VBAによる差分法の計算

富阪幸治

2000年2月10日

ビジュアルベーシック（VB）を元にしたマクロ記述言語であるVBA(Visual Basic Application)は、Excelで一連の手続きをマクロ化して、繁雑な操作を毎回毎回繰り返すことを避けるために用いられる。しかし、それのみならずVBはいわゆる「手続き記述」言語であり、差分法による流体力学計算方法を表現する能力を十分持っている。Excelに標準で附属するVBAは、ここで述べたような数値シミュレーションを安価に試してみることができるものとして興味が持てる。

一例を次のページに掲げる。これは、空間１次精度の風上差分を用いて、advection方程式

$$\frac{\partial \rho}{\partial t} + c\frac{\partial \rho}{\partial x} = 0,$$

（ただしcは一定）という問題を解くものである。差分化すると、

$$\rho_i^{n+1} = \rho_i^n - c\frac{\Delta t}{\Delta x}(\rho_i - \rho_{i-1}),$$

となる。$\nu \equiv c\Delta t/\Delta x$である。

1 プログラム

実際にはつながっているが少しずつ切って説明してゆく。プログラムを説明すると、SubとEnd Subで囲まれたかたまりが、プロシージャでマクロとしてExcelから実行可能な単位を形成している。

```
Sub initialize()
  Range("A3:L102").Select
  Selection.Delete Shift:=xlToLeft
  Call main(0)
End Sub
```

プロシージャinitialize()は、ワークシートの計算結果を書き出す部分を消去した後、配列a（ここに密度分布が格納されている）に初期状態を設定する。

```
Sub cal()
  Call main(1)
End Sub
```

プロシージャcal()は初期状態から（$\nu = 0.25$のもと）100時間ステップ密度分布を計算し、それを、10ステップおきに、ワークシートの１列を使って書き込んでゆく。

プロシージャgraph_Write()

```
Sub graph_Write()
  Call graph(0)
End Sub
```

は１１列の時間ステップごとの密度分布を線グラフにし、またプロシージャgraph_Delete()

```
Sub graph_Delete()
  Call graph(1)
End Sub
```

は、すでに作成されているグラフを消去する。これら４つのプロシージャは、Excelのメニューから、ツール→マクロ→マクロとすると選択画面に現れ、一つを選んで実行することができる。付録のExcelワークシートではオートシェイプである楕円に各マクロの呼び出しが登録されており（オートシェイプ上でマウスの右ボタンをクリックし、マクロの登録を選ぶと、登録されているものを知ったり、変更することができる）、これをクリックすることによっても実行できるようになっている。

2 計算および初期化

計算およびその初期化はプロシージャmain(L)を呼び出すことによって行なわれており、L=0の場合は初期化が、L=1の場合は時間ステップを進めてゆく計算が、行なわれる。この部分が差分による計算を実際に行なっている手続きを記述している部分である。

```
Sub main(L)
Dim a(200), b(200), nu As Single
Dim i, im, j, k As Integer
Dim lab As String

km=100

GoSub initialize
j = 1
lab = "初期値"
GoSub output

If L = 0 Then
  Exit Sub
End If

For k = 1 To 100
  GoSub OneStep
  If k Mod 10 = 0 Then
    j = k / 10 + 1
    lab = CStr(k)
    GoSub output
  End If
Next

Exit Sub
```

ここまでがmain()プロシージャの主プログラムに当たる部分でそこからGo Sub呼び出しによってinitialize:、OneStep:、output:のサブルーティンが呼び出される。L=0の場合には、Exit Subでステートメントでinitializeサブルーティンによって初期化を行ない、それをoutputサブルーティンでワークシートの１列目に書き出したのみで、プロシージャの実行を終る。これはinitialize()プロシージャから呼び出された場合である。initialize()プロシージャはその前に１列目より右側のセルを削除している。

L=1の場合は、その後、OneStepサブルーティンを必要回数（いまは100回）呼び出すことによって、時間進化系列を順に追跡する。

次にサブルーティンを一つずつ見てゆくと、

```
OneStep:
  For i = 2 To im
    b(i) = a(i) - nu * (a(i) - a(i - 1))
  Next
  For i = 2 To im
    a(i) = b(i)
  Next
Return
```

