

計算科学特論、演習資料 1

今日のテーマ:

講義で学習した微分方程式の数値解法について、実際の数値計算プログラムで性質等を体験する。

Scilab (サイラボ) を使います。

Scilab, A Free Scientific Software Package

Scilab は工学・理学での応用に役立つ強力でオープンな計算環境を提供する数値計算のためのソフトウェアパッケージです。Scilab には何百もの数学関数が含まれ、加えて、様々なプログラミング言語 (C, C++, Fortran...) で作成されたプログラムを必要に応じて取り込む機能を備えています。洗練されたデータ構造(リスト、多項式、有理関数、線形システム...)を扱うことができる仕組みと、インタプリタ、高度なプログラミング言語を備えています。

Scilab ホームページ (<http://www.scilab.org>)から抜粋・直訳

よく似たソフトウェアに MATLAB や octave などがあります。scilab や octave には、商用で非常に広く使われている MATLAB に対するフリー・オープンな代替品としての側面があります。ただし、スクリプト言語の仕様は完全に互換ではありません。また、scilab は積極的に独自の機能・仕様を採用し、MATLAB からは徐々に離れつつあると言われていています。その一方で octave はユーザインタフェースを含めてそっくりになるモードを備える等、MATLAB 互換路線を継続するようです。興味があれば octave も試してみてください。

今日の予定

1. scilab 練習
2. scilab プログラム例解説
3. サンプルプログラム解析
4. レポート課題について説明

今日の内容は差分法のみです。有限要素法は別の機会に実施します。

1 scilab 練習

起動と終了

```
% scilab[Enter]  
quit[Enter]
```

基本操作

コンソールに直接コマンドを入力すれば、複素数や行列を扱うことのできる計算機として動作します。
入力履歴の表示や編集も可能です。

計算

```
-->1+2 [Enter]  
ans =
```

「1+2」と入力しEnter キーを押すことを示しています。
基本的に、この資料では入力する部分を赤字で示します。
実際の画面では赤くなりませんので注意してください。

```
3.
```

入力した数式の評価が帰ります。

```
-->3*3 [Enter]  
ans =
```

```
9.
```

```
-->2^0.5 [Enter]  
ans =
```

べき乗は ^ を使い、整数だけでなく実数べきが使えます。
0.5 乗は、平方根に一致します。

```
1.4142136
```

変数

```
-->X=4.2 [Enter]  
X =
```

=により変数への代入が可能です。
変数名はアルファベットで始まるアルファベットまたは、
数字 / アンダースコアからなる文字列が利用できます。

```
4.2
```

定数

```
-->%pi [Enter]  
%pi =
```

よく使われる定数が % で始まる名前であらかじめ変数
定義されています。π は %pi になります。

```
3.1415927
```

```
-->%e [Enter]  
%e =
```

オイラー常数（自然対数の底）は %e

```
2.7182818
```

```
-->1+%i*2 [Enter]
```

```
ans =
```

```
1. + 2.i
```

虚数単位は%iであり、組合せて複素数を表現することができます。

```
-->Z=2+%i*3 [Enter]
```

```
Z =
```

```
2. + 3.i
```

複素数を変数に代入することもできます。

```
-->Z*Z [Enter]
```

```
ans =
```

```
- 5. + 12.i
```

任意の操作、演算が複素数に関して利用できます。

```
-->%i^0.5 [Enter]
```

```
ans =
```

```
0.7071068 + 0.7071068i
```

```
-->[1 2; 3 4] [Enter]
```

```
ans =
```

```
1.    2.
```

```
3.    4.
```

行列は [] で表現します。列の区切りは空白または「,」、カンマを、行の区切りは「;」セミコロンを使います。

```
-->eye(2,2) [Enter]
```

```
ans =
```

```
1.    0.
```

```
0.    1.
```

```
-->zeros(2,2) [Enter]
```

```
ans =
```

```
0.    0.
```

```
0.    0.
```

```
-->ones(2,2) [Enter]
```

```
ans =
```

```
1.    1.  
1.    1.
```

```
-->A=[1 2; 3 4] [Enter]
```

```
A =
```

```
1.    2.  
3.    4.
```

```
-->A' [Enter]
```

```
ans =
```

```
1.    3.  
2.    4.
```

```
%pi[Enter]
```

```
%e[Enter]
```

