有限要素法による定常 Navier-Stokes 方程式の

数値解析プログラム (Ver.1) 取り扱い説明書

平成７年３月１７日

〒２１４ 川崎市 多摩区 東三田１ー１ー１明治大学 生田校舎
明治大学大学院 理工学研究科 基礎理工学専攻 数学系 修士課程 ２年 工藤丈征

目 次

1 何はさておき、プログラムの紹介 2
1.1 はじめに . . . 2
1.2 あれやこれや . . . 2
1.3 問題はこれだ！ . . . 3
1.4 レイノルズ数 (Reynolds Number) . . . 4
1.5 質量保存則 (Flux Condition) . . . 4

2 データ・ファイルを作るぞ！ 5
2.1 領域 Ω の分割 . . . 5
2.2 要素と節点の対応 . . . 8
2.3 累積節点番号 . . . 9
2.4 ν & ρ . . . 10
2.5 境界条件 . . . 10
2.6 外力 . . . 11
2.7 まとめ . . . 12

3 いざ、実行 14
3.1 プログラムのコピーとコンパイル . . . 14
3.2 実行の仕方 . . . 14
3.3 流れの様子を見る . . . 15
3.4 レイノルズ数が高いものを計算するとき . . . 15

4 栄光の記録 16

A ポストスクリプト・ファイルの鉄人 19
A.1 ファイルの生成と書き換え . . . 19
A.2 ズームで見る . . . 19
A.3 印刷 . . . 20

図 目 次

1 重なり, すき間, 頂点が他の要素の辺上にある . . . 5
2 領域の分割 I . . . 6
3 領域の分割 II . . . 7
4 番号の付け方 . . . 8
5 例題の流れの様子 . . . 16
6 Problem 1 . . . 17
7 Problem 2 . . . 18

1 何はさておき、プログラムの紹介

1.1 はじめに

公開, 配布に関して, 特に条件を課しませんが, 何か起こった場合でも責任は負いません.

テストした環境は, SPARC station Sun OS 4.1.1 (24MB メモリ) で, X Windows system 上で動作する GLSC* を用いています.

おそらく, X Windows system 上で動作するワークステーションでは, 少しの修正で使えるはずです.

1.2 あれやこれや

このたび制作したプログラムは「有限要素法流体解析」[1] を参考にして制作しました. 少しコメントをしておくと

- 1 要素当り流速に 6 節点, 圧力に 3 節点を要する, 流速については 2 次, 圧力については 1 次の補間関数を用いています.
- 連立一次方程式は, 疎行列用内積形式の Gauss 法を用いています. [10]
- Newton 法の収束判定は, 誤差：

$$\sqrt{\frac{1}{N}\sum_{i=1}^{N}|\Delta x_i|^2}$$

が 10^{-8} より小さくなったときとしています.

ユーザーの立場として,「このプログラムは本当に, 本当に, 本当に, 正しい計算を行なっているのか？」と聞きたくなると思います. 制作者の立場として言えることは

- 17 ページの Problem 1 は厳密解を求めることができます. [12] その厳密解と計算結果を比較すると数値データでは若干の誤差が見えるものの, 流れの様子をグラフにしたときには “全く同じ” と表現していいくらいのものになります.
- 18 ページの Problem 2 は有限要素法で流体を解析する例題として良く登場するものです. [3], [6] Problem 1 のように数値データは比較できませんが, 文献に載っているグラフと同じものが描けていると思います.

尚, このレポートは, あくまでもマニュアルです. 流体力学や有限要素法について知識を得たい方は, 後ろにあげる参考文献などで勉強して下さい.

*GLSC とは, Graphic Library for Scientific Computing の略で, 龍谷大学 理工学部 数理情報学科の小林亮氏, 高橋大輔氏, 中野浩氏, 松木平淳太氏によって開発された科学技術計算の結果をディスプレイ上に表示したり, プリンタに出力したりするための簡単なグラフィックライブラリです.

1.3 問題はこれだ！

このプログラムは非圧縮粘性流を記述する定常 Navier-Stokes 方程式：

$$-\nu\Delta\boldsymbol{u} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \frac{1}{\rho}\nabla p = \boldsymbol{f},$$
$$\text{div } \boldsymbol{u} = 0$$

を有限要素法 (Finite Element Method, F.E.M.) を用いて解析するものです.
正確に問題を書くと次のようになります.

問題

$\mathbf{R}^2$ の有界領域を Ω とし, その境界を Γ とする. ただし Γ は

$$\Gamma = \Gamma_1 \cup \Gamma_2 \quad (\Gamma_1 \cap \Gamma_2 = \phi,\ \Gamma_1 \neq \phi) \tag{1}$$

であるとする. このとき方程式：

$$-\nu\Delta\boldsymbol{u} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \frac{1}{\rho}\nabla p = \boldsymbol{f} \quad (\text{in } \Omega), \tag{2}$$
$$\text{div } \boldsymbol{u} = 0 \qquad (\text{in } \Omega) \tag{3}$$

と, 境界条件：

$$\boldsymbol{u} = \boldsymbol{b} \qquad (\text{on } \Gamma_1), \tag{4}$$
$$\boldsymbol{\sigma} = \boldsymbol{S} \qquad (\text{on } \Gamma_2) \tag{5}$$

を満たす流速 $\boldsymbol{u} = (u, v)$, 圧力 p を求めよ[†].
ただし,

ν：動粘性係数, ρ：密度 $\cdots$ それぞれ正定数.
$\boldsymbol{f}$：外力 $\cdots\Omega$ 上のベクトル値関数.
$\boldsymbol{\sigma}$：表面力,

$$\boldsymbol{\sigma} = \left(-pn_x + \nu\rho\left(\nabla u + \frac{\partial \boldsymbol{u}}{\partial x}\right)\cdot\boldsymbol{n},\ -pn_y + \nu\rho\left(\nabla v + \frac{\partial \boldsymbol{u}}{\partial y}\right)\cdot\boldsymbol{n}\right),$$

$\boldsymbol{n} = (n_x, n_y)$ は境界における外向き単位法線ベクトル.
$\boldsymbol{b}, \boldsymbol{S}$：それぞれ Γ_1, Γ_2 上のベクトル値関数.

