神戸市立工業高等専門学校
電気工学科／電子工学科
専門科目「数値解析」

2017.12.1

演習6

山浦 剛（tyamaura@riken.jp）

講義資料ページ

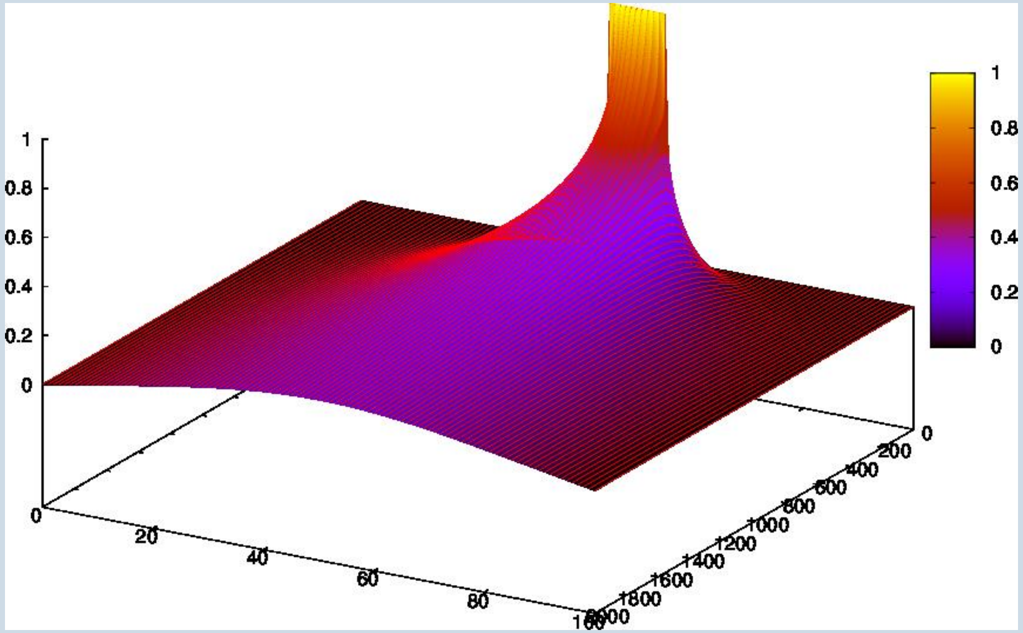
- http://climate.aics.riken.jp/members/yamaura/numerical_analysis.html

復習：拡散方程式

- 1次元拡散方程式の一般形は、次のようになる。
  - $\frac{\partial u}{\partial t} = \kappa \frac{\partial^2 u}{\partial x^2}$
- 従属変数uを格子点の位置での変数U_j^nで代表させると、差分化された拡散方程式は、
  - $\frac{1}{\Delta t}\left\{U_j^{n+1} - U_j^n\right\} = \frac{\kappa}{\Delta x^2}\left\{U_{j+1}^n - 2U_j^n + U_{j-1}^n\right\} \quad (j = 1,2,\ldots,N-1, n = 0,1,\ldots)$
- この式を漸化式として変形させると、
  - $U_j^{n+1} = \alpha U_{j+1}^n + (1-2\alpha)U_j^n + \alpha U_{j-1}^n$　ただし、$\alpha = \kappa \frac{\Delta t}{\Delta x^2}$
- このときの初期条件と境界条件は、
  - $U_j^0 = \phi(x_j) \quad (j = 0,1,\ldots,N)$
  - $U_0^n = U_0^N = 0 \quad (n = 0,1,\ldots)$

例題1：拡散方程式

- パラメータは以下の通り。
  - $\Delta t = 10^{-3}, \Delta x = 10^{-2}, \kappa = 0.02, N = 100$
  - 初期時刻から2秒後までを計算する
- 初期条件は、
  - $u(x,0) = \begin{cases} 1 & (0.45 \le x \le 0.55) \\ 0 & (0 \le x < 0.45, 0.55 < x \le 1) \end{cases}$
- 境界条件は、
  - $U_0^n = U_N^n = 0 \qquad (n = 0,1,\dots)$



