

渦法の速度計算における渦度変換手法

Assessment of Vorticity Inversion for Velocity Computation in Vortex Particle Method

- 小林裕, 慶大院, 神奈川県横浜市港北区日吉 3-1-4-1, E-mail: ykobayashiusa@keio.jp
 Matthias Kirchhart, RWTH, Templergraben 55, 52062 Aachen, ドイツ, E-mail: matthias@kirchhart.de
 小尾晋之介, 慶大理工, 神奈川県横浜市港北区日吉 3-1-4-1, E-mail: obsn@mech.keio.ac.jp
 Yutaka Kobayashi, Keio Univ., 3-14-1 Hiyoshi, Kouhoku-ku, Yokohama, Kanagawa, Japan
 Matthias Kirchhart, RWTH., Templergraben 55, 52062 Aachen, German
 Obi Shinnosuke, Keio Univ., 3-14-1 Hiyoshi, Kouhoku-ku, Yokohama, Kanagawa, Japan

There was an accuracy issue regarding particle regularization in vortex computation part of vortex method for the case of a bounded domain. In a previous study the authors developed a particle regularization scheme for a three dimensional bounded domain. In this study we organized a way to convert vorticity field into velocity field along with boundary conditions for the case of a three dimensional bounded domain. In the present study, we formulated a vector Poisson equation and approximated the vector potential using finite element method. We verified our code using smooth vorticity and velocity boundary condition from a homogeneous viscous Channel flow, and obtained a first order convergence result in terms of velocity.

1. 背景

1.1 渦法

渦法とは, 非圧縮 Navier-Stokes 方程式を解く数値計算手法である. 速度と圧力を変数として使用する代わりに渦度 ω に関して記述する渦度輸送方程式 (1) を解く⁽¹⁾.

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla) \omega = \omega \cdot \nabla \mathbf{u} + \nu \nabla^2 \omega \quad (1)$$

ここで, \mathbf{u} は速度, t は時間, ν は流体の動粘性係数である. 格子法とは異なり, 渦法ではこの渦度輸送方程式を Lagrange 的に解く. まず, 渦度場を粒子で式 (2) のように離散化する. そして, 粒子の位置における局所的な速度に応じて粒子の位置を更新することで対流を計算する. ここで, \mathbf{x}_i および Γ_i はそれぞれ粒子 i の位置および循環, δ は Dirac のデルタ関数である.

$$\omega(t, \mathbf{x}) = \sum_{i=1}^N \Gamma_i(t) \delta(\mathbf{x} - \mathbf{x}_i(t)) \quad (2)$$

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{u}(\mathbf{x}_i(t), t) \quad (3)$$

そのため, 対流の計算では数値発散の心配がないことが渦法を使用する利点の一つとして挙げられる.

粘性流体を扱う場合, 粘性拡散の影響を考慮する必要がある. それには, 式 (1) の右辺第二項の粘性拡散項を粘性拡散モデルで近似して求める. また, 三次元流れに関しては, 渦伸長の影響を考慮するために式 (1) の右辺第一項の伸長項の計算も行う.

渦法における粘性拡散や渦伸長の計算には, メッシュの格子の代わりに粒子を使用する. そのため, 基本的に計算領域全体のメッシュを作成する必要がないということも渦法による流体数値解析の利点として挙げられる.

一方, 渦法が抱えている問題の一つに, 渦法を使用した有界領域における流体解析において, 渦度場から速度場を計算する際に行う粒子の正規化がある.

1.2 粒子の正規化

渦法を用いた流体解析の中で渦度場から速度場を求める際, まず初めに粒子の正規化を行う. 非有界領域では分布関数 ζ_σ と離散化された渦度 ω_h との畳み込み (4) によって粒子を正規化する. よく使用されるものとしてガ

ウス分布関数 (5) が挙げられる.

$$\omega_\sigma = \omega_h * \zeta_\sigma \quad (4)$$

$$\zeta_\sigma = \frac{1}{(2\pi\sigma^2)^{3/2}} \exp\left(-\frac{\mathbf{r}_{ij}^2}{2\sigma^2}\right) \quad (5)$$

一方, 有界領域で同様の方法をとると境界の近傍で境界の外側にも渦度場が分布してしまい, 境界がぼやけてしまうので誤差の要因となる.

非有界領域における正規化問題に対し, 様々な正規化手法が提唱されてきた. そのうちの一つに粒子の虚像を境界の外に配置して対処する方法⁽¹⁾があるが, 境界が平坦であることを前提としており, 尖った角等を有する領域での使用は困難である. また, 粒子を格子点としたメッシュを随時作成して区分的な線形近似を行う方法⁽²⁾が Russo and Strain によって提唱されたが, 時間の経過とともに粒子の分布も歪んでいくため結局扱いが困難になる. Vortex-In-Cell (VIC) スキームでは, 格子に基づいた渦度場の近似に積分公式を用いる. しかしながら, 境界付近では積分公式を個々の場合に適合させる必要があるため, 任意の形状で使用することができない⁽³⁾.

