

Numerical simulations for three-dimensional Navier-Stokes equations using digital color image

Man Zhou* Hideyuki Koshigoe*¹

*Graduate school of Engineering, Chiba University

Abstract In this report we propose a numerical method to analyze incompressible viscous flows modelled by the Navier-Stokes equations around circular cylinders and buildings in Lisbon. The solution method combines finite difference method, digital color image and fictitious domain method. The digital color image consists of so many pixels and each of which forms a small square, is addressable and has the color intensity of RGB. Getting these informations from the digital image, we introduce the notion of cell which is a set of pixels, and approximate the complex shaped domain by these cells. We call it the generation of regular cells. We then construct the finite difference method on cells and show numerical simulations for the Navier-Stokes equations around circular cylinders and buildings in Lisbon. Developing these techniques makes it possible to apply to mathematical and environmental problems in complex shaped domains since the numerical method we shall propose here is based on the digital color image.

¹Corresponding author: H.Koshigoe, E-mail Address: koshigoe@tu.chiba-u.ac.jp

デジタルカラー画像を用いた 3次元 Navier-Stokes 方程式の数値シミュレーション

周 曼* 腰越 秀之*

*千葉大学大学院工学研究科

概要 デジタルカラー画像を用いた円柱周りの3次元 Navier-Stokes 方程式に対する数値計算法, および, リスボン市街地における風の流れの数値シミュレーションを示す. ここでの数値計算法の特徴は有限差分法と近年発達したデジタルカラー画像の結合である. デジタル画像は極微小な正方形のピクセルから構成され, 各ピクセルは赤 (R), 緑 (G), 青 (B) の輝度値, および, アドレスを有している. この点に着目し, 我々は, デジタル画像からこれらの情報を読み取る. それらの情報を用いて, 我々は, ピクセルのいくつかの集合をセルと定義し, セルの輝度値とアドレスという概念を導入する. 故に, 本論文では, 格子点(節点)を用いる従来の数値計算法とは異なり, セル上の有限差分法を提案する. この立場から円柱やリスボン市街地における建物周りの流れの数値シミュレーションを示す.

本論文で提案する数値計算法はデジタルカラー画像を基にしているので, 複雑な形状の領域における偏微分方程式に対しても数値計算ができるし, いろいろな環境数値問題にも広く応用が可能である.

1 はじめに

本研究では、リスボン（ポルトガル）市街地における風の道について考察する．研究の対象はリスボン市街地のバイシャ（地区）と呼ばれ、2つの丘に挟まれている地域である．そこは、1755年のリスボン地震により一度廃墟になったが、再開発され、美しい街になっている．バイシャは、南側のテージョ川から北側に向かって市街地が広がり、そこには、コメルシオ広場、ロシオ広場、フィゲイラ広場という3つの広場がある．図1は Google Earth から得たリスボン市街地のデジタル画像であり、画像の左側がテージョ川である．数値シミュレーションは、デジタル画像と流体の運動方程式 (Navier-Stokes 方程式) を用い、テージョ川から吹く風が市街地の街路でどのように流れていくかを示す．



図 1: Lisbon and the Tejo river (Google Earth)

数値計算は、いままで格子点（節点）という概念を用いていた．例えば、2階導関数を近似するとき、2次元空間の場合は3つの格子点の値を用い、2次元空間ならば、5つの格子点、3次元空間ならば、7つの格子点の値を用いる．そのため、数値計算する前に、計算対象領域に適合した格子点（節点）を生成する．計算対象領域が複雑になることにしたがって、格子点（節点）を生成する時間が要する．本研究では、格子点（節点）を生成するという視点から離れ、デジタルカラー画像のピクセル情報から計算セルを生成（3節）し、数値計算する方法（5節）を示す．その方法は、「デジタル画像は極微小な正方形のピクセルから構成され、各ピクセルは赤 (R), 緑 (G), 青 (B) の輝度値、および、アドレスを持つ」ということを有限差分法に取り込み、仮想領域法を用いて数値計算することである．

本研究では、次の Navier-Stokes 方程式 (1.1) と連続の式 (1.2) を満たす速度ベクトル v と圧力 p を数値計算する．

$$\frac{\partial v}{\partial t} + (v \cdot \nabla)v = -\nabla p + \frac{1}{Re} \Delta v \quad (1.1)$$

$$\nabla \cdot v = 0 \quad (1.2)$$

ここで、 v は、2次元空間の場合に2成分 (u, v) を持ち、3次元空間の場合に3成分 (u, v, w) を持つ． p はスカラーである． Re はレイノルズ数を表す．それ故、Navier-Stokes 方程式の数値計算は、2次元ならば3成分 (u, v, p)、3次元ならば4成分 (u, v, w, p) を各時刻毎に求める問題である．

本研究では、次の (I),(II),(III) の場合の数値シミュレーションを示す．

(I) 円柱周りの 2 次元 Navier-Stokes 方程式

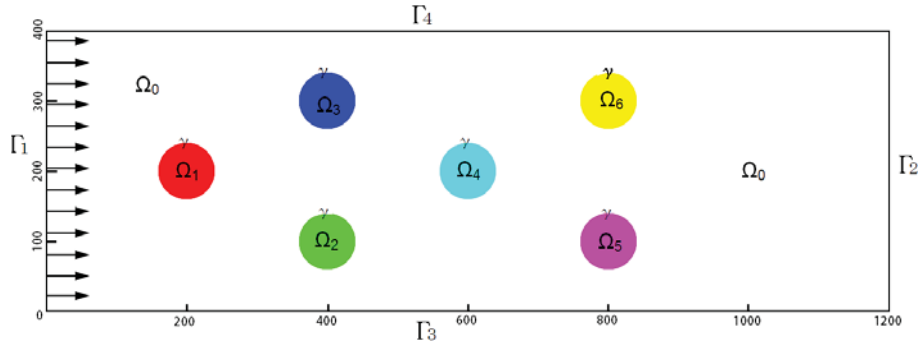


図 2: Digital color image of 6 circular cylinders

図 2 の長方形領域 (Ω) は 480,000 ピクセル (横 1200 ピクセル, 縦 400 ピクセル) のデジタル画像である．カラー画像として 7 色を用いる．白い領域を Ω_0 と表し, 6 つのカラー円柱をそれぞれ $\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6$ と表し, 6 つのカラー円柱の境界を γ と表す．また, 外周の境界は $\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4$ とする．

