentialEquation を作成します。
- <http://aoba.cc.saga-u.ac.jp/lecture/ModelingAndSimulation/javasrc/DifferentialEquation/EulerMethod.zip> をダウンロードします。
- ダウンロードを解凍し、プロジェクトのソースファイルの下にフォルダごと置きます。
- Euler 法で時間  $h$  だけシステムを進めるクラス Euler.java を確認します。
- Euler 法で調和振動子のシミュレーションを行うクラス ThrowBallEuler.java の微分方程式部分を記述します。
- プロジェクトウィンドウ内の「ライブラリ」を右クリックし、「jar/フォルダの追加」から MyLib.jar を追加します。
- ビルドできることを確認しましょう。

**課題 3** クラス Euler.java に、一般的な Euler 法が書かれていることを確認しなさい。

**課題 4** クラス ThrowBallEuler.java に、調和振動子の微分方程式

$$\frac{d^2x}{dt^2} = 0$$

$$\frac{d^2y}{dt^2} = -g$$

に対応した記述を追加しなさい。

続いて、以下の手順で実行します。

- プロジェクトウィンドウ内のクラス `ThrowBallEuler` を右クリックし、メニュー内の「クラスを実行」によって実行します。
- `$HOME\Documents\NetBeansProjects\DifferentialEquation` のしたに `ThrowBallEuler-output.txt` ができていることを確認します。

クラス `ThrowBallEuler` の `main` メソッドを見てみましょう。

```
String className = ThrowBallEuler.class.getSimpleName();
```

によって、クラス `ThrowBallEuler` の名前を文字列 `className` に保存しています。出力用のファイルは

```
String filename = className+"-output.txt";
```

で、クラス名の後ろに `-output.txt` を追加して、命名し、

```
BufferedWriter out = FileIO.openWriter(filename);
```

でファイルへの出力を指定しています。

#### ソースコード 3.1 `throwBallEuler.plt`

```
1 set terminal png 14
2 set xlabel "x"
3 set ylabel "y"
4 set title "ThrowBallEuler"
5 set output "ThrowBallEuler.png"
6 plot "ThrowBallEuler-output.txt" ps 2
```

最後に、結果を出力しましょう。gnuplot は、作図作業命令をファイルに保存することができます。`$HOME\Documents\NetBeansProjects\DifferentialEquation` に `throwBallEuler.plt` をソースファイル 3.1 の内容で作成します。

NetBeans はテキストエディタとしても利用できます。Program 3.1 を NetBeans を使って作成しましょう。プロジェクトが表示されている左のスペースにある「ファイル」というタブを開けます。対象となるフォルダで、マウス右ボタンを押し、「新規」を選び

ます。ファイルの種類を選ぶところから「その他」→「空のファイル」を選び、ファイルを作成します。拡張子の指定を忘れないようにしてください。あとは、マウス右ボタンで「開く」を選択することで、編集を開始することができます。

#### 課題 5 シミュレーションの結果を作図しなさい。

このファイルをダブルクリックすると、ファイル `ThrowBallEuler-output.txt` からデータを読みだしてプロットするします。結果を確認しましょう。

出力ファイルが作成されない場合は、`throwBallEuler.plt` に誤りがあることが考えられます。そのような場合には、エクスプローラで、`throwBallEuler.plt` があるディレクトリに移動し、PATH 表示部分でマウスをクリックして PATH の文字列を反転させます (図 1)。その状態で `cmd` とタイプしリターンすることで、コマンドプロンプトを開きます。そこで、

```
gnuplot throwBallEuler.plt
```

とタイプすることで、エラーを含め出力することができます。

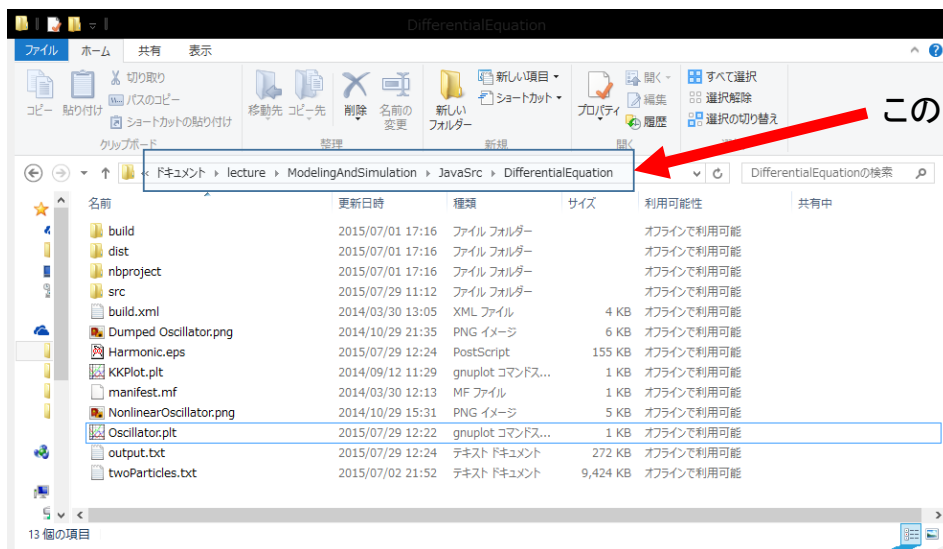


図1 エクスプローラからコマンドプロンプトを開く。