[†]このプログラムでは $\Gamma_2 = \phi$ の場合, 圧力はある 1 点 (x_p, y_p) で $p = p(x_p, y_p) = 0$ という条件を加えて任意定数の自由度を取り除いています. [7]

1.4 レイノルズ数 (Reynolds Number)

流体力学でもっとも重要な役割を演ずる無次元量にレイノルズ数というものがあります. ここでは流れを特徴づける代表的な長さ (例えば管の半径や円の直径など) を L , 代表的な速度の大きさを U としたときにレイノルズ数を

$$Re = \frac{UL}{\nu}$$

で定義することにします.

このプログラムでは中程度のレイノルズ数 ($Re \approx 40$) までしか計算できませんので, 次の節で説明するデータ・ファイルを作るときには, このことを考慮に入れて下さい.

1.5 質量保存則 (Flux Condition)

ここでは $\Gamma_2 = \phi$ とします.

今, 考えているのは非圧縮流体です. したがって領域内部での質量は保存されます. このことから境界から入ってくる流れの量 (流量) と出ていく流量は同じでなければなりません. よって境界値 $\boldsymbol{a}$ は次のような条件 (Flux Condition) を満たさなければなりません.

$$\int_\Gamma \boldsymbol{a} \cdot \boldsymbol{n} \, ds = 0. \tag{6}$$

この式は (3) の両辺を Ω 上で積分し, Gauss の発散定理を用いて得られます.

$$0 = \int_\Omega \mathrm{div}\ \boldsymbol{u} \, dx = \int_\Gamma \boldsymbol{u} \cdot \boldsymbol{n} \, ds = \int_\Gamma \boldsymbol{a} \cdot \boldsymbol{n} \, ds.$$

レイノルズ数と同様に, データ・ファイルを作るときには, このことを考慮に入れて下さい.

2 データ・ファイルを作るぞ！

この節では例題を使ってプログラムに入力する (領域の分割や境界条件などに関する) データ・ファイルの作り方を説明します. とりあえずこの節を最後まで読んで下さい.

例題[‡]

領域を

$$\Omega = \Big\{(x,y) \;\Big|\; (x-1)^2+y^2 > 4,\ x^2+y^2 < 25\Big\}$$

とし, その境界を

$$\begin{aligned} C_1 &= \Big\{(x,y) \;\Big|\; x^2+y^2 = 25\Big\}, \\ C_2 &= \Big\{(x,y) \;\Big|\; (x-1)^2+y^2 = 4\Big\} \end{aligned}$$

とする ((1) において $\Gamma = \Gamma_1 = C_1 \cup C_2$, $\Gamma_2 = \phi$).

また, 方程式 (2) において $\nu = 4.0 \times 10^{-2}$, $\rho = 1.0$, $\boldsymbol{f} = \boldsymbol{0}$ とし, 境界条件は

$$\begin{aligned} \boldsymbol{u}\Big|_{C_1} &= (-\sin\theta, \cos\theta), \\ \boldsymbol{u}\Big|_{C_2} &= (0,0) \end{aligned}$$

とする. この境界条件は Flux Condition (6) を満たしている.

この条件の下でこの例題を有限要素法を用いて解析せよ.

2.1 領域 Ω の分割

まず, 領域 Ω を 6 ページの図 2[§] のように三角形[¶] で分割し, さらに, その三角形に番号 [0] ~ を付けて下さい (1 からではなく 0 から数えて下さい). ただし, 次のような分割は避けて下さい.

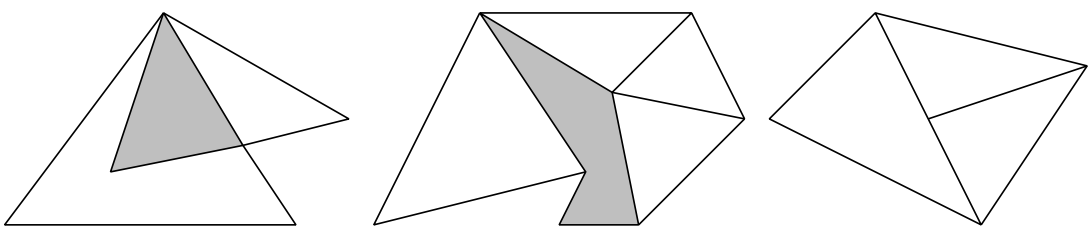

図 1: 重なり, すき間, 頂点が他の要素の辺上にある

この各三角形のことを有限要素 (以下, 略して要素) といい, 番号を要素番号といいます.

[‡][11] の Appendix II を改題.

[§]図 2 の要素番号はとても見辛いと思います. お詫びいたします.

[¶]四角形で分割する方法もあります.