ここでは, Γ_1 から流れを与えて, 障害物を除いた領域 Ω_0 における Navier-Stokes 方程式

$$\begin{aligned} \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} &= -\nabla p + \frac{1}{Re} \Delta \mathbf{v} && \text{in } (0, T) \times \Omega_0 \\ \nabla \cdot \mathbf{v} &= 0 && \text{in } (0, T) \times \Omega_0 \end{aligned}$$

を次の境界条件 (a-1), (a-2) のもとで数値計算する．

(a-1) 障害物 $\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6$ の側面 γ 上の境界条件：

$$\begin{aligned} \frac{\partial p}{\partial \mathbf{n}} &= 0.0 && \text{on } (0, T) \times \gamma \\ \mathbf{v} &= (0.0, 0.0) && \text{on } (0, T) \times \gamma \end{aligned} \quad (1.3)$$

(a-2) 外周 Γ_i ($i = 1, 2, 3, 4$) 上の境界条件：

$$\begin{aligned} \mathbf{v} &= (0.2, 0.0) && \text{on } (0, T) \times \Gamma_1 \\ \frac{\partial \mathbf{v}}{\partial \mathbf{n}} &= (0.0, 0.0) && \text{on } (0, T) \times (\partial\Omega \setminus \Gamma_1) \\ p &= 0.0 && \text{on } (0, T) \times \Gamma_2 \\ \frac{\partial p}{\partial \mathbf{n}} &= 0.0 && \text{on } (0, T) \times (\partial\Omega \setminus \Gamma_2) \end{aligned}$$

(II) リスボン市街地における 2 次元 Navier-Stokes 方程式



図 3: Digital image of Lisbon city (Google Earth)

図3は Google Earth から取った横 1000 ピクセル，縦 300 ピクセルの計算領域の画像である．数値計算には，図3の長方形領域を Ω で表し， $\Omega = (0, 1000) \times (0, 300)$ ピクセルである．また，道路と広場 (A,B,C) を白い領域 Ω_0 で表し，建物 (障害物) を黒い領域 Ω_2 で表し，テージョ川を青い領域 Ω_3 で表す．また，外周の境界を $\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4$ と表す．

ここでは，テージョ川から吹く風を与え，障害物を除いた領域 ($\Omega \setminus \overline{\Omega_2}$) における Navier-Stokes 方程式

$$\begin{aligned} \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} &= -\nabla p + \frac{1}{Re} \Delta \mathbf{v} && \text{in } (0, T) \times (\Omega \setminus \overline{\Omega_2}) \\ \nabla \cdot \mathbf{v} &= 0 && \text{in } (0, T) \times (\Omega \setminus \overline{\Omega_2}) \end{aligned}$$

を境界条件 (b-1), (b-2) および付加条件 (b-3) のもとで数値計算する．

(b-1) 障害物 (建物) Ω_2 の境界面 γ 上の境界条件：

$$\begin{aligned} \frac{\partial p}{\partial \mathbf{n}} &= 0.0 && \text{on } (0, T) \times \gamma \\ \mathbf{v} &= (0.0, 0.0) && \text{on } (0, T) \times \gamma \end{aligned} \quad (1.4)$$

(b-2) 外周 $\Gamma_i (i = 1, 2, 3, 4)$ 上の境界条件：

$$\begin{aligned} \text{速度 } \mathbf{v}: & \quad \mathbf{v} = (0.2, 0.0) && \text{on } (0, T) \times \Gamma_1 \\ & \quad \frac{\partial \mathbf{v}}{\partial \mathbf{n}} = (0.0, 0.0) && \text{on } (0, T) \times (\partial\Omega \setminus \Gamma_1) \\ \text{圧力 } p: & \quad p = 0.0 && \text{on } (0, T) \times \Gamma_2 \\ & \quad \frac{\partial p}{\partial \mathbf{n}} = 0.0 && \text{on } (0, T) \times (\partial\Omega \setminus \Gamma_2) \end{aligned}$$

(b-3) 障害物 (建物) Ω_2 とテージョ川 Ω_3 における付加条件：

$$\begin{aligned} \mathbf{v} &= (0.0, 0.0) && \text{in } (0, T) \times \Omega_2 \\ \mathbf{v} &= (0.2, 0.0) && \text{in } (0, T) \times \Omega_3 \end{aligned}$$

(III) 円柱周りの 3 次元 Navier-Stokes 方程式

計算領域は，(I) で示した図2における円柱の高さが指定された 3 次元領域図5である．

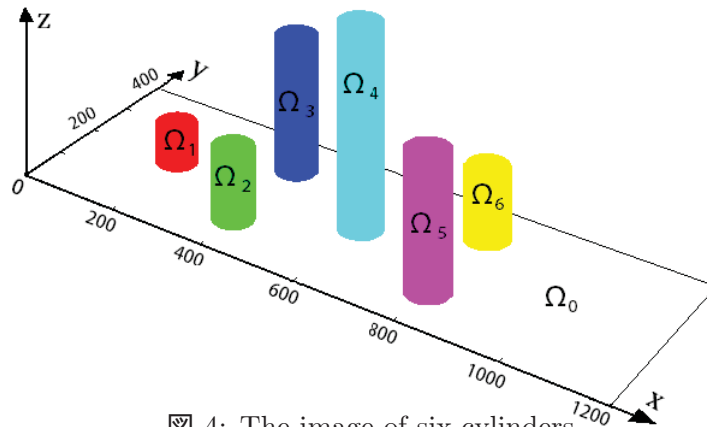


図4: The image of six cylinders

図4の直方体領域を Ω と表し， $\Omega = (0, 1200) \times (0, 400) \times (0, 400)$ ピクセルである． Ω の $x = 0$ ピクセルと $x = 1200$ ピクセルにおける $y - z$ 断面をそれぞれ Γ_1, Γ_2 とする． $y = 0$ ピクセルと $y = 400$ ピクセルにおける $x - z$ 断面をそれぞれ Γ_3, Γ_4 とする． $z = 0$ ピクセルと $z = 400$ ピクセルの $x - y$ 断面をそれぞれ Γ_5, Γ_6 とする．高さが指定された 6 つの円柱を $\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6$ と表し，それぞれの境界面を $\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6$ と表す．また，

$\Omega_0 = \Omega \setminus (\Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4 \cup \Omega_5 \cup \Omega_6)$, $\gamma = \gamma_1 \cup \gamma_2 \cup \gamma_3 \cup \gamma_4 \cup \gamma_5 \cup \gamma_6$ とおく .

