

Numerical simulation of flow around three circular cylinders using pixel data and fictitious domain

Junichi Masuda* Tomoko Ojima† Hideyuki Koshigoe*

*Graduate school of Engineering, Chiba University †Toshiba Corporation

画像のピクセルと仮想領域を用いた3つの円柱周りの 数値シミュレーション

増田 潤一* 小島 智子† 腰越 秀之*

* 千葉大学大学院工学研究科 † (株) 東芝

Abstract. In this report we propose a numerical method using pixels in a digital image and show a numerical simulation for an incompressible viscous flow around three circular cylinders. The characteristic we present here combines the digital image, the fictitious domain and the finite difference method. Since the digital image consists of so many pixels and each of pixels forms a small square, the finite difference method under the regular mesh is applied to Navier-Stokes equations in complex shaped domains. Hence the numerical algorithm is easily constructed and it will be widely applied to various problems in the environmental mathematics.

概要 本研究では、デジタル画像のピクセル情報を用いた数値計算法を提案し、3つの円柱周りの非圧縮性流れの数値シミュレーションを示す。デジタル画像は、極微小な正方形をしたピクセルから構成され、各ピクセルは輝度値の情報をもっている。本研究の特徴は、デジタル画像のピクセル情報を数学解析の仮想領域法と結合させた有限差分法である。

*1

*1 Corresponding author: Hideyuki Koshigoe (Email address:koshigoe@faculty.chiba-u.jp)

1. はじめに

本研究では画像のピクセル情報を基に、仮想領域と正方格子による有限差分法の数値計算法を示す. 対象とする問題は Fig.1 のように、一様な流れの中に置かれた3つの円柱周りの流れの数値シミュレーションである.

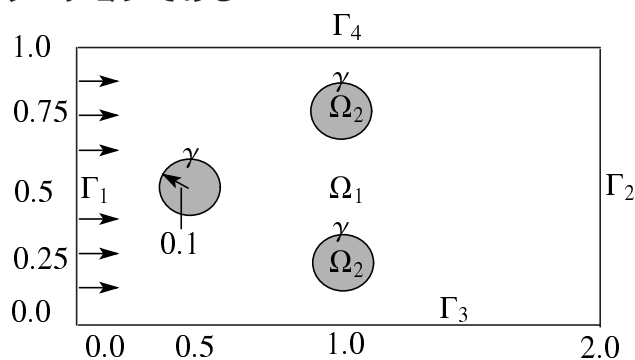


Fig.1. The domain of analysis.

Fig.1 は, Ω が領域 $(0,2) \times (0,1)$, Ω_2 が Ω 内の3つの円柱領域, $\Omega_1 = \Omega \setminus \overline{\Omega_2}$, および, Γ_1 が流入口, Γ_2 が流出口, Γ_3, Γ_4 が壁を表している.

以後用いる記号 \mathbf{v} , p , および, Re は流体の速度ベクトル, 圧力, および, レイノルズ数を表す. \mathbf{n} は境界での外向き単位法線ベクトルである.

本研究で考察する Navier-Stokes 方程式と境界条件は次の通りである.

3円柱を除いた領域 Ω_1 における Navier-Stokes 方程式と連続の式:

$$(1.1) \quad \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\nabla p + \frac{1}{Re} \Delta \mathbf{v} \quad \text{in } (0, T) \times \Omega_1$$

$$(1.2) \quad \nabla \cdot \mathbf{v} = 0 \quad \text{in } (0, T) \times \Omega_1.$$

円柱 Ω_2 の側面 γ 上の \mathbf{v} , p の境界条件:

$$(1.3) \quad \begin{cases} \frac{\partial p}{\partial \mathbf{n}} = 0 & \text{on } (0, T) \times \gamma \\ \mathbf{v} = 0 & \text{on } (0, T) \times \gamma. \end{cases}$$

外周 $\Gamma_i (i = 1, 2, 3, 4)$ 上の \mathbf{v} , p 境界条件:

$$(1.4) \quad \begin{cases} \mathbf{v} = (0.2, 0.0) & \text{on } (0, T) \times \Gamma_1 \\ \frac{\partial \mathbf{v}}{\partial \mathbf{n}} = 0 & \text{on } (0, T) \times \Gamma_2 \\ \mathbf{v} = 0.0 & \text{on } (0, T) \times (\Gamma_3 \cup \Gamma_4) \\ p = 0.0 & \text{on } (0, T) \times \Gamma_2 \\ \frac{\partial p}{\partial \mathbf{n}} = 0 & \text{on } (0, T) \times (\partial \Omega_1 \setminus \Gamma_2). \end{cases}$$