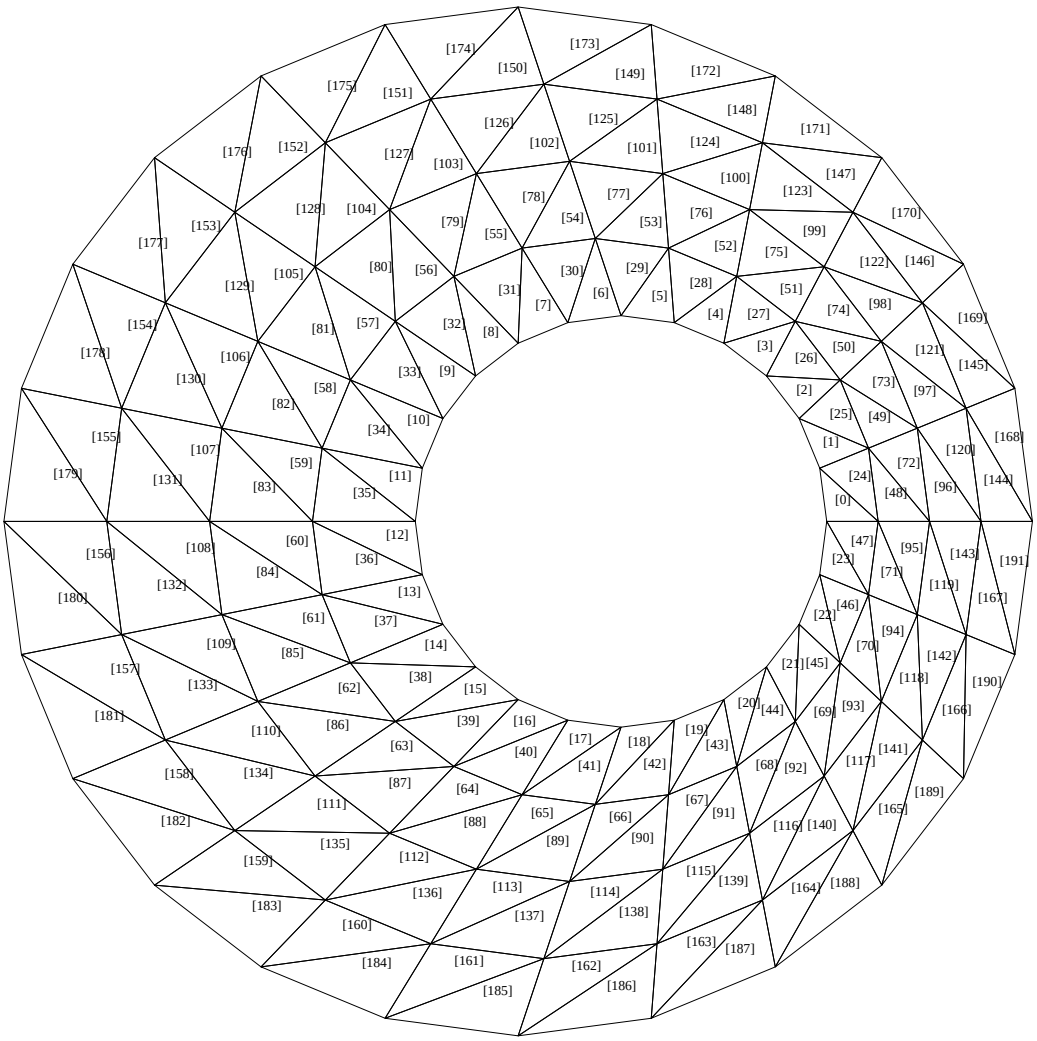
[0] [1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [13] [14] [15] [16] [17] [18] [19] [20] [21] [22] [23] [24] [25] [26] [27] [28] [29] [30] [31] [32] [33] [34] [35] [36] [37] [38] [39] [40] [41] [42] [43] [44] [45] [46] [47] [48] [49] [50] [51] [52] [53] [54] [55] [56] [57] [58] [59] [60] [61] [62] [63] [64] [65] [66] [67] [68] [69] [70] [71] [72] [73] [74] [75] [76] [77] [78] [79] [80] [81] [82] [83] [84] [85] [86] [87] [88] [89] [90] [91] [92] [93] [94] [95] [96] [97] [98] [99] [100] [101] [102] [103] [104] [105] [106] [107] [108] [109] [110] [111] [112] [113] [114] [115] [116] [117] [118] [119] [120] [121] [122] [123] [124] [125] [126] [127] [128] [129] [130] [131] [132] [133] [134] [135] [136] [137] [138] [139] [140] [141] [142] [143] [144] [145] [146] [147] [148] [149] [150] [151] [152] [153] [154] [155] [156] [157] [158] [159] [160] [161] [162] [163] [164] [165] [166] [167] [168] [169] [170] [171] [172] [173] [174] [175] [176] [177] [178] [179] [180] [181] [182] [183] [184] [185] [186] [187] [188] [189] [190] [191]

図 2: 領域の分割 I

次に 図 3‖ のように各要素の頂点と各辺の中点に番号 0 ~ を付けて下さい**. ただし, 0 番目は頂点にとって下さい. この点のことを節点といい, その番号のことを節点番号といいます.