ここでは , Γ_1 から流れを与え , 6 つ円柱を除いた 3 次元領域 Ω_0 における Navier-Stokes 方程式

$$\begin{aligned} \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} &= -\nabla p + \frac{1}{Re} \Delta \mathbf{v} && \text{in } (0, T) \times \Omega_0 \\ \nabla \cdot \mathbf{v} &= 0 && \text{in } (0, T) \times \Omega_0 \end{aligned}$$

を次の境界条件 (c-1) , (c-2) のもとで数値計算する .

(c-1) 障害物 $\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6$ の側面 γ 上の境界条件 :

$$\begin{aligned} \frac{\partial p}{\partial \mathbf{n}} &= 0.0 && \text{on } (0, T) \times \gamma \\ \mathbf{v} &= (0.0, 0.0, 0.0) && \text{on } (0, T) \times \gamma \end{aligned} \quad (1.5)$$

(c-2) 外周 Γ_i ($i = 1, 2, 3, 4$) 上の境界条件 :

$$\begin{aligned} \mathbf{v} &= (0.2, 0.0, 0.0) && \text{on } (0, T) \times \Gamma_1 \\ \frac{\partial \mathbf{v}}{\partial \mathbf{n}} &= (0.0, 0.0, 0.0) && \text{on } (0, T) \times (\partial\Omega \setminus \Gamma_1) \\ p &= 0.0 && \text{on } (0, T) \times \Gamma_2 \\ \frac{\partial p}{\partial \mathbf{n}} &= 0.0 && \text{on } (0, T) \times (\partial\Omega \setminus \Gamma_2) \end{aligned}$$

上にあげた (I),(II),(III) の数値計算プログラムは次の点に留意して作成した .

Navier-Stokes 方程式の数値計算では , MAC 法が知られている [8] . 圧力 (p) に関するポアソン方程式の数値解を求める . 計算された圧力 (p) を用い , 速度ベクトル (\mathbf{v}) を陽解法で求める . MAC 法はこの計算を各時間ステップ毎に繰り返していく方法である . しかし , 陽解法は時間の刻み幅の取り方に制限があり , 不便である . 本研究では , MAC 法の陽解法部分を陰解法で定式化して数値計算を行い , その数値解を OpenGL で可視化する .

2 計算セル (regular cell) の生成

2.1 計算セルの生成方法

デジタルカラー画像は障害物領域を画像として自動的に含む。また，デジタルカラー画像がピクセルから構成され，各ピクセルは極微小な正方形の形，赤 (R), 緑 (G), 青 (B) の輝度値，および，アドレスを有している。この情報に着目し，計算対象領域をデジタル画像化することによって，障害物を含む計算領域全体がピクセルで近似する。デジタルカラー画像のピクセル情報を用い，まず，セルを構成し，セルの輝度値を計算する。次に，各セルの輝度値によって，セルはどの部分領域に属するかを判定し，計算セルの領域を近似する。次の 3step で説明する。

Step 1. セルの構成

本論文では，横 N_c ピクセルと縦 N_c ピクセルから構成されている正方形をセルと定義する。

例： $N_c = 5$ の場合，図 6 のように，各セルは横 5 ピクセル縦 5 ピクセル合わせて 25 個ピクセルからなっている。

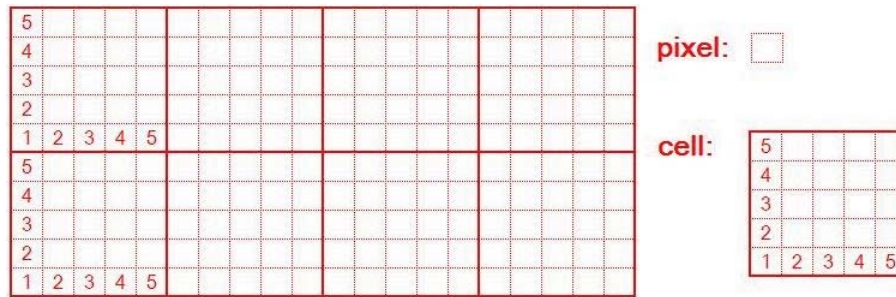


図 5: The relation between pixels and cell

このように，デジタル画像をたくさんのセルに分け，また，ピクセルの輝度値から，各セルの輝度値が計算できる。

Step 2. セルの輝度値

本論文では，セルに含むピクセルの輝度値 (x_R, x_G, x_B) の平均値をセルの輝度値と定める。

各ピクセルは赤 (R), 緑 (G), 青 (B) 三原色の輝度値を持ち，それぞれの輝度値は 8 ビット 256 階調である。その情報を 24bit BMP ファイルの画像から読み取り，セルの輝度値 (y_R, y_G, y_B) が図 6 ように計算する。

以上の例より，

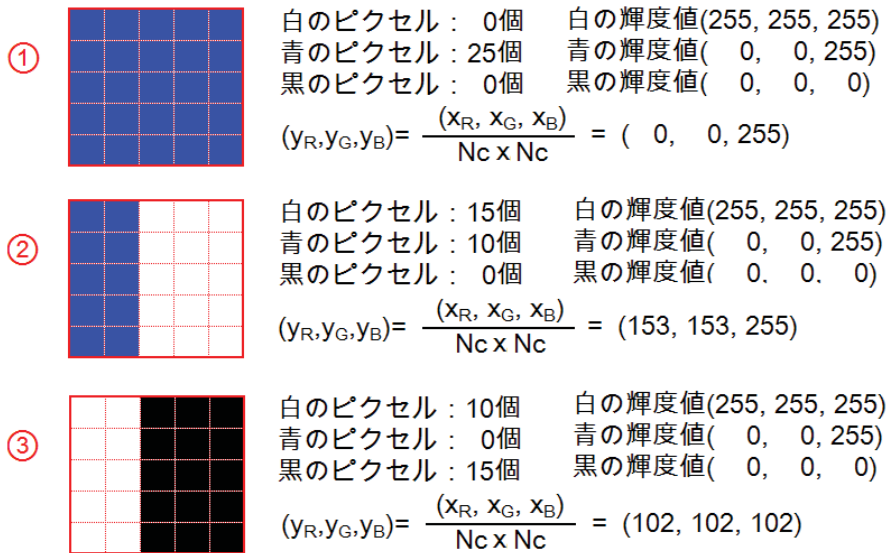
$$\begin{cases} y_R = \frac{x_R}{N_c \times N_c}; & x_R \text{ はセル内に含むピクセルの } R \text{ の輝度値の合計} \\ y_G = \frac{x_G}{N_c \times N_c}; & x_G \text{ はセル内に含むピクセルの } G \text{ の輝度値の合計} \\ y_B = \frac{x_B}{N_c \times N_c}; & x_B \text{ はセル内に含むピクセルの } B \text{ の輝度値の合計} \end{cases}$$