および, 初期条件 :

$$(1.5) \quad v = 0.0, \quad p = 0.0 \quad \text{in } (0, T) \times \Omega_1.$$

この問題の数値計算において, 我々は正方格子を用い, 流速ベクトルと圧力を同一の正方格子点上にとる. 正方格子を用いる着想の原点はデジタル画像にある. 実際, デジタル画像の基礎単位は極微小な正方形のピクセルから構成されていながら, 輪郭がはっきりした画像になっている. そこから, 任意の領域がピクセルで近似できるので, ピクセル上の有限差分法が構成できる. この構成法を示すことが目的であり, [1,6,7,8] を基に考察している.

本研究の内容は, 次の通りである :

1. 各ピクセルの輝度値を用いて, セルの概念を定義する.
2. 各セルが円柱領域 Ω_2 に属するかの判定式を導入する.
3. セルと仮想領域法を用いた有限差分法の数値計算アルゴリズムを示す.
4. (1.1)-(1.5) の数値シミュレーションを示す.
5. 5 つの円柱周りの流れを示す.

以後, 便宜的に, 3 つの円柱が占める領域を障害物と呼ぶことにする.

2. 画像のピクセルによる領域 Ω_1 と Ω_2 の決定方法

2.1 領域 (Ω_1, Ω_2) とピクセル

Ω_1 と Ω_2 を含む対象領域 Ω を画像的に確定する. それは Ω_1 を「白」, Ω_2 を「黒」とする 2 値画像の作成である. 今回, その画像は Fig.2 のような 810×410 ピクセルになっている. 2 値画像の各ピクセルは黒が「0」, 白が「255」の輝度値をもつので, その情報を数値計算に援用する.

Fig.2 は 2 値画像のピクセルの輝度値の様子を示している. 左側の 10×10 ピクセルの輝度値の様子をみるために拡大すると, 右側のように白黒のピクセルの配置になっている.

2.2 ピクセルによるセルの定義

画像のピクセルを縦・横に 10 個ずつとった 10×10 個のピクセルの集合を我々はセルと呼び, セルの中心に数値計算上の格子点を配置する. この格子点の属性は, セル内のピクセルの輝度値によって決定される.

Fig.3 は, セルと格子点の関係を示している. 正方形の破線がセルでその中心に格子点を配置している.

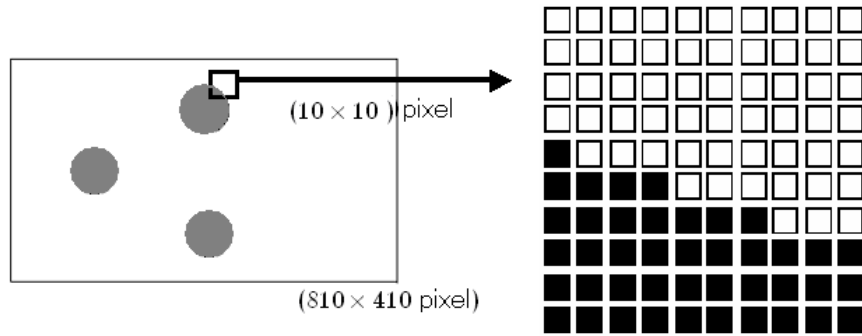


Fig.2. The image expanded partly.

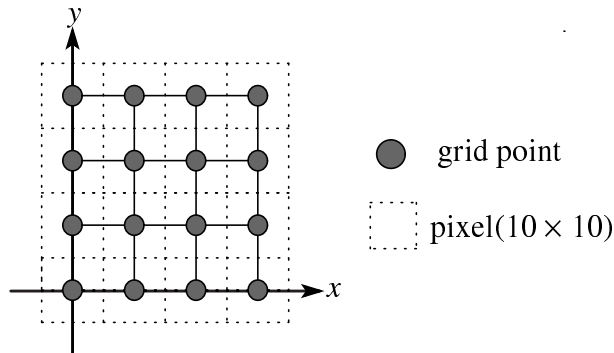


Fig. 3. The relation between grid point and pixel.