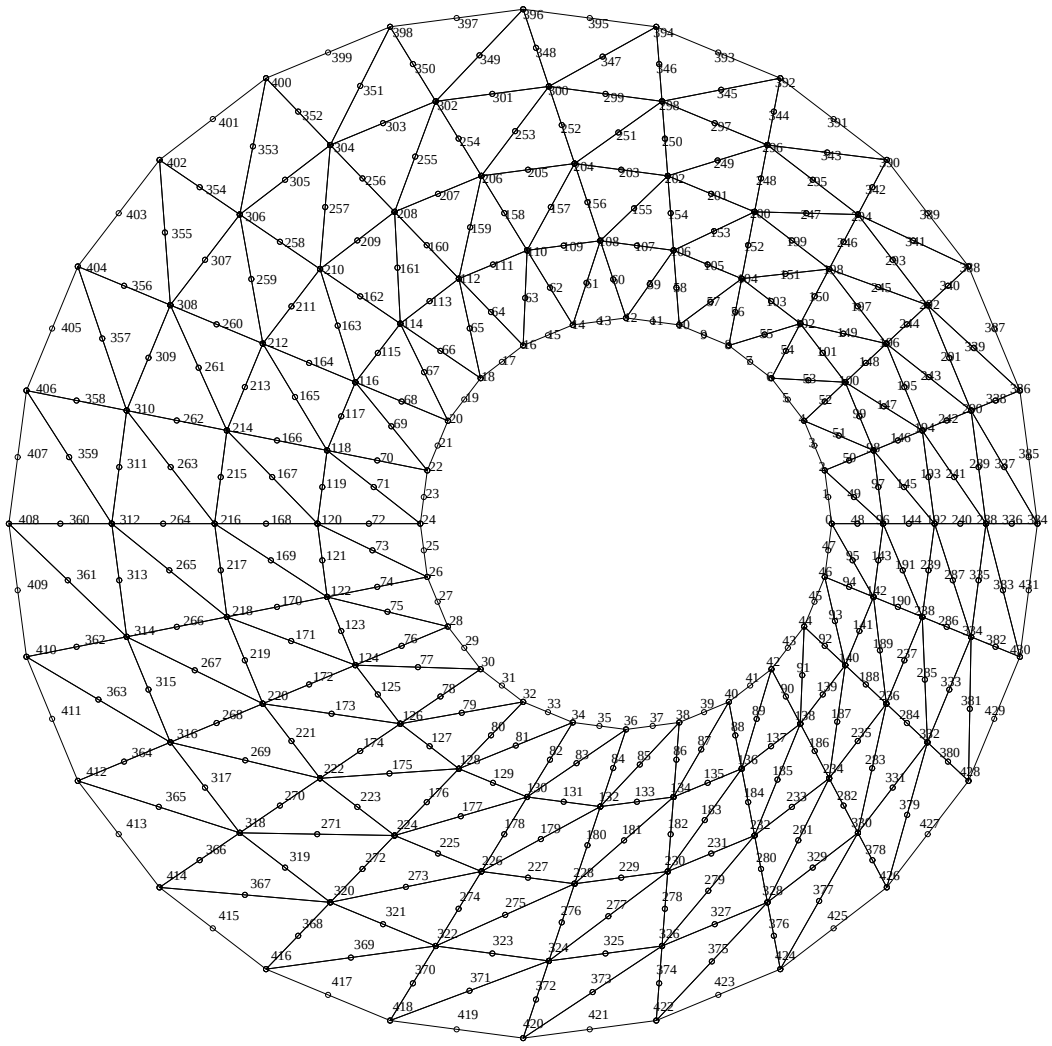

図 3: 領域の分割 II

要素と節点に番号が付けられたら, 要素と節点の個数をデータ・ファイル (ここではファイル名を `file.data` としましょう) に書き込んで下さい.

例題では要素数は 192, 節点数は 432 (0 から始まっているので最後の番号に 1 をたすとその個数になる).

`file.data` の中身は次のようになります.

‖図 3 の節点番号もとても見辛いと思います. お詫びいたします.

**節点のとり方にも色々バリエーションがありますが Inf-Sup 条件に関する注意が必要です. [8]

表 1: file.data の中身

```
192     432


```

次に, 各節点における x 座標値, y 座標値を並べてデータ・ファイルに書き込んで下さい. もちろん 0 から順番に.

例題では次のようになります.

表 2: file.data の中身

```
192     432

3.00000000000000000000     0.00000000000000000000
2.96592582628906820119     0.25881904510252073948
            ⋮                          ⋮
4.82962913144534056187     -1.29409522551260791623
4.91481456572267028093     -0.64704761275630395811

```

2.2 要素と節点の対応

1 つの要素には 6 つの節点があり, それぞれに節点番号がついています. 有限要素法では, 各要素における節点の節点番号は何番と何番と $\cdots$ と何番であるという情報が必要です.

そこで要素の 6 つの節点に第 1 節点, 第 2 節点, $\cdots$, 第 6 節点とローカルに番号付けをし, “第 i 要素の第 j 節点の節点番号は k 番である” というデータを作ります.

第 1 節点から第 6 節点の番号の付け方は次の図のとおりです.

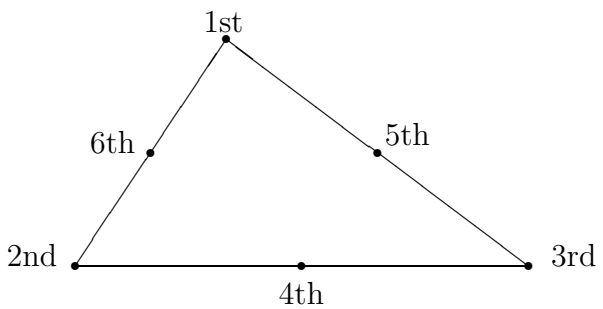

図 4: 番号の付け方

つまり, 第 1 節点を三角形の頂点のうちの 1 点と定め, その点から時計回りに $5, 3, 4, 2, 6$ とします. この番号付け以外の番号付けはしないで下さい.

第 i 要素の第 j 節点の節点番号を $num[i][j]$ (number の略) としましょう. そうすると例題では

$$
\begin{array}{lll}
num[0][1] = 0, & num[0][2] = 96, & num[0][3] = 2, \\
num[0][4] = 49, & num[0][5] = 1, & num[0][6] = 48. \\
num[1][1] = 2, & num[1][2] = 98, & num[1][3] = 4, \\
num[1][4] = 51, & num[1][5] = 3, & num[1][6] = 50. \\
\quad\vdots & \quad\vdots & \quad\vdots \\
num[191][1] = 288, & num[191][2] = 420, & num[191][3] = 384, \\
num[191][4] = 431, & num[191][5] = 336, & num[191][6] = 383
\end{array}
$$

となります.

これをデータ・ファイルに横並びで書き込んで下さい.

表 3: `file.data` の中身

```
          ⋮                              ⋮
4.91481456572267028093     -0.64704761275630395811

  0    96     2    49     1    48
  2    98     4    51     3    50
  ⋮     ⋮     ⋮     ⋮     ⋮     ⋮
334   428   430   429   382   381
288   430   384   431   336   383
```

2.3 累積節点番号

このプログラムでは混合型有限要素法 (または混合補間法) という方法を採用したので, 離散化したときの流速と圧力の未知数の数が異なります.