セルの輝度値の計算式は次のようになっている：

$$(y_R, y_G, y_B) = \left(\frac{x_R}{N_c \times N_c}, \frac{x_G}{N_c \times N_c}, \frac{x_B}{N_c \times N_c} \right)$$

Step 3. 計算セルの生成

デジタルカラー画像を対象とした計算領域 Ω が i 色のカラー画像である。領域 $\Omega_k (0 \leq k \leq i)$



の各ピクセルの輝度値は (R_k, G_k, B_k) とする, この時, 計算したセルの輝度値 $(\tilde{R}_k, \tilde{G}_k, \tilde{B}_k)$ により, 計算領域 $\tilde{\Omega}_k$ を近似する. その近似方法の 1 つは次の通りである.

$$\begin{aligned}
 \text{領域 } \tilde{\Omega}_1 &= \{ \text{セルの集合}; (y_R, y_G, y_B) = (\tilde{R}_1, \tilde{G}_1, \tilde{B}_1) \} \\
 \text{領域 } \tilde{\Omega}_2 &= \{ \text{セルの集合}; (y_R, y_G, y_B) = (\tilde{R}_2, \tilde{G}_2, \tilde{B}_2) \} \\
 &\vdots \\
 \text{領域 } \tilde{\Omega}_k &= \{ \text{セルの集合}; (y_R, y_G, y_B) = (\tilde{R}_k, \tilde{G}_k, \tilde{B}_k) \} \\
 \text{領域 } \tilde{\Omega}_0 &= \Omega \setminus (\tilde{\Omega}_1 \cup \tilde{\Omega}_2 \cup \dots \cup \tilde{\Omega}_k)
 \end{aligned}$$

以上は計算領域 Ω に対する計算セルの生成である.

2.2 6つのカラー円柱における計算セルの生成例

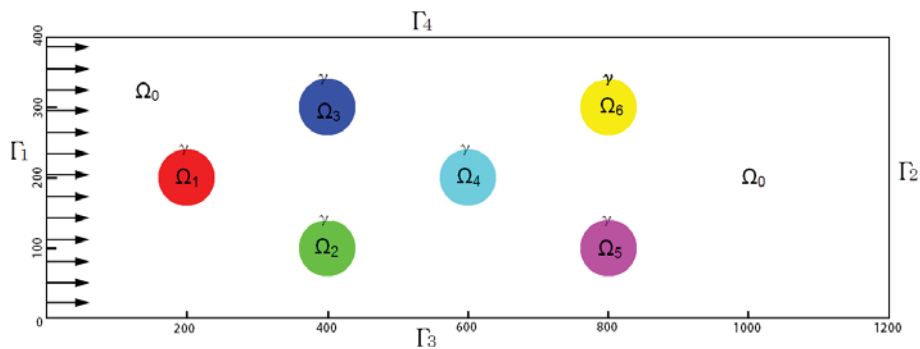


図 6: Digital color image of 6 circular cylinders

上図の長方形領域 (Ω) は 480,000 ピクセル (横 1200 ピクセル, 縦 400 ピクセル) のデジタル画像である. カラー画像として 7 色を用いる. 白い領域を Ω_0 と表し, 6 つのカラー円柱をそれぞれ $\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6$ と表す. 前節で紹介した方法を利用して上図の計算セルを生成する. ここでは, 本研究で開発した数値計算プログラムを用いて, $N_c = 1, 2, 4, 8$ の場合に対する計算セルを生成し, 領域近似の様子を次に示す.

$N_c = 1$ の時 :

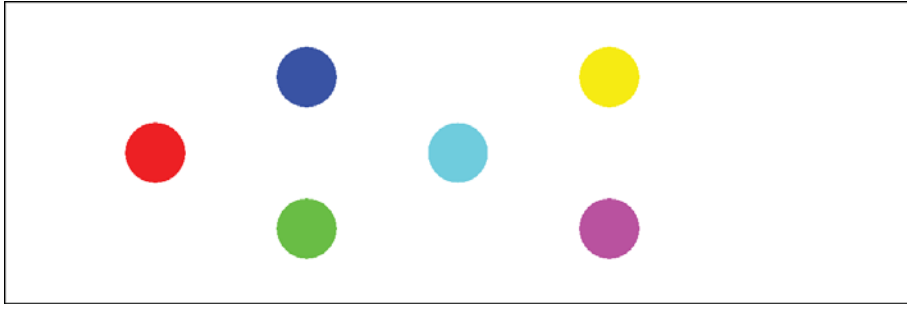
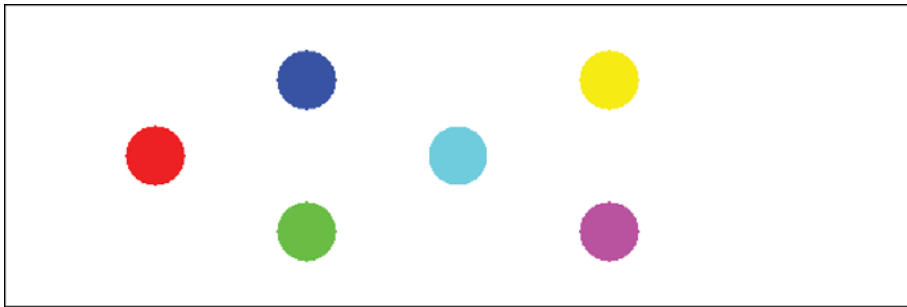
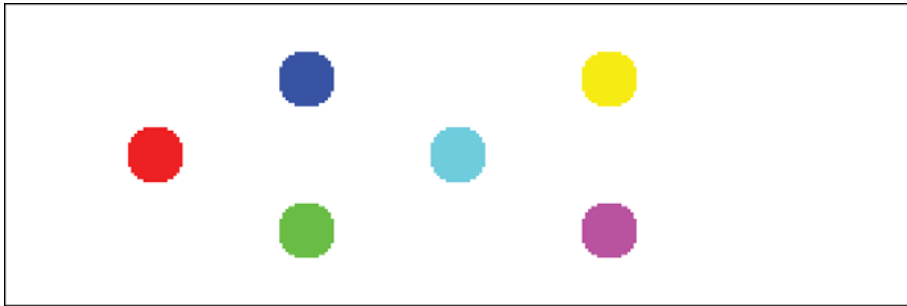