2.3 ピクセルの輝度値による Ω_1, Ω_2 の判定

各セル (格子点) が画像として領域 Ω_2 の外か内にあるかを決定する方法を述べる. 2 値化によって処理された画像の各ピクセルは「0」か「255」の輝度値をもっているので, 各セルにおいて, セル内の 10×10 個のピクセルの輝度値を用いて, (2.1) 式を定義する.

$$\begin{aligned}
 (2.1) \quad y &= \frac{\text{セル内のピクセルの輝度値の総和 (x)}}{\text{白の輝度値とセル内のピクセル数の積}} \\
 &= \frac{x}{255 \times 100} \quad (0.0 \leq y \leq 1.0)
 \end{aligned}$$

この (2.1) を用いて, 数値 y が与えられた閾値 z ($0.0 \leq z \leq 1.0$) より大きいならば, そのセルは Ω_2 の外にあるとし, 数値 y が z 以下ならば, そのセルは Ω_2 内にあると決める. 以上より, 閾値 z を定めると (2.1) により, セルが領域 Ω_1, Ω_2 のどちらに属するかを自動的に決定できる. 我々の方法はこの意味でメッシュレスである. 本研究では, $z = 0.0$ として領域を決定している.

Fig.4 の①, ②, ③はセル内のピクセルの様子を示している. $z = 0.0$ の場合, ①, ②のセルは Ω_2 の外, ③は Ω_2 の内となる.

今後, 次の記号を用いる. $\widetilde{\Omega}_1 = \{y \neq 0.0 \text{ であるセルの集合}\}$, $\widetilde{\Omega}_2 = \{y = 0.0 \text{ であるセルの集}$

合}, および, $\Omega = \widetilde{\Omega}_1 \cup \widetilde{\Omega}_2$.

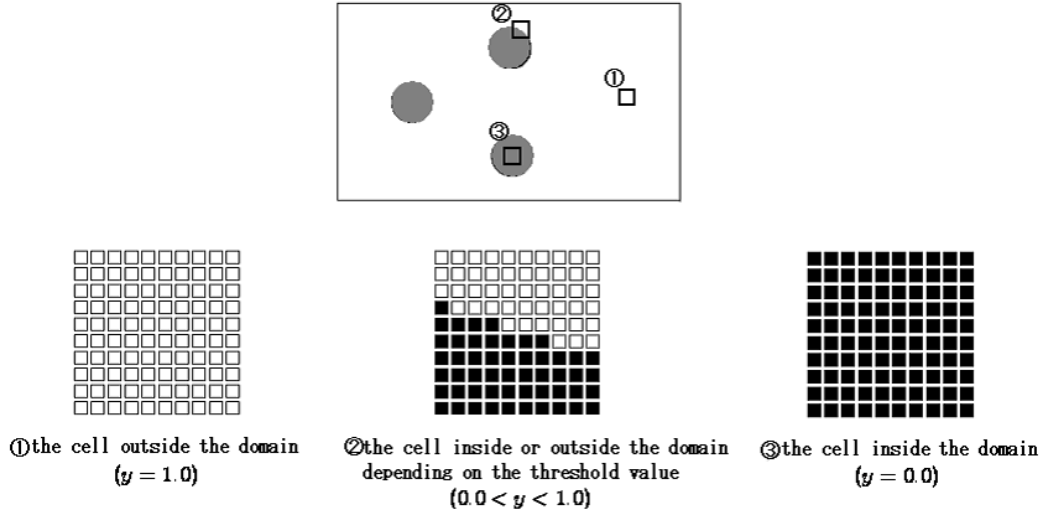


Fig.4. The decision on the domain by the formula (2.1).

3. 数値計算法 (セルの輝度値を用いた仮想領域法)

3.1 Navier-Stokes 方程式の再定式化

3 円柱の外側の領域 Ω_1 における Navier-Stokes 方程式 (1.1), 連続の式 (1.2);

$$(1.1) \quad \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\nabla p + \frac{1}{Re} \Delta \mathbf{v} \quad \text{in } (0, T) \times \Omega_1$$

$$(1.2) \quad \nabla \cdot \mathbf{v} = 0 \quad \text{in } (0, T) \times \Omega_1.$$

に対する数値計算を考察する.