どういうことかというと, 流速はすべての節点において値を計算しますが, 圧力は節点によって値を計算したりしなかったりします. 具体的には圧力は要素の頂点では値を計算しますが, 辺の中点では値を計算しません.

したがって節点によっては, その節点における未知数の個数が, 流速の x 成分, y 成分の 2 個であったり, 圧力を加えた 3 個であったりします. そこで第 i 節点における未知数の個数を第 i 節点における自由度と呼び f_i と書くことにします (大雑把に言ってしまえば, その節点が頂点であれば自由度は 3 , 辺の中点であれば自由度は 2).

例題では

$$f_0 = 3,\ f_1 = 2,\ f_2 = 3,\ \cdots,\ f_{431} = 2.$$

第 0 節点は頂点に取ってあるので必ず $f_0 = 3$ になっているはずです. 確認を！

ここで第 i 節点における累積節点番号 $cum[i]$ (cumulative の略) を次のように定義します.

$$\begin{cases} cum[0] = 0, \\ cum[i] = \displaystyle\sum_{n=0}^{i-1} f_n \qquad (i \geq 1). \end{cases}$$

例題では

$$cum[1] = 3,\ cum[2] = 5,\ \cdots,\ cum[431] = 982.$$

そして $cum[431] + f_{431} = 984$ が例題の未知数の全個数になります (テストした環境では, 未知数の個数は千個ぐらいが丁度いい).

こうして累積節点番号と未知数の個数がわかったところでデータ・ファイルに書き込んで下さい.

例題では次のようになります.

表 4: `file.data` の中身

```
  ⋮     ⋮     ⋮     ⋮     ⋮     ⋮
288   430   384   431   336   383

 0
 3
 5
 ⋮
979
982

984
```

2.4 ν & ρ

ここは黙って ν と ρ の値をデータ・ファイルに書き込んで下さい.

例題では次のようになります.

2.5 境界条件

境界条件 (4) を第 1 種境界条件, 境界条件 (5) を第 2 種境界条件と呼ぶことにします.

まず, 第 1 種境界条件である節点の個数と第 2 種境界条件である節点の個数を数えてデータ・ファイルに書き込んで下さい.

例題では次のようになります.

表 5: file.data の中身

```
982

984

0.04   1.0
```

表 6: file.data の中身

```

0.04   1.0

96   0
```

次に, 第 1 種境界条件である節点の節点番号とそこでの境界値の x 成分, y 成分を並べてデータ・ファイルに書き込んで下さい (節点の順番はどうでもいい).
第 2 種境界条件も同様にして続けて書き込んで下さい.
例題では次のようになります.

表 7: file.data の中身

```

96   0

  0   0.00000000000000000000   0.00000000000000000000
384   0.00000000000000000000   1.00000000000000000000
  ⋮              ⋮                        ⋮
431   0.13052619222005257260   0.99144486137381027113
```

2.6 外力

最後に外力は, 座標値と同様に 0 から順番に各節点における外力の x 成分, y 成分を並べて書き込んで下さい.
例題では次のようになります.

表 8: file.data の中身

```
  ⋮                    ⋮                              ⋮
431    0.13052619222005257260    0.99144486137381027113

0.0      0.0
 ⋮        ⋮
0.0      0.0
```

2.7 まとめ

以上のことをまとめると, 例題のデータ・ファイル file.data は 13 ページの表 9 のようになります.

このようなデータ・ファイルを手作業で作るのは非常に大変です. 制作者は (規則性のある分割が可能であったので) 規則性のある分割にしてデータ・ファイルを作りました. 要素自動分割 [9] を取り入れられれば, 鬼に金棒なのですが $\cdots$.

表 9: file.data の中身

```
192     432

3.00000000000000000000     0.00000000000000000000
2.96592582628906820119     0.25881904510252073948
            ⋮                          ⋮
4.82962913144534056187     -1.29409522551260791623
4.91481456572267028093     -0.64704761275630395811

  0    96     2    49     1    48
  2    98     4    51     3    50
  ⋮     ⋮     ⋮     ⋮     ⋮     ⋮
334   428   430   429   382   381
288   430   384   431   336   383

 0
 3
 5
 ⋮
979
982

984

0.04   1.0

96   0

  0   0.00000000000000000000    0.00000000000000000000
384   0.00000000000000000000    1.00000000000000000000
  ⋮                ⋮                          ⋮
431   0.13052619222005257260    0.99144486137381027113

0.0      0.0
 ⋮        ⋮
0.0      0.0
```

3 いざ、実行

この節でも例題を使って説明をします.

3.1 プログラムのコピーとコンパイル

~kudo-t/FEM の中から fluid.c, matutil* [††], navier*, stokes* をコピーして下さい.

```
cp ~kudo-t/FEM/fluid.c .
cp ~kudo-t/FEM/matutil* .
cp ~kudo-t/FEM/navier* .
cp ~kudo-t/FEM/stokes* .
```

コンパイルの仕方は

```
cc fluid.c matutil.o -lglscd -lX11 -lm
cc -o navier.c -O4 matutil.o
cc -o stokes.c -O4 matutil.o
```

-O4 は最適化オプションです.

3.2 実行の仕方

まず, Newton 法の初期値となるデータを作ります. そのデータ・ファイル名を stokes.data としましょう. 実は, Newton 法の初期値とは 3 ページの問題で (2) を

$$-\nu\Delta \boldsymbol{u} + \frac{1}{\rho}\nabla p = \boldsymbol{f} \qquad (\text{in } \Omega) \tag{7}$$

にした Stokes 問題を解いたものです. さきほどの file.data を使って

```
stokes file.data stokes.data
```

として下さい.

次に,

```
navier file.data stokes.data navier.data
```

とすると目的の求めるべきデータ navier.data が得られます.