図 7: The generation of regular cells

$N_c = 2$ の時 :



$N_c = 4$ の時 :



$N_c = 8$ の時 :



3 仮想領域法

圧力 (p) に関して障害物 (円柱及び建物) の境界にノイマン境界条件 (1.3,1.4,1.5) が課されている．このノイマン境界条件の解析的近似法として，障害物領域を計算領域内に埋め込み (domain embedding)，障害物領域には特異摂動の方程式とその境界 γ 上には接合境界条件を設定するという方法がある．この方法を流体の圧力ポアソン方程式に適用する．

一般に，補助領域法は最初 Lions[4] によって定式化された．その後，多くの研究者がその理論と数値計算法の研究を行い，仮想領域法，領域分割法として発展してきた．本研究では，文献 [1],[2],[3] を参照して，ノイマン境界条件のもとでの定式化について簡潔に述べる．

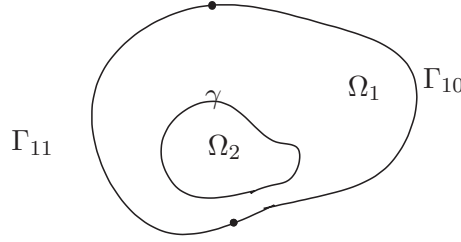


図 8: Fictitious domain

領域 Ω_1 におけるポアソン方程式：

$$\begin{aligned}
 -\Delta u_1 &= f && \text{in } \Omega_1 \\
 \frac{\partial u_1}{\partial \mathbf{n}} &= 0 && \text{on } \gamma \\
 u_1 &= 0 && \text{on } \Gamma_{10} \\
 \frac{\partial u_1}{\partial \mathbf{n}} &= 0 && \text{on } \Gamma_{11}
 \end{aligned} \tag{3.1}$$

に対し， γ 上のノイマン条件 (3.1) を近似する方法として仮想領域法を用いる．その定式化として次の方程式系を設定する．なお，(3.4)-(3.5) が γ 上の接合境界条件という (J.L.Lions[4]) ．

$$-\Delta u_1^\epsilon = f \quad \text{in } \Omega_1 \tag{3.2}$$

$$-\epsilon \Delta u_2^\epsilon = 0 \quad \text{in } \Omega_2 \tag{3.3}$$

$$u_1^\epsilon = u_2^\epsilon \quad \text{on } \gamma \tag{3.4}$$

$$\frac{\partial u_1^\epsilon}{\partial \mathbf{n}} = \epsilon \frac{\partial u_2^\epsilon}{\partial \mathbf{n}} \quad \text{on } \gamma \tag{3.5}$$

$$u_1^\epsilon = 0 \quad \text{on } \Gamma_{10} \tag{3.6}$$

$$\frac{\partial u_1^\epsilon}{\partial \mathbf{n}} = 0 \quad \text{on } \Gamma_{11} \tag{3.7}$$

この時， $\epsilon \rightarrow 0$ 時， $\frac{\partial u_1^\epsilon}{\partial \mathbf{n}} \rightarrow 0$ on γ が成り立つ．この証明は文献 [2],[4] を参照して示せる．

4 Navier-Stokes 方程式の数値シミュレーション

4.1 円柱周りの 2 次元 Navier-Stokes 方程式の数値シミュレーション

2 節の計算セルの生成と 3 節仮想領域法を結合させ，Navier-Stokes 方程式の数値計算を行う．この計算はセル上の有限差分法という新しい定式化した方法である ([5][6][7][9][10]) ．

(i) 圧力 p を計算する

計算セル $\{\tilde{\Omega}_0\}$ を用い

$$\tilde{a}(x, y) = \begin{cases} 1 & \text{in } \tilde{\Omega}_0 \\ \epsilon & \text{in } (\Omega \setminus \tilde{\Omega}_0) \end{cases}$$

を導入し，仮想領域法を用いた圧力ポアソン方程式

$$-\nabla \cdot (\tilde{a}(x, y) \nabla p) = \begin{cases} \frac{\partial(\nabla \cdot \mathbf{v})}{\partial t} + \nabla \cdot ((\mathbf{v} \cdot \nabla) \mathbf{v}) & \text{in } (\tilde{\Omega}_0) \\ 0 & \text{in } (\Omega \setminus \tilde{\Omega}_0) \end{cases}$$

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

を数値計算する．

(ii) 速度ベクトル \mathbf{v} を陰解法，三次精度上流差分近似で計算する

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

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

$$\begin{cases} \mathbf{v} = (0.2, 0.0) & \text{on } (0, T) \times \Gamma_1 \\ \frac{\partial \mathbf{v}}{\partial \mathbf{n}} = (0.0, 0.0) & \text{on } (0, T) \times (\partial\Omega \setminus \Gamma_1) \end{cases}$$

付加条件 (円柱の内部)：

$$\mathbf{v} = (0.0, 0.0) \quad \text{in } (0, T) \times (\tilde{\Omega}_1 \cup \tilde{\Omega}_2 \cup \tilde{\Omega}_3 \cup \tilde{\Omega}_4 \cup \tilde{\Omega}_4 \cup \tilde{\Omega}_5 \cup \tilde{\Omega}_6)$$