今回, 圧力ポアソン方程式;

$$(3.1) \quad -\Delta p = \frac{\partial(\nabla \cdot \mathbf{v})}{\partial t} + \nabla \cdot ((\mathbf{v} \cdot \nabla) \mathbf{v}) - \frac{1}{Re} \Delta(\nabla \cdot \mathbf{v}) \quad \text{in } (0, T) \times \Omega_1$$

を次の境界条件の下で解く.

円柱 Ω_2 の側面 γ 上の p の境界条件:

$$(3.2) \quad \frac{\partial p}{\partial \mathbf{n}} = 0 \quad \text{on } (0, T) \times \gamma.$$

および, 外周 $\Gamma_i (i = 1, 2, 3, 4)$ 上の p の境界条件:

$$(3.3) \quad \begin{cases} p = 0.0 & \text{on } (0, T) \times \Gamma_2 \\ \frac{\partial p}{\partial \mathbf{n}} = 0 & \text{on } (0, T) \times (\partial \Omega_1 \setminus \Gamma_2). \end{cases}$$

この場合, (3.1)-(3.3) に仮想領域法を用いて, 円柱 γ 周りのノイマン境界条件を満たす近似解を構成する. その方法として, 河原田 [2] や J.L.Lions[5] を適用する. つまり,

$$(3.4) \quad a(x, y) = \begin{cases} 1 & \text{on } \Omega_1 \\ \varepsilon & \text{on } \Omega_2 \end{cases}$$

を導入し, 方程式 (3.1) を次の方程式;

$$(3.5) \quad -\nabla \cdot (a(x, y)\nabla p) = \frac{\partial(\nabla \cdot v)}{\partial t} + \nabla \cdot ((v \cdot \nabla)v) - \frac{1}{Re}\Delta(\nabla \cdot v) \quad \text{in } (0, T) \times \Omega$$

および, 外周の境界条件 (3.3) の問題に再定式化する. 河原田や J.L.Lions の解析方法を用いると, $\varepsilon \rightarrow 0$ の時, (3.5), (3.3) の解がノイマン境界条件;

$$\frac{\partial p}{\partial \mathbf{n}} = 0 \quad \text{on } (0, T) \times \gamma$$

を満たす解に収束することが示せる. 本研究ではこの立場から, ピクセル情報を用いた Navier-Stokes 方程式を考察する.

3.2 ピクセル情報に基づく数値計算アルゴリズム

画像のピクセル情報によって, セルによる領域 $\widetilde{\Omega}_1$ と $\widetilde{\Omega}_2$ が決まっているので,

$$\widetilde{a}(x, y) = \begin{cases} 1 & \text{on } \widetilde{\Omega}_1 \\ \varepsilon & \text{on } \widetilde{\Omega}_2 \end{cases}$$

とにおいて, (3.5) を

$$(3.6) \quad -\nabla \cdot (\widetilde{a}(x, y)\nabla p) = \frac{\partial(\nabla \cdot v)}{\partial t} + \nabla \cdot ((v \cdot \nabla)v) - \frac{1}{Re}\Delta(\nabla \cdot v) \quad \text{in } (0, T) \times \Omega$$

の形に再定式化する. この時, この問題 (3.6), (3.3) は数値的に全領域 Ω で解ける.

次に, (3.6) の数値解 p を用いて, 速度ベクトル v を (1.1) で解く. 具体的には,

$$(3.7) \quad \frac{\partial v}{\partial t} + (v \cdot \nabla)v = -\nabla p + \frac{1}{Re}\Delta v \quad \text{in } (0, T) \times \widetilde{\Omega}_1$$

$$(3.8) \quad v = \mathbf{0} \quad \text{in } (0, T) \times \widetilde{\Omega}_2$$

および, 外周 $\Gamma_i (i = 1, 2, 3, 4)$ 上の v の境界条件;

$$(3.9) \quad \begin{cases} v = (0.2, 0.0) & \text{on } (0, T) \times \Gamma_1 \\ \frac{\partial v}{\partial \mathbf{n}} = 0 & \text{on } (0, T) \times \Gamma_2 \\ v = 0.0 & \text{on } (0, T) \times (\Gamma_3 \cup \Gamma_4) \end{cases}$$

の下で数値計算を行う.