実行をすると収束の様子を表す " ||(Δu, Δv, Δp)|| =... " が何行か出力されます. このとき ||(Δu, Δv, Δp)|| が 1 つ前の値よりも大きくなってしまったら, 収束しないので正しい結果は得られません. 計算を中止して下さい.

[††] matutil* は桂田先生からいただいたプログラムです.

3.3 流れの様子を見る

グラフは GLSC を用いて描いています.

```
a.out file.data navier.data
```

とすると背景の色, 領域の色, 流れの様子を表す矢印↑ の色と相対的な長さを聞いてきます[‡‡]ので答えて下さい. そうすると, 流れの様子を表すグラフが出力されます.

GLSC のウィンドウをクリックするとグラフは消えます.

色の組合せや, 矢印↑ の長さは何回かグラフを出して調節して下さい.

例題のグラフは 図 5 のようになります.

もちろん

```
a.out file.data stokes.data
```

とすれば (7) の場合の Stokes 流れの様子が描かれます.

3.4 レイノルズ数が高いものを計算するとき

ここでは, 動粘性係数を変えてレイノルズ数を変えることにします.

2 つの動粘性係数の値を $\nu_1, \nu_2 (\nu_1 > \nu_2)$ とし, 他の値は全て同じで, 動粘性係数の値が ν_1 であるデータ・ファイルを `file.data1`, ν_2 であるデータ・ファイルを `file.data2` とします.

このとき

```
stokes file.data1 stokes.data1
navier file.data1 stokes.data1 navier.data1
navier file.data2 navier.data1 navier.data2
```

とすれば, レイノルズ数が高いものも計算できます.

普通にやっても Newton 法が発散してできないときは, この方法でやって下さい. この方法でも駄目なときは, 諦めて下さい.

[‡‡] モノクロでは白は白, それ以外は全て黒になります. なるべくカラーを使いましょう.

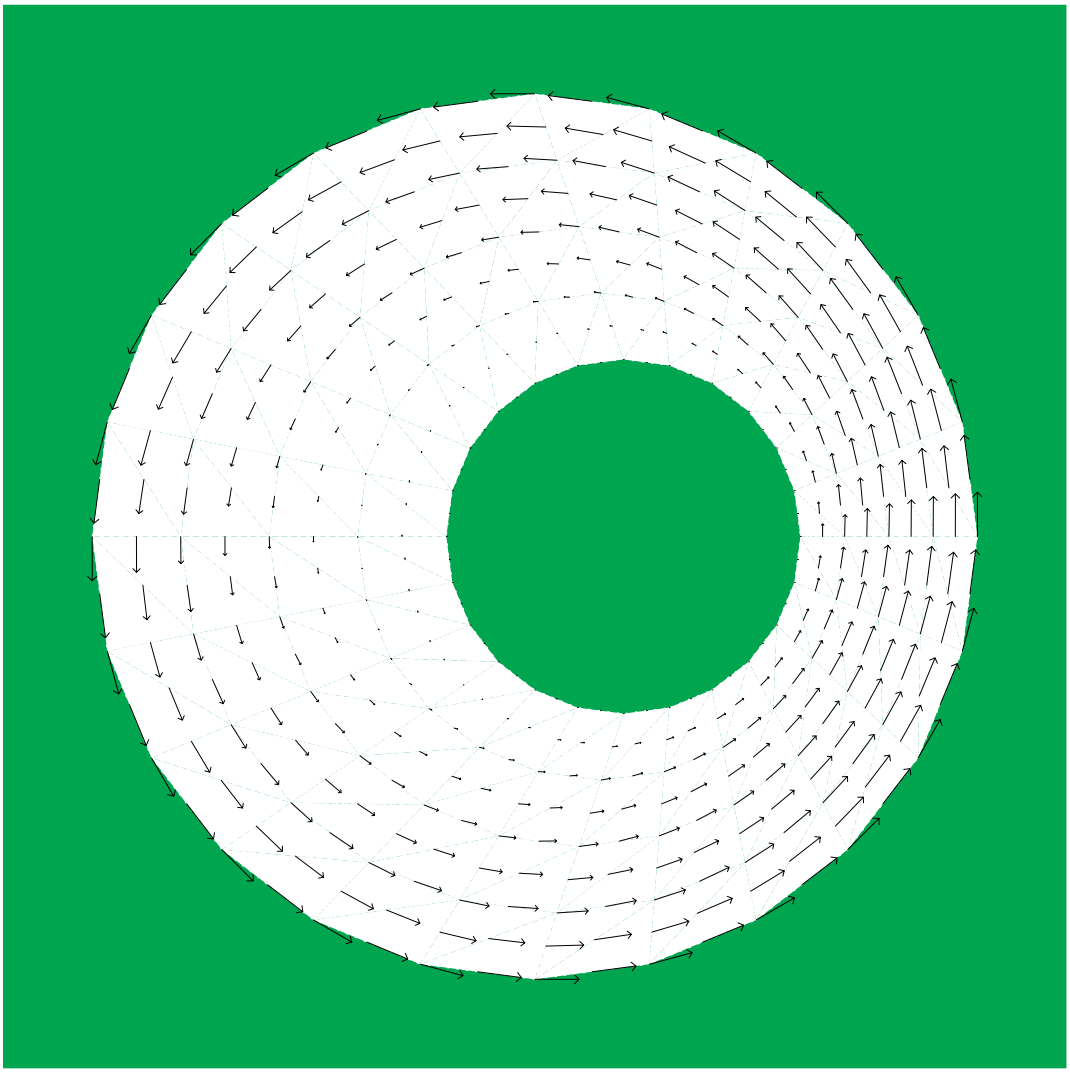

図 5: 例題の流れの様子

4 栄光の記録

この節では, このプログラムで実際に計算した例をいくつかあげます.