を満たす数値解を求める．

数値計算アルゴリズム

1. 全領域 Ω において，圧力ポアソン方程式の圧力 p を数値的に解く．
2. 手順 1 で得られた圧力 p を使い，全領域 Ω における Navier-Stokes 方程式の速度ベクトル \mathbf{v} を数値的に解く．
3. 時間ステップを一つ進め，上の各ステップ 1,2 を繰り返し，圧力 p と速度 \mathbf{v} を順次求めていく．

数値計算結果

画像のピクセル数：480,000(1200 × 400) *pixel*

時間の刻み幅： $\Delta t = 0.01$

$N_c = 4 \text{ pixel}$

レイノルズ数 $Re = 2,000$ ，時間の *step* 数が 2000 ステップの時，流れの数値シミュレーションは下図 9 である．ここで， $maxU = 0.303866$ ， $minU = 0.0$ ．

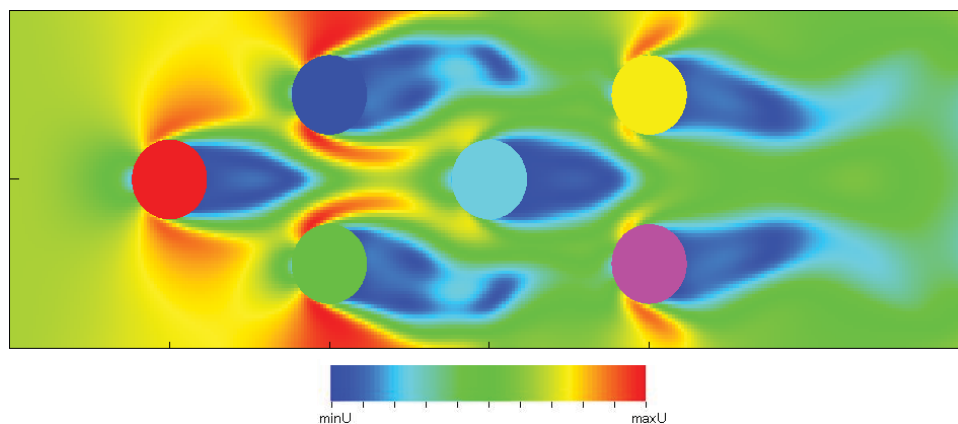


図 9: The numerical simulation

図 9 の色は速度ベクトル (u,v) の大きさに対応した色付けを *OpenGL* で行っている．この流れのベクトル図から見ると，円柱の後方に双子渦とカルマン渦が表れている様子がよくわかる．

4.2 リスボン市街地における風の流れの数値シミュレーション

円柱周りの 2 次元 Navier-Stokes 方程式の数値計算法を用いて，リスボン市街地における風の流れの数値計算を行う．

画像のピクセル数：300,000(1000 × 300) *pixel*

時間の刻み幅： $\Delta t = 0.01$

$N_c = 5 \text{ pixel}$

レイノルズ数 $Re = 2,000$ ，時間の *step* 数が 5000 ステップの時，風の流れの数値シミュレーションは下図 10 である．ここで， $maxU = 0.997398$ ， $minU = 0.0$ ．

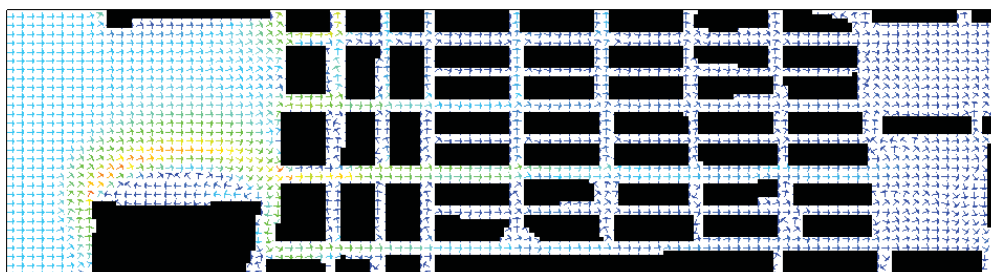


図 10: The numerical simulation

更に，レイノルズ数を大きくとった場合 ($Re = 10,000$) の数値シミュレーションを次に示す．

時間の *step* 数が同じく 5,000 ステップの時，風の流れの数値シミュレーションは下図 11 である．ここで $maxU = 1.09198$, $minU = 0.0$.

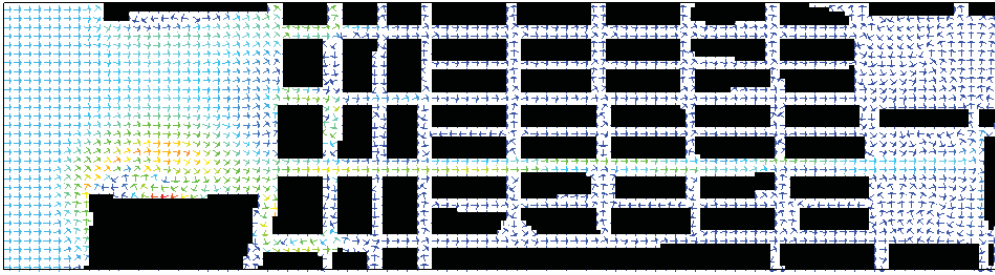


図 11: The numerical simulation

以上，図 10 と図 11 は速度ベクトル (u,v) の向きを矢印で，その大きさを色で表している．図 10 と図 11 ではレイノルズ数の違いより，広場における風の流れの様子が少し異なるが，数値計算による図 10 と図 11 の風の流れは，2つのベクトル図から見ると，テージョ川から吹く風がリスボン市街地の大通りを通して北の広場に抜けていく．また，風が小さい道にも入り，流れていく様子が見える．

4.3 3次元 Navier-Stokes 方程式の数値シミュレーション

3節で紹介した計算セルの生成と4節の仮想領域法を結合させ，3次元の Navier-Stokes 方程式を数値計算する．

画像のピクセル数 ($z = 0$ の $x - y$ 平面) : $480,000(1200 \times 400)pixel$
 レイノルズ数 : $Re = 2,000$
 時間の刻み幅 : $\Delta t = 0.01$
 $N_c = 16pixel$

時間の *step* 数は 5,000 ステップの時，流れの数値シミュレーションは下図 12 である．ここで、 $maxU = 0.275692$, $minU = 0.0$.