OneStepサブルーティンは差分法による時間発展の計算手続きを記述している部分である。つぎにinitializeサブルーティンは配列a(i)にi=1,. . .,39とi=61,. . .,100は0、i=40,. . ., 60は1を設定しているが、これは矩形の密度分布を与えたことに相当している。

```
initialize:
  im = 100
  nu = 0.25
  For i = 1 To im
    a(i) = 0
  Next
  For i = 40 To 60
    a(i) = 1
  Next
Return

output:
  Set ws = Worksheets("Sheet1")
  ws.Cells(2, j) = lab
  For i = 1 To im
    ws.Cells(i + 2, j) = a(i)
  Next
Return
```

```
End Sub
```

output サブルーティンは j 列にその時の配列 a の値を表示してゆく機能を持つ。j 列 2 行にラベルを、3 行以降に配列 a の値を表示する。最後の End Sub で main() プロシージャが終結する。

3　グラフ

ここからプロシージャgraph() を解説する。L=0 であった場合（graph_Write() から呼び出された場合）は、シート 1 の (150, 150, 320, 200) の矩形部分に線グラフ xlLine を作成する。この時、Source:=ws.Range(”a3:l102”) で a3 セルと L102 セルで囲まれた部分をデータとして用いて、PlotBy:=xlColumns で列方向にデータを分割してグラフ化することを指定している。

また L=1 であった場合（graph_Delete() から呼び出された場合）は、ChartObjects.Add で生成されたチャートオブジェクト（Static な変数 Grp1 に記憶されている）を消去する。

```
Sub graph(L)
Static Grp1
Select Case L
Case 0
  Set ws = Worksheets("Sheet1")
  Set Grp1 = ws.ChartObjects.Add(150, 150, 320, 200)
  Grp1.Chart.ChartWizard Source:=ws.Range("a3:k102"), _
    Gallery:=xlLine, _
    Format:=1, _
    PlotBy:=xlColumns, _
    CategoryLabels:=0, _
    SeriesLabels:=0, _
    HasLegend:=1, _
    Title:="時間進化", _
    CategoryTitle:="", _
    ValueTitle:="", _
    ExtraTitle:=""
Case 1
  Grp1.Delete
Case ""
Case Else
  MsgBox "Graph 入力間違い"
End Select
End Sub
```

付録

プログラム全体をまとめて掲げておく。

```
Sub initialize()
  Range("A3:k102").Select
  Selection.Delete Shift:=xlToLeft
  Call main(0)
End Sub

Sub cal()
  Call main(1)
End Sub

Sub graph_Write()
  Call graph(0)
End Sub

Sub graph_Delete()
  Call graph(1)
End Sub

Sub main(L)
Dim a(200), b(200), nu As Single
Dim i, im, j, k As Integer
Dim lab As String

km = 100

GoSub initialize
j = 1
lab = "初期値"
GoSub output

If L = 0 Then
  Exit Sub
```

```
End If

For k = 1 To km
  GoSub OneStep
  If k Mod 10 = 0 Then
    j = k / 10 + 1
    lab = CStr(k)
    GoSub output
  End If
Next

Exit Sub

OneStep:
  For i = 2 To im
    b(i) = a(i) - nu * (a(i) - a(i - 1))
  Next
  For i = 2 To im
    a(i) = b(i)
  Next
Return

initialize:
  im = 100
  nu = 0.25
  For i = 1 To im
    a(i) = 0
  Next
  For i = 40 To 60
    a(i) = 1
  Next
Return

output:
  Set ws = Worksheets("Sheet1")
  ws.Cells(2, j) = lab
  For i = 1 To im
    ws.Cells(i + 2, j) = a(i)
  Next
Return
```

```
End Sub

Sub graph(L)
Static Grp1
Select Case L
Case 0
  Set ws = Worksheets("Sheet1")
  Set Grp1 = ws.ChartObjects.Add(150, 150, 320,
  Grp1.Chart.ChartWizard Source:=ws.Range("a3:k1
    Gallery:=xlLine, _
    Format:=1, _
    PlotBy:=xlColumns, _
    CategoryLabels:=0, _
    SeriesLabels:=0, _
    HasLegend:=1, _
    Title:="時間進化", _
    CategoryTitle:="", _
    ValueTitle:="", _
    ExtraTitle:=""
Case 1
  Grp1.Delete
Case ""
Case Else
  MsgBox "Graph 入力間違い"
End Select
End Sub
```