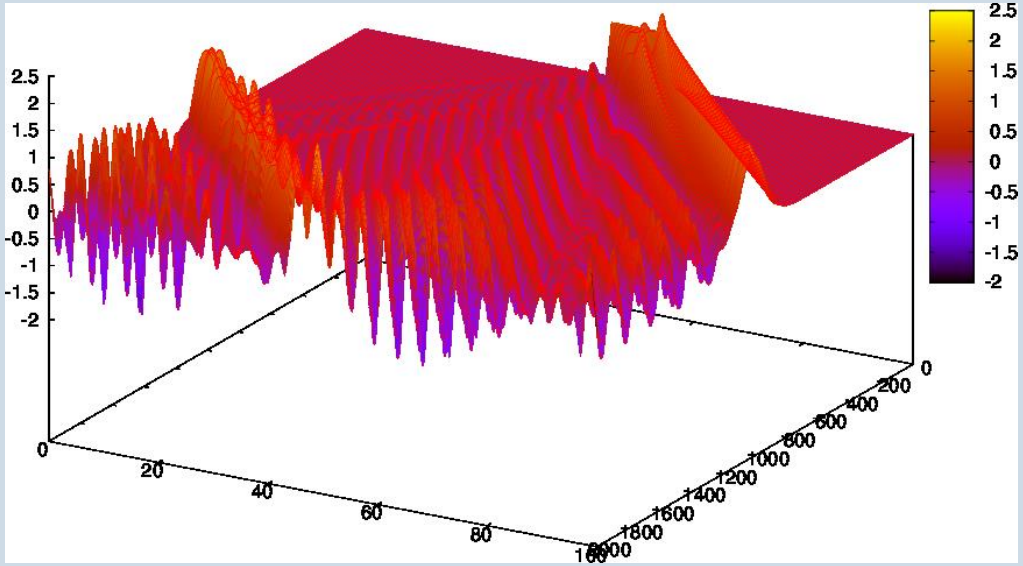
拡散方程式の解の様子。領域中央に矩形の物理量があり、時間経過（奥から手前）に従って緩やかに広がっていく様子が見える。

復習：波動方程式（移流方程式）

- 1次元波動方程式（移流方程式）の一般形は、次のようになる。
  - $\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2}$ または形式的に因数分解して、$\frac{\partial u}{\partial t} = -c \frac{\partial u}{\partial x}$ （x軸の正方向に進行する波）
- 従属変数uを格子点の位置での変数U_j^nで代表させると、差分化された移流方程式は、
  - $\frac{1}{\Delta t}\left\{U_j^{n+1} - U_j^n\right\} = -\frac{c}{2\Delta x}\left\{U_{j+1}^n - U_{j-1}^n\right\} \quad (j = 1,2,\ldots,N-1, n = 0,1,\ldots)$
- この式を漸化式として変形させると、
  - $U_j^{n+1} = U_j^n + \alpha\left\{U_{j+1}^n - U_{j-1}^n\right\}$
  - ただし、$\alpha = -\frac{c}{2}\frac{\Delta t}{\Delta x}$
- uの初期条件より、
  - $U_j^0 = \phi(x_j) \qquad (j = 0,1,\ldots,N)$

例題2: 移流方程式

- パラメータは以下の通り。
  - $\Delta t = 10^{-3}, \Delta x = 10^{-2}, c = 0.5, N = 100$
  - 初期時刻から2秒後までを計算する
- 初期条件は、
  - $u(x,0) = \begin{cases} 1 & (0.45 \leq x \leq 0.55) \\ 0 & (0 \leq x < 0.45, 0.55 < x \leq 1) \end{cases}$
- 境界条件は、
  - $U_0^n = U_{N-1}^n,\ U_N^n = U_1^n$ （周期境界条件）



移流方程式の解の様子。1階導関数を中心差分で近似したため、計算誤差によるノイズが領域内に広がっていく。

補足：描画プログラム

```
#!/bin/bash

gnuplot << EOD
set term postscript eps enhanced color solid
set output “test.eps"
set pm3d
set yrange [] reverse

splot "output.csv" matrix with dots
EOD
```

```
0.000000000000    0.004063391627    0.008127544189 ...
0.000000000000    0.004063543814    0.008127840086 ...
0.000000000000    0.004063694306    0.008128132608 ...
0.000000000000    0.004063843105    0.008128421762 ...
...
```

- プログラムの出力を、右図のように保存する。行が空間格子点の値、列が時間変化を示す行列ファイルのようになっている。ファイル名はoutput.csv としている。
- 適当な名前を付けて保存（draw.sh等）し、シェルスクリプトの実行を行うと、test.eps というベクター画像ファイル（EPSファイル）が出力される。

課題

- 例題2について、移流方程式の数値解で発生する数値誤差(ノイズ)を落としたい。
  - 風上差分方程式を用いて、例題2の数値解を改善するプログラムを作成せよ。
- 次の条件が与えられたとき、拡散方程式の数値解を求めるプログラムを作成せよ。
  - $\Delta t = 10^{-3}, \Delta x = 10^{-2}, \kappa = 0.02, N = 100$
  - 初期時刻から2秒後までを計算する
  - 初期条件は、$U_j^0 = \begin{cases} 1 & (0.45 \le x \le 0.55) \\ 0 & (0 \le x < 0.45, 0.55 < x \le 1) \end{cases}$
  - 境界条件は、$U_0^n = U_N^n = 0 \qquad (n = 0,1,\dots)$