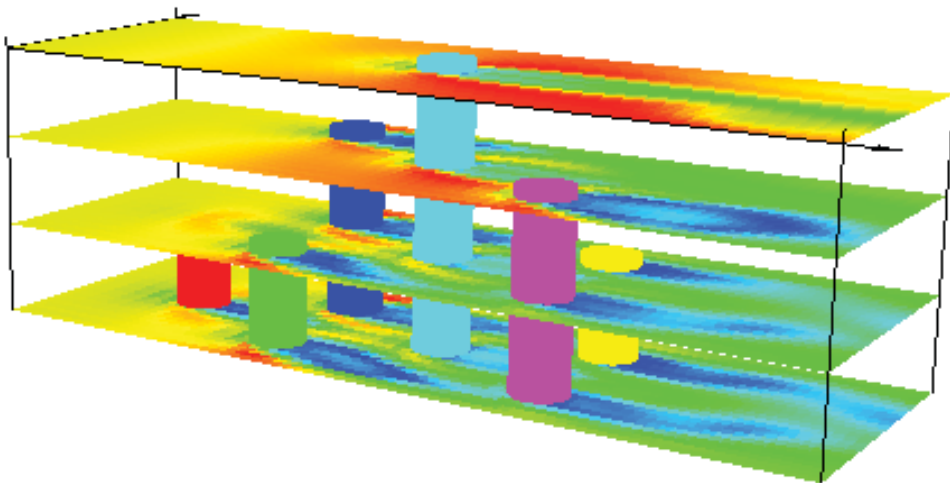


図 12: The numerical simulation around six cylinders

数値計算は 3次元 Navier-Stokes 方程式の数値解 (u,v,w,p) を求め，速度ベクトル (u,v,w) の大きさに対応させて色付けを OpenGL で行っている．ここでは， $z = 0, 128, 256, 384$ ピ

クセルにおける $x - y$ 平面の流れる様子を示している．また，ここで作成した OpenGL のプログラムは， z 軸を中心として図 12 の直方体 (Ω) を回転させることができる．

下図 (i)-(vi) は z 軸を中心に図 12 を反時計周りに回転させた数値シミュレーションである．

(i)

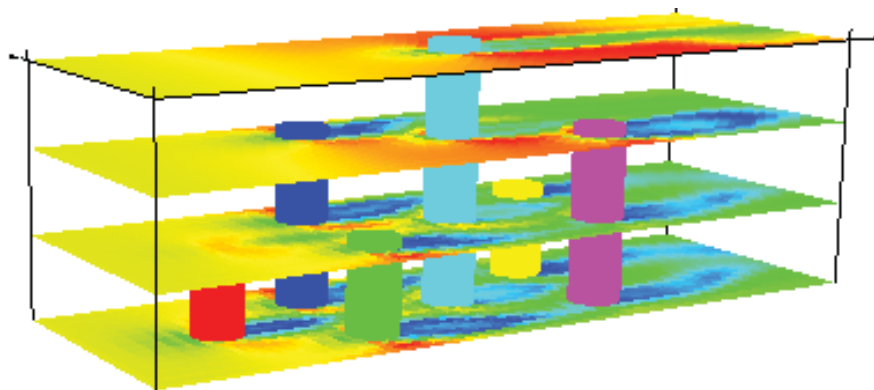


図 13: The numerical simulation around six cylinders

(ii)

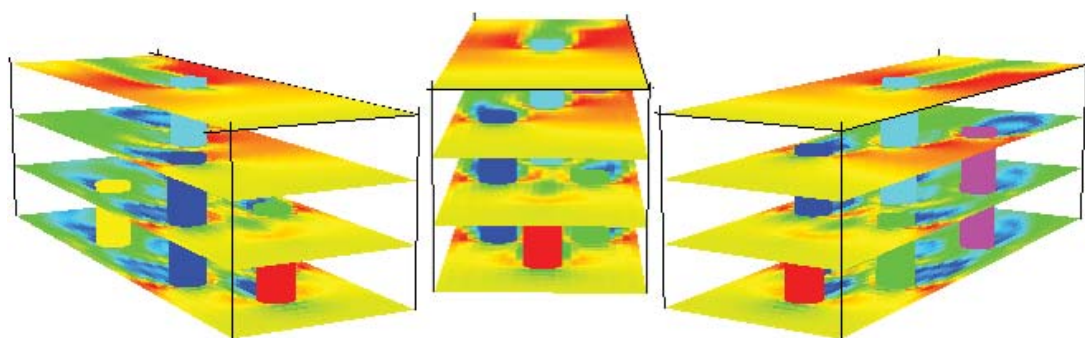


図 14: The numerical simulation around six cylinders

(iii)

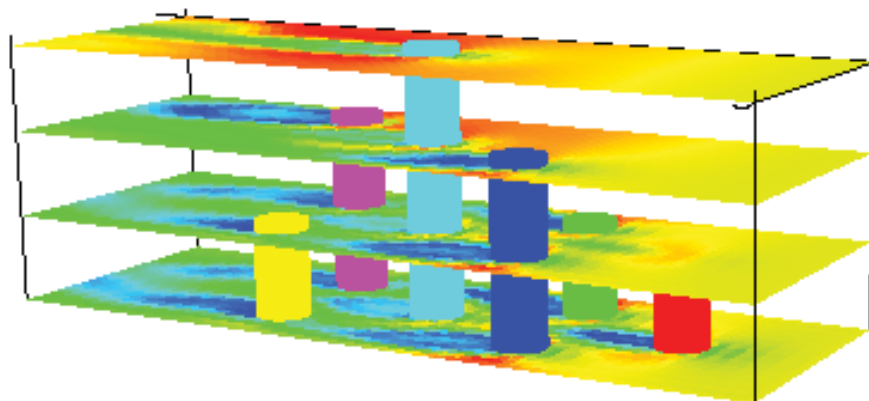


図 15: The numerical simulation around six cylinders

(iv)