Problem 1 領域を

$$\Omega = \left\{(x,y) \,\middle|\, 1 < x^2 + y^2 < 9\right\}$$

とし, その境界を

$$\begin{aligned} C_1 &= \left\{(x,y) \,\middle|\, x^2 + y^2 = 1\right\}, \\ C_2 &= \left\{(x,y) \,\middle|\, x^2 + y^2 = 9\right\} \end{aligned}$$

とする.

また, 方程式 (2) において $\nu = 1.0$, $\rho = 1.0$, $\boldsymbol{f} = \boldsymbol{0}$ とし, 境界条件は

$$\begin{aligned} \boldsymbol{u}\Big|_{C_1} &= \boldsymbol{e}_r + \boldsymbol{e}_\theta, \\ \boldsymbol{u}\Big|_{C_2} &= \frac{1}{3}\boldsymbol{e}_r - \boldsymbol{e}_\theta \end{aligned}$$

とする. ただし $\boldsymbol{e}_r, \boldsymbol{e}_\theta$ は極座標系 (r, θ) の単位ベクトル.

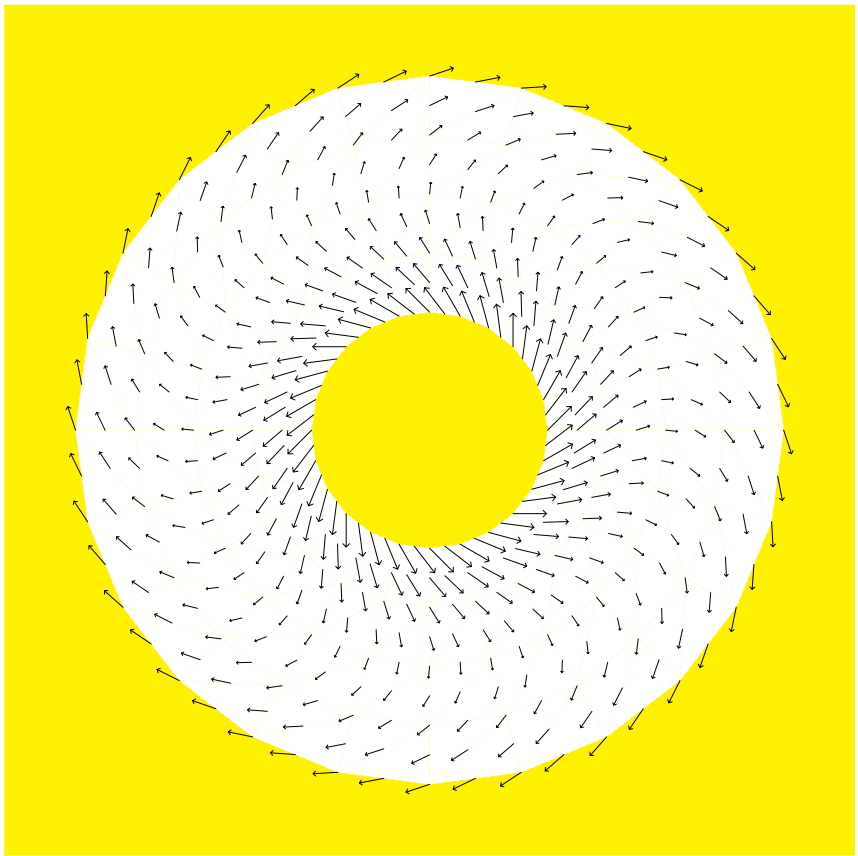

図 6: Problem 1

Problem 2 (Cavity Flow) 領域を

$$\Omega = \Big\{(x,y) \;\Big|\; 0 < x, y < 1\Big\}$$

とし, その境界を

$$\begin{aligned} C_1 &= \Big\{(x,1) \;\Big|\; 0 \le x \le 1\Big\}, \\ C_2 &= \partial\Omega \setminus C_1 \end{aligned}$$

とする.

また, 方程式 (2) において $\nu = 0.01$, $\rho = 1.0$, $\boldsymbol{f} = \boldsymbol{0}$ とし, 境界条件は

$$\begin{aligned} \boldsymbol{u}\Big|_{C_1} &= (1,0), \\ \boldsymbol{u}\Big|_{C_2} &= (0,0) \end{aligned}$$

とする.