以上より、本研究で用いる数値計算アルゴリズムは次の通りである。

- (1) 全領域 Ω において、(3.6), (3.3) の圧力ポアソン方程式を数値的に解く。
- (2) 手順 (1) で得られた圧力を用いて、全領域 Ω における (3.7)-(3.9) の Navier-Stokes 方程式を数値的に解いて、速度を求める。
- (3) 時間ステップを 1 つ進め、上のステップ (1), (2) を繰り返して、圧力と速度を順次求めていく。

なお、(3.7) の移流項に関しては、3 次精度上流差分を用いている [4,9].

4. 数値計算シミュレーション

最初の問題 (1.1)-(1.5) を上記の数値計算アルゴリズム (1)-(3) によって計算する。この計算のパラメータは次に示す通りである；

画像のピクセル数 : 332,100

レイノルズ数: $Re = 3,000$

時間の刻み幅: $\Delta t = 0.01$

および、障害物設定の閾値を $z = 0.0$ としている。つまり、1 セル (10×10 ピクセル) の中で、白ピクセルが一つも無く、黒ピクセルだけで占有されている状態である。

数値計算結果は、Fig.5 のようになる。

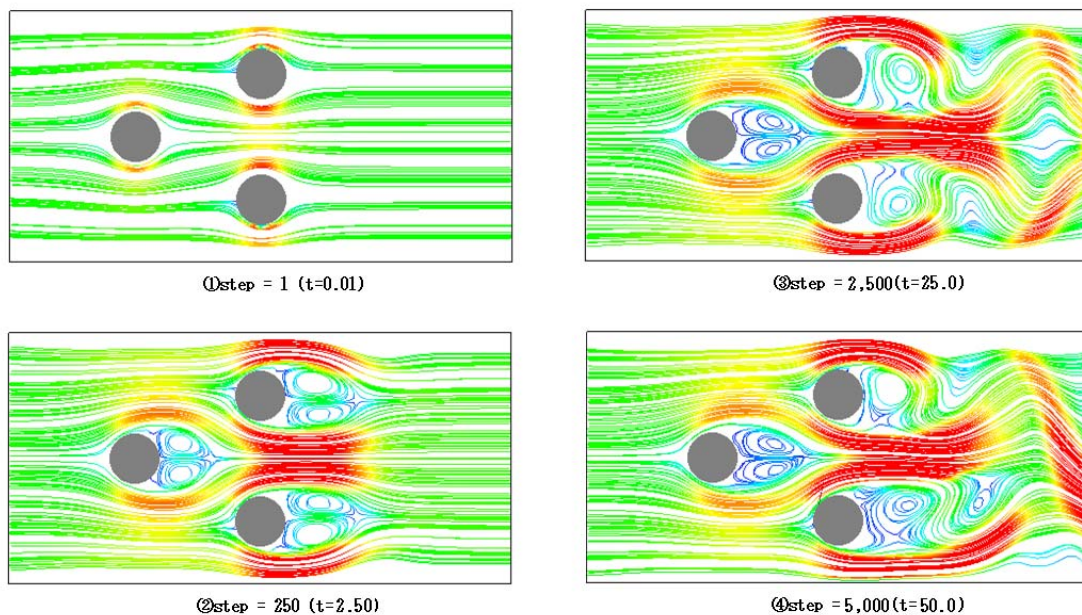


Fig.5. The result of numerical computation with the stream line by time step.

Fig.5 の数値計算結果を見てみると、

- ① 時間 step=1 ($t=0.01$) においては円柱に沿って流れが起き始め、
 - ② 時間 step=250 ($t=2.50$) には、3つの円柱の後方において一对の渦対が現れる。
- そして、
- ③ 時間 step=2,500 ($t=25.0$) になると、前方の渦対の対称性は崩れないが、後方の円柱においてはその対称性が崩れ、上下において対称なカルマン渦へと発展してゆく。
- しかし、
- ④ 時間 step=5,000 ($t=50.0$) では、後方カルマン渦の上下の対称性も崩れてしまい、全体として複雑な渦構造をなしてゆくことが分かる。

5. 5つの円柱周りの流れ

最後に、5つの円柱周りの流れ問題に適用する。この解析モデルは、上下に一方向の様な流れを与えた中に、5つの円柱を障害物として配置した Fig.6 の場合である [10].