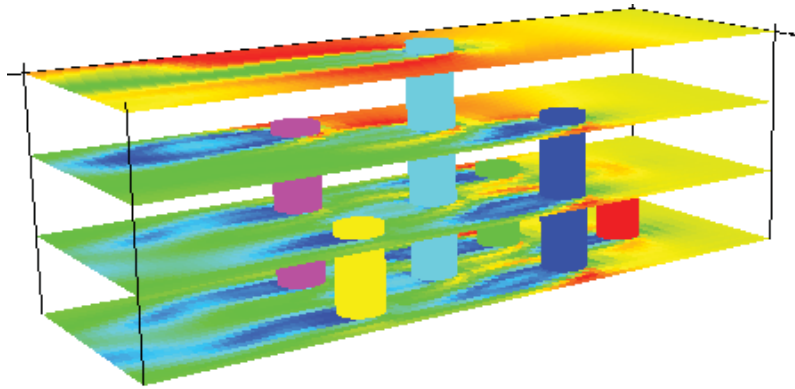


図 16: The numerical simulation around six cylinders

(v)

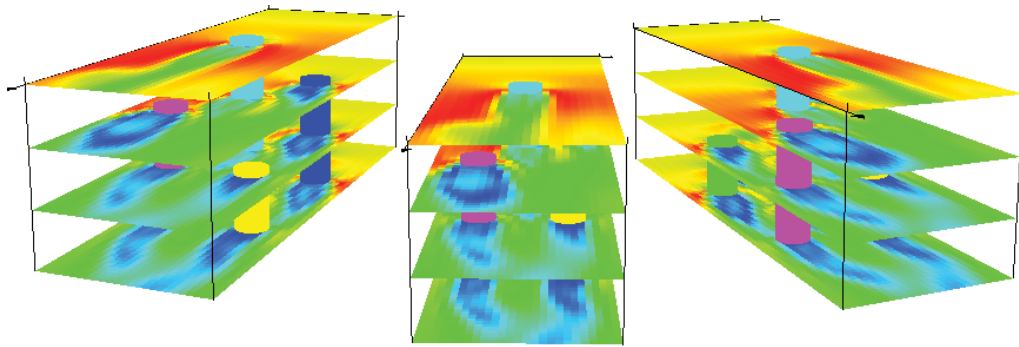


図 17: The numerical simulation around six cylinders

(vi)

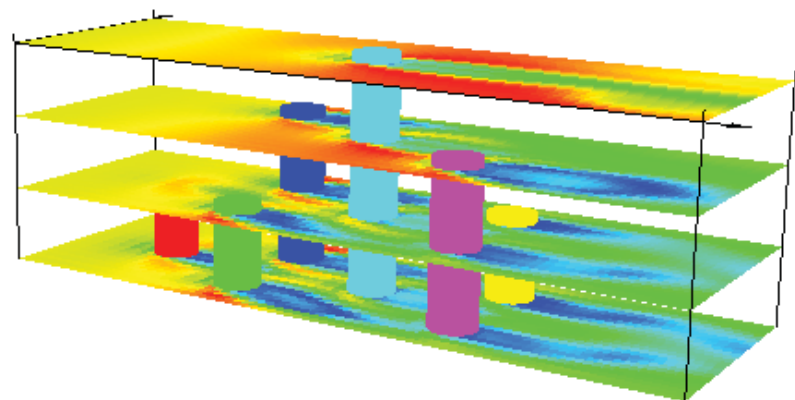


図 18: The numerical simulation around six cylinders

以上の数値シミュレーションは、直方体を回転させることによって、すべての円柱周りの流れが可視化できる。これより、円柱後方に出現する双子渦とカルマン渦を良くとらえていることがわかり、本研究で開発したデジタルカラー画像を用いた数値計算プログラムの有効性を示している。

5 まとめ

- (1) デジタルカラー画像を用いた Navier-Stokes 方程式に対する数値計算法と数値シミュレーションを示した。
- (2) この数値シミュレーションでは、まず、計算対象領域から BMP 画像を作成することから初め、次のプログラムを作成し、数値計算を行った。

24 bit BMP ファイルの画像から計算セルを生成するプログラム。

Navier-Stokes 方程式の計算では、計算セルを用い、LU 分解、3 次精度上流差分、陰解法を基にした有限差分法のプログラム。

2 次元の流れの可視化では、OpenGL を用い^[5]、流れの方向はベクトルで、その大きさはカラーで表示するプログラム。

以上のことは、数値計算にデジタル画像を援用することの有用性を示すとともに、有限差分法とデジタルカラー画像の情報を用いた新しい数値計算法の開発が期待される。

謝辞 本研究の一部は科学研究費補助金（課題番号：22540113）の支援を受けて行われた。

参考文献

- [1] H.Fujita, H.Kawahara and H.Kawarada, Distribution theoretic approach to fictitious domain method for Neumann problems, East-West J.Number Math.3(2)(1995),pp.111-126.
- [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 (1989).
- [3] H.Koshigoe,T.Shiraishi and M.Ehara, Distribution algorithm in finite difference method and its application to a 2D simulation fo temperature inversion, Journal of computational and applied mathematics,vol.231(2009)pp.102-108.
- [4] J.L.Lions,Perturbations Singulieres dansles Problems aux Limites et en Controle Optimal, Lecture Notes in Mathematics 323(1973).
- [5] 増田潤一, 佐伯公美, 吉井孝子, 腰越秀之, 画像のピクセルと仮想領域を用いた3つの円柱周りの数値シミュレーション, 第57回理論応用力学講演会講演論文集, pp.433-434(2008).
- [6] 吉井孝子, 増田潤一, 腰越秀之, デジタルカラーを用いた産業廃棄物中の流れの数値シミュレーション, 第22回数値流体力学シンポジウム, Proceedings CD, K6-5(2008).
- [7] T.Yoshii and H.Koshigoe,A computational method for groundwater flow through industrial waste by use of digital color image,Lecture Notes in Computer Science 6329,Springer(2010), pp.288-296.
- [8] 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 (1994).
- [9] 周曼, 吉井孝子, 腰越秀之, 円柱周りの流れの数値シミュレーションとその応用-デジタルカラー画像のピクセルを用いて-, 日本応用数理学会講演予稿集, pp.329-330(2011).
- [10] 周曼, 吉井孝子, 腰越秀之, デジタルカラー画像を用いた非圧縮性流れに対する仮想領域法, 第25回数値流体力学シンポジウム (日本流体力学学会), Web-Proceedings,C03-1(2011).