提出方法

- 〆切：2017/12/15（金）講義開始まで
- メールにプログラムを添付
  - 主題：演習6レポート（学籍番号）
  - 宛先：tyamaura@riken.jp
  - 本文：なくてもOK
  - 添付：学籍番号_課題番号.f90 を2ファイル
    - 課題6-1：r???????_kadai06-1.f90
    - 課題6-2：r???????_kadai06-2.f90

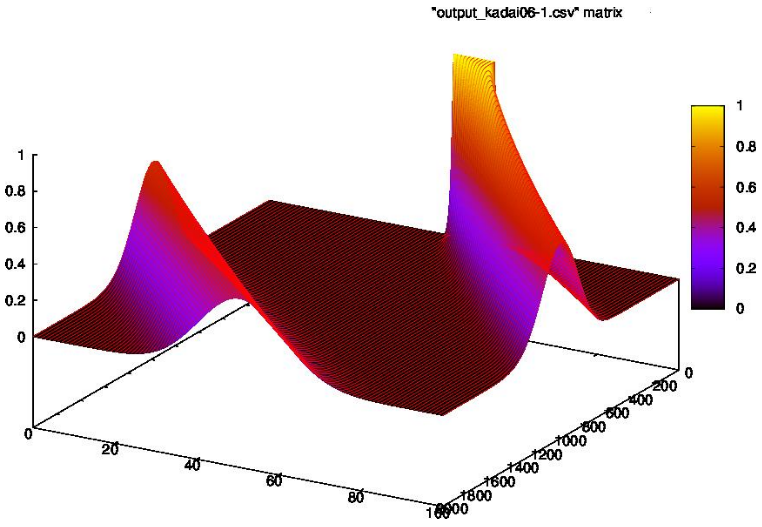
後期中間試験

- 12月5日（火） 2時間目（50分）
  - 関数電卓の持ち込みを許可（必須）
- 出題範囲
  - 第５章「常微分方程式」、第６章「偏微分方程式」
  - 関連する講義ノートも含む
- 出題レベル
  - 知識を問う問題
  - 数値計算プログラムに関する問題
  - 計算問題
    - 問題レベルは講義中の練習問題と同等

課題1：解答例

- サンプル6-1を書き換え
- 風上差分方程式の導出

```
program advection
  implicit none

  integer, parameter :: xmax = 100
  real(8), parameter :: tmax = 2.0d0
  real(8), parameter :: dt = 1.0d-3
  real(8), parameter :: dx = 1.0d-2
  real(8), parameter :: c  = 5.0d-1

  integer :: n, x
  real(8) :: u0(0:xmax)
  real(8) :: u1(0:xmax)
  real(8) :: a, b

  a = -c * dt / dx * 0.5d0
  b = abs(c) * dt / dx * 0.5d0

  do x = 0, xmax
    if( x >= 45 .and. x <= 55 ) then
      u0(x) = 1.0d0
    else
      u0(x) = 0.0d0
    end if
  end do

  do n = 1, int( tmax / dt )
    write(*,'(101(f18.12,1x))') u0(:)

    do x = 1, xmax-1
      u1(x) = u0(x) + a * ( u0(x+1) - u0(x-1) ) &
              + b * ( u0(x+1) - 2.0d0 * u0(x) + u0(x-1) )
    end do

    u1(0) = u0(xmax-1)
    u1(xmax) = u0(1)

    u0(:) = u1(:)
  end do

  stop
end program
```



課題2：解答例

- サンプル6-1を書き換え
- 拡散方程式の導出

```
program diffusion
  implicit none

  integer, parameter :: xmax = 100
  real(8), parameter :: tmax = 2.0d0
  real(8), parameter :: dt = 1.0d-3
  real(8), parameter :: dx = 1.0d-2
  real(8), parameter :: kp = 2.0d-2

  integer :: n, x
  real(8) :: u0(0:xmax)
  real(8) :: u1(0:xmax)
  real(8) :: a

  a = kp * dt / dx**2

  do x = 0, xmax
    if( x >= 45 .and. x <= 55 ) then
      u0(x) = 1.0d0
    else
      u0(x) = 0.0d0
    end if
  end do

  do n = 1, int( tmax / dt )
    write(*,'(101(f18.12,1x))') u0(:)

    do x = 1, xmax-1
      u1(x) = a * u0(x+1) &
              + ( 1.0d0 - 2.0d0 * a ) * u0(x) &
              + a * u0(x-1)
    end do

    u0(:) = u1(:)
  end do

  stop
end program
```