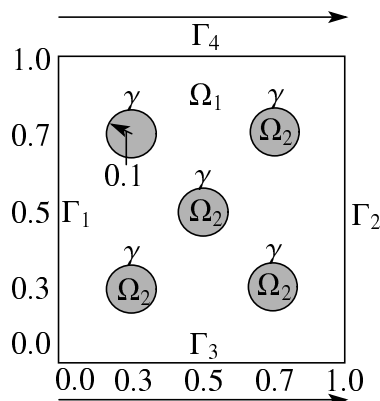


Fig.6. The domain of analysis

次のデジタル画像と計算パラメータのもとで数値計算を行う;

レイノルズ数: $Re = 1,000$

時間の刻み幅: $\Delta t = 0.001$

画像のピクセル数: 656,100

セルの個数: 6,561 (= 81×81)

障害物設定: (2.1) の閾値 $z = 0.0$

数値計算結果は、次の Fig.7 である。それは時間ステップ数 10,000 の時の流れ図である。このように今回示した数値計算法は、メッシュ生成を必要とせず、極微小な正方形の画像ピクセルの配置情報のみで計算できる特徴がある。

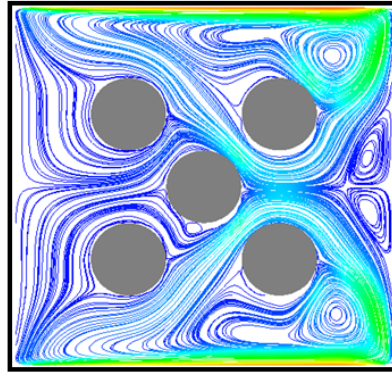


Fig.7. The result of numerical computation with the stream line (step=10,000,t=10).

6. まとめ

今回、2次元空間において、時間ステップの経過による流れの数値計算を行い、3つの円柱周りの数値シミュレーションを示した。この結果は、いままで知られている1つの円柱周りの流れとは異なり、3つの円柱のうち先頭の円柱の後方に現れる渦対は時間が経過してもその渦対の対象性が維持されているという流れを示した。また、計算方法としては、障害物の設定において、画像のピクセル情報と領域(Ω)の正方格子とを関連づけることによって、数学解析の仮想領域法と関連付けた数値計算を実現した。これにより、多くの形状領域の障害物に対してでも有限差分法で数値計算することが可能になっている。

今後、この数値計算法を2次元障害物を移動させた場合や、3次元空間の場合にも適用して、より汎用的な計算法を開発する。

参考文献

- [1] 小島 智子, 白石 卓巳, 腰越 秀之, 2次元キャビティ流れの数値計算と可視化-FFT と OpenGL を用いて-, 計算工学講演会論文集, 第11巻, 第2号 (2006), pp.501-502 .
- [2] 河原田 秀夫, 自由境界問題 理論と数値解法, 東京大学出版会, 東京, 1989.
- [3] H. Koshigoe, Direct Method for Solving a Transmission Problem with a Discontinuous Coefficient and the Dirac Distribution, Lecture Notes in Computer Science, 2659, Springer(2003), pp.388-398.
- [4] 桑原 邦郎, 河村 哲也, 流体計算と差分法, 朝倉書店, 東京, 2005.
- [5] J.L.Lions, Perturbations Singulieres dans les Problems aux Limites et en Controle Optimal, Lecture Notes in Mathematics 323, Springer-Verlag, 1973.
- [6] 増田 潤一, 佐伯 公美, 腰越 秀之, 2値画像と差分法による円柱周りのシミュレーショ

- ン, 第 21 回数値流体シンポジウム (日本流体力学会)Proceedings CD(2007).
- [7] 増田 潤一, 佐伯 公美, 松山 孝子, 腰越 秀之, 画像のピクセルと仮想領域を用いた 3 つの円柱周りのシミュレーション, 第 57 回理論応用力学講演会講演論文集 (2008), pp433-434.
- [8] 佐伯 公美, 小島 智子, 腰越 秀之, 地理情報を利用した差分法の数値計算-仮想領域と OpenGL を用いた可視化について-, 日本応用数理学会 2007 年度年会講演予稿集 (2007), pp256-257.
- [9] 高見 穎郎, 河村 哲也, 偏微分方程式の差分解法, 東京大学出版会, 東京, 1994.
- [10] フリーメッシュ法に関する研究 (5 つの円柱周りの非圧縮性流体の解析), http://garlic.q.t.u-tokyo.ac.jp/research/What_is_FMM/fmm.html#s3 (2008.9) .