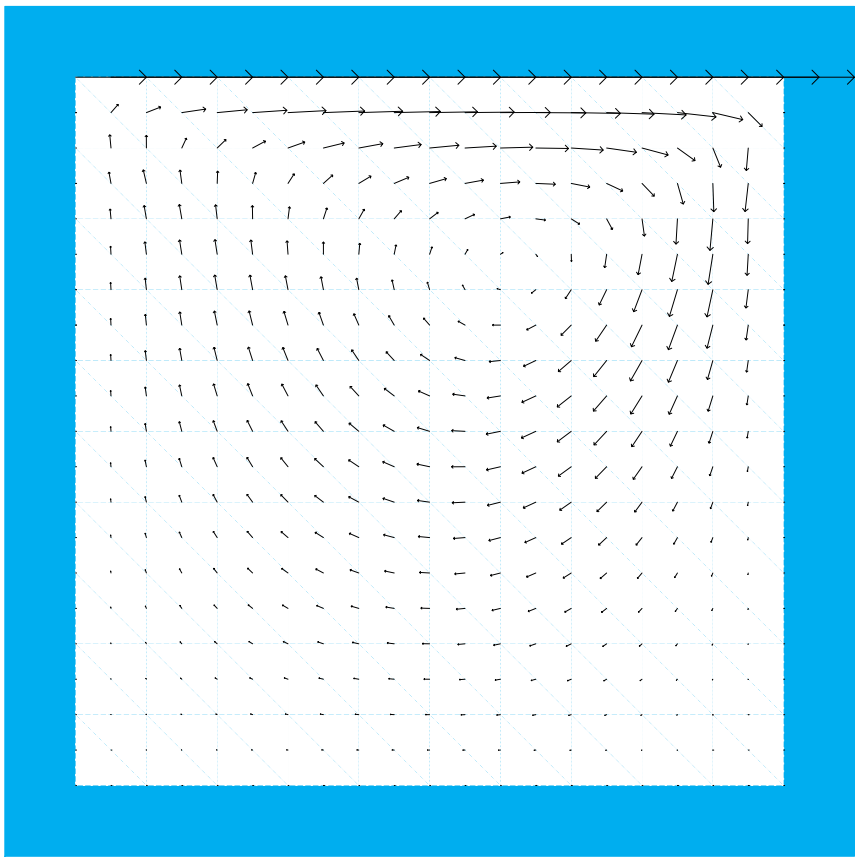

図 7: Problem 2

A ポストスクリプト・ファイルの鉄人

ここで紹介するプログラム (GLSC 付属の g_out, GNU の ghostview) は, このプログラムの制作者が作ったものではありません.

A.1 ファイルの生成と書き換え

a.out ··· を実行し終わるとディレクトリの中に Kudo というファイルができます (これをメタファイルという). そこで

```
g_out -ivm 1.3 Kudo
```

とするとディレクトリの中に Kudo.i00 という Illustrator 形式のポストスクリプト・ファイルができます[‡].

次に

```
ghostview Kudo.i00 &
```

とすると a.out ··· を実行したときと同じグラフが出力されますが, おそらく中心がずれていると思います. これでは困るのでポストスクリプト・ファイルを書き換えます. そのやり方を紹介します.

方法 1 (平行移動) ポストスクリプト・ファイルの最初に % で始まるコメント文が続きます. その何行かに渡ったコメント文の次の行に

x y translate

(x , y は整数) と書くとグラフが (x, y) だけ平行移動します.

方法 2 (フレーム枠の位置と大きさの変更) %%Boundingbox: の行に移動して下さい. そこに書かれている 4 つの数字はフレーム枠の x 座標値の最小値, y 座標値の最小値, x 座標値の最大値, y 座標値の最大値です. この値をうまく調節して下さい.

測定単位にはポストスクリプトの 1 単位である 1 / 7 2 インチ (約 0.35 ミリメートル) が採用されています.

ghostview ではマウス・カーソルの座標値が左上に表示されます. それを書き換えの目安にするのがコツです.

ghostview を終了するときは左上の File から Quit を選択して下さい.

グラフを中心にずらすだけなら方法 1 で十分だと思います. 方法 2 は LaTeX 文書へ取り込むとき, 無駄な空白部分を減らすのに効果的です. 方法 1 と方法 2, 両方やるのがいいでしょう.

A.2 ズームで見る

ghostview にはズーム機能があります. 拡大したい部分にマウス・カーソルを持っていき, どれかボタンをクリックすると zoom のウィンドウが開き, 拡大されたグラフが出力されます (右, 真中, 左の順で倍率が高い).

[‡]以上ＧＬＳＣの解説書を参照.

A.3 印刷

グラフを印刷するには, ポストスクリプト・ファイルの最後の %%EOF の上の行に

```
/#copies 1 def
showpage
```

と書き込んで下さい. そうすると `lpr Kudo.i00` で印刷できます.

また, 印刷するのであれば背景の色は, 緑か黄か水をお勧めします. 領域の色は白, 矢印↑ の色は黒がいいと思います. カラー・プリンターがあれば話は別ですが $\cdots$.

謝辞

このプログラムを制作するに当たり, 色々と相談にのっていたただいた明治大学 森本浩子教授, 桂田祐史専任講師, NEC C&C 研究所 藤尾秀洋氏にお礼を申し上げます.

また, Fortran, C 言語について解らない部分を教えてくれた, 上野康浩君, 北村潤君にもお礼を言いたい.

その他, プログラムの制作以外にも御指導いただいた多くの先生方に感謝いたします.

１９９５年３月 工藤丈征

参考文献

[1] 川原睦人, 「有限要素法流体解析」, 日科技連.

[2] 菊地文雄, 「有限要素法概説」, サイエンス社.

[3] 矢川元基, 吉村忍, 「有限要素法」, 培風館.

[4] 矢川元基, 「流れと熱伝導の有限要素法入門」, 培風館.

[5] 森正武, 「数値計算プログラミング」, 岩波書店.

[6] 保原充, 大宮司久明, 「数値流体力学」, 東京大学出版会.

[7] C. Cuvelier and A. Segal and A. A. van Steenhoven, " Finite element methods and Navier-Stokes equation," Dordrecht ; Tokyo : D. Reidel , 1986.

[8] Girault, V. and Raviart, P. A. "Finite element methods for Navier-Stokes Equations," Springer.

[9] 谷口健男, 「ＦＥＭのための要素自動分割」, 森北出版.

[10] 小国力, 「行列計算ソフトウェア」, 丸善.

[11] Temam, R., "Navier-Stokes Equations," North-Holland, Amsterdam, 1977.

[12] Morimoto, H., A Solution to the Stationary Navier-Stokes Equation under the Boundary Condition with Non-Vanishing Outflow, Memoirs of the Inst. of Sci. and Tec. Meiji Univ. Vol. 31 No. 2 1992.

[13] 今井功, 「流体力学 (前編)」, 裳華房.

[14] Ｌ. Ｄ. ランダウ, Ｅ. Ｍ. リフシッツ, (竹内均 : 訳), 「流体力学 1」, 東京図書.

[15] Adobe Systems, (野中浩一：訳), 「PostScript チュートリアル & クックブック」, アスキー出版局.

[16] Adobe Systems, (松村邦仁：訳), 「PostScript リファレンス・マニュアル」, アスキー出版局.

[17] David A. Holzgang, (大木敦雄：訳), 「ポストスクリプト入門」, パーソナルメディア.