複素数

```
1+%i*2[Enter]
```

```
Z=2+%i*3[Enter]
```

```
Z*Z[Enter]
```

```
%i ^ 0.5[Enter]
```

行列

```
[ 1 2 ; 3 4 ][Enter]
```

```
eye(2,2)[Enter]
```

```
zeros(2,2)[Enter]
```

```
%ones(2,2)[Enter]
```

```
A=[1 2 ; 3 4][Enter]
```

```
A' [Enter]
```

```
A'*A[Enter]
```

```
%eye/A
```

※四則演算、割り算を使った連立方程式の解法・逆行列の計算等ができる。

行列の要素どうしの演算

```
A=[1 2; 3 4][Enter]
```

```
B=A' [Enter]
```

A.*B[Enter]

A./B[Enter]

A+%i*B[Enter]

(A+%i*B)' [Enter]

※基本的に全ての数値は複素数を要素とする行列

便利な機能

継続行

... を行末に置く

A=[1 2 3; ...

4 5 6][Enter]

等間隔ベクトル

初期値:増加量:終値

A=1:0.5:3[Enter]

zeros(), ones(), eye()

引数が複数の整数: $n_1 \times n_2 \times \dots$ 行列、

引数が行列: 引数と同じ大きさの行列、

A=zeros(2,4)[Enter]

B=ones([1:4])[Enter]

C=eye(A' B')[Enter]

デリミタ「;」

行末にセミコロンを置くと、結果出力が抑制されます。

プログラム中では計算結果を表示する必要がある場合以外は「;」を忘れずに。

A=zeros(2,4); [Enter]

※セミコロンを着けなかった場合と比較してください。

関数定義

```
function [x,y]=myfct(a,b)
```

```
x=a+b
```

```
y=a-b
```

```
endfunction
```

```
myfct(1,2)[Enter]
```

```
myfct([1 2; 3 4],[1 2; 3 4]')[Enter]
```

※別ファイルで定義し、getf()で読み込むこともできる。

```
getf('functiondefinition.sci')[Enter]
```

グラフィックス

```
t=(0:0.05:1)' [Enter]
```

```
ct=cos(2*pi*t)[Enter]
```

```
plot2d(t,ct)[Enter]
```

```
square(-0.5,-1.5,1.5,1.5)[Enter]
```

```
tht=2*%pi*t[Enter]  
plot2d(tht,ct)[Enter]  
square(-%pi/2,-1.5,2.5*%pi,1.5)[Enter]
```

スクリプト

```
// コメント  
cd('~okano')[Enter]  
exec('script.sce')[Enter]
```

2 scilab プログラム例解説

script.sce

```
f=scf();
x=-10:10; y=-10:10; m=rand(21,21);
grayplot(x,y,m)
t=%pi:0.1:%pi; m=sin(t)*cos(t);
xclick();
clf();
grayplot(t,t,m);
f.color_map=graycolormap(16);
xclick();
f.color_map=jetcolormap(32);
xclick();
f.color_map=hotcolormap(64);
```

3 サンプルプログラム解析

授業情報ウェブサイトからサンプルプログラムとして用意したファイルをダウンロードし、zip ファイル中に含まれる Laplace 方程式、熱伝導方程式の数値計算プログラム

laplace1.sce laplace2.sce heateq1.sce heateq2.sce
について、課題を実施してください。

1. 4つのサンプルプログラムそれぞれについて、解いている問題・使われている数値計算法を調べてください。
2. 境界条件を変更してください。
3. 格子点数もしくは問題領域の分割数を変更してください。
4. 計算結果を表示するグラフィクスの color_map を適切に変更してください。
5. 以下の条件で熱伝導方程式の時間発展問題を解くプログラムを作成してください。
 1. 問題領域を原点と点(1,1)を頂点とする正方形領域とする
 2. 時間 t, 座標 x,y の温度分布 $u(t,x,y)$ が以下の境界条件・初期条件を満たすものとする
 - $u(t,0,y) = 0, u(t,1,y)=1, u(t,x,0)=u(t,x,1)=x, u(0,x,y)=0$
6. 3で作成したプログラムを修正・実行して以下のことを調べてください。
 1. 厳密解 $u(x,y)$ として x, y, xy, x+y 等を用いた場合の計算結果の正しさ。
 2. 格子点分布もしくは領域の分割数を変更した場合の計算結果の正しさとの関係。