

実習テキスト

0.1 スカラー方程式の差分解法

スカラー方程式の差分解法を試してみましょう。以下のファイルが用意されています。

```
# ls scalar
Makefile  ftsc.f  pldt.pro  rddt.pro
```

プログラムは Fortran 言語を用いて書かれています。Fortran は数値シミュレーションの分野のプログラミングでもっとも用いられており、Fortran プログラムファイルの名前の最後は.fです。

0.2 プログラムのコンパイルと実行 (make)

次にプログラムをコンパイルする方法について説明します。ディレクトリ scalar に移動した後に make を実行します。するとプログラムがコンパイルされ、実行されます。出力ファイルは output です。

```
# cd scalar
# make
f77 -c -o ftsc.o ftsc.f
ftsc.f:
    MAIN:
f90 ftsc.f
a.out
```

0.3 結果の表示

結果の表示には IDL といった可視化プログラムを利用します。IDL は数値シミュレーション結果を可視化をするのによく用いられます。(※ idl は高価なソフトウェアなので、今回の実習で全員が起動できるだけのライセンスがありません。ライセンスが足りない場合、idl はデモモードで起動します。デモモードでは 7 分間は同様に動きます。).

0.3.1 IDL の起動 (idl)

まずは idl を実行してみましょう。

```
# idl
```

すると以下のようになり、IDL が起動します。

```
IDL Version 5.3 (sunos sparc). (c) 1999, Research Systems, Inc.  
Installation number: 70271.  
Licensed for use by: CHIBA-UNIV  
  
IDL>
```

0.3.2 データ読み込み (.r rddt)

データの読み込みには rddt.pro というプログラムを用います。以下のように入力してみてください。 .r は run を意味します。ファイル "output" からデータが読み込まれ、idl でデータを利用できるようになります。

```
IDL> .r rddt
```

0.3.3 データ読み込み (.r pldt)

データの表示には pldt.pro というプログラムを用います。以下のように入力してみてください。

```
IDL> .r pldt
```

0.3.4 IDL の終了 (end)

end を入力すると IDL を終了することができます。デモモードでは 7 分後に自動的に終了します。

```
IDL> end
```

0.3.5 IDL のプログラムについて

idlのプログラムは簡単な文法で書かれています。rddt.proやpldt.proを各自見てみましょう。

rddt.pro

```
*****|
;      Read data
;*****|
      nx=0
      ns=0

;-----|
; / Get grid number /
;-----|
      openr,1,'output'
      readf,1,nx,ns
      print,nx,ns

;-----|
; / Set hairesu /
;-----|
      j=intarr(nx,ns)
      u=fltarr(nx,ns)
      j_tem=intarr(1)
      u_tem=fltarr(1)

;-----|
; / Read data /
;-----|
      for k=0,ns-1 do begin
        for i=0,nx-1 do begin
          readf,1,j_tem,u_tem
          j(i,k)=j_tem
          u(i,k)=u_tem
        endfor
      endfor
      close,1
      end
```

pldt.pro

```
;*****|
;      Plot data
;*****|
      set_plot,'x'
;      set_plot,'ps'
;      device,filename='idl.ps'

x=intarr(nx)
y=fltarr(nx)

for k=0,ns-1 do begin
  for i=0,nx-1 do begin
    x(i)= j(i,k)
    y(i)= u(i,k)
  endfor
  if (k eq 0) then begin
    plot,x,y, yrange=[0,1.5]
  endif
  if (k ne 0) then begin
    oplot,x,y
  endif
endfor
end
```

0.4 夏の学校ホームページURL

メインページ

http://www.c.chiba-u.ac.jp/netlab/mhd_summer2001/index.html

1次元パッケージのデモのページ

<http://www.c.chiba-u.ac.jp/netlab/1dkadai/>

0.5 1次元パッケージの基礎

1次元パッケージのうちスカラー方程式の差分解法を動かしてみましょう。スカラー方程式を解くパッケージは以下のディレクトリにあります。

```
# ls md_scalar
Makefile  bnd.f      main.f      model.f     pldt.pro   rddt.pro
```

0.5.1 プログラムのコンパイル (make)

次にプログラムをコンパイルする方法について説明します。パッケージの一番上のディレクトリで make を実行します。すると以下のようにコンパイルを開始し、終了します。

```
# make
f77 -I../ -c -o cflhd.o cflhd.f
cflhd.f:
        cflhd:
(後略)
```

0.5.2 プログラムの実行 (make)

プログラムの実行は基本課題ディレクトリに移動して、make することで実行します。以下のように計算結果のファイル (out.cdf) を出力して終了します。

```
# cd md_scalar
# make
f77 -o a.out main.o model.o bnd.o \
-L../ -lhdmlw -lhdroe -lbc -lcommon -lnc \
-L/usr/local/netcdf/lib -lnetcdf
a.out
  write    step=      0 time= 0.000E+00 nd =  1
  write    step=     50 time= 0.125E+02 nd =  2
  write    step=    100 time= 0.250E+02 nd =  3
  stop     step=    100 time= 0.250E+02
  ### normal stop ###
```

0.5.3 結果の表示

idl を起動し、データの読み込み (rddt.pro) と表示 (pldt.pro) をしてみましょう。