近年, 滑らかな一の分割を使用した有限要素法 (Smooth PUFEM) を応用した正規化手法が Kirchhart and Obi⁽⁴⁾で提唱された. これは, Melenk and Babuska⁽⁵⁾による滑らかな一の分割を使用した有限要素法を基に, Friedrichs の軟化子およびデカルト格子を使用して滑らかな空間を得る手法である. ここでは, 任意の形状の有界領域で使用できることが謳われている.

1.3 渦度の変換

正規化を行った後, 非有界領域では Biot-Savart の法則を使用して渦度場から速度場を計算する.

$$\begin{aligned} \mathbf{u} &= \int \mathbf{K}(\mathbf{x} - \mathbf{y}) \omega(\mathbf{y}) d\mathbf{y} \\ &= -\frac{1}{4\pi} \int \frac{\mathbf{x} - \mathbf{y}}{\|\mathbf{x} - \mathbf{y}\|^3} \times \omega(\mathbf{y}) d\mathbf{y} \end{aligned} \quad (6)$$

Biot-Savart の法則 (6) は境界条件を考慮することができない. Kirchhart and Obi⁽⁴⁾では, 境界条件を必要としない渦度場, すなわち境界に近づくにつれて流れが減衰し, 境界上では消滅するような流れ場 (7) を使用した検証

であったため、渦度場を変換する際に境界条件の扱いを心配することなく Biot-Savart の法則を使用した速度場の計算が行われた。

$$\mathbf{u}(x) = \begin{pmatrix} x_2 \\ -x_1 \\ 0 \end{pmatrix} \exp\left(-\frac{1}{1-4|x|^2}\right) \quad (7)$$

一方、管内流や Cavity 内流れをはじめとする有界領域内の流れは多くの場合、滑りなし条件等の境界条件を規定する必要がある。そのような流れの渦度場を使用した有界領域での速度計算にはさらなる工夫が必要となる。

2. 目的

以上を踏まえ、本研究の目的として、渦法を用いた流体解析の中で渦度場を速度場に変換する際の、三次元有界領域での解析を想定した場合の渦度変換手法を整理することとする。その際に用いる渦度場として、解析解を有する三次元有界領域内流れのものを使用し、その解析解を使用して検証を行う。その際、粒子の正規化の必要がない、滑らかな渦度場が得られていることを仮定する。

3. ベクトル Poisson 方程式

三次元有界領域において渦度場から速度場 \mathbf{u} を関係式

$$\boldsymbol{\omega} = \nabla \times \mathbf{u} \quad (8)$$

から計算する際、本研究ではベクトルポテンシャル $\boldsymbol{\psi}$ を使用する。ベクトルポテンシャルに関して成り立つ恒等式 (9)

$$\nabla \times (\nabla \times \boldsymbol{\psi}) = \nabla (\nabla \cdot \boldsymbol{\psi}) - \Delta \boldsymbol{\psi} \quad (9)$$

に、定義式 $\mathbf{u} = \nabla \times \boldsymbol{\psi}$ および発散なし条件 $\nabla \cdot \boldsymbol{\psi} = 0$ を代入することで、ベクトル Poisson 方程式 (10) を得る。これを $\boldsymbol{\psi}$ に関して、境界条件式 (11)、式 (12) とともに解く。

$$-\Delta \boldsymbol{\psi} = \boldsymbol{\omega} \quad (10)$$

$$(\nabla \times \boldsymbol{\psi}) \cdot \mathbf{n} = \mathbf{u} \cdot \mathbf{n} \quad \text{on boundary } \Omega \quad (11)$$

$$\boldsymbol{\psi} \cdot \mathbf{n} = 0 \quad \text{on boundary } \Omega \quad (12)$$

境界条件には、境界表面における速度の \mathbf{u} の法線方向成分もしくは接線方向成分のいずれかを規定する。本研究では、より自然とされる、速度の法線方向成分を規定する。同時に、ベクトルポテンシャル $\boldsymbol{\psi}$ の法線方向成分も規定する。

最後にベクトルポテンシャルの回転をとることで速度場が求まる。

$$\mathbf{u} = \nabla \times \boldsymbol{\psi} \quad (13)$$

$$= \left(\frac{\partial \psi_z}{\partial y} - \frac{\partial \psi_y}{\partial z} \right) \mathbf{e}_x + \left(\frac{\partial \psi_x}{\partial z} - \frac{\partial \psi_z}{\partial x} \right) \mathbf{e}_y + \left(\frac{\partial \psi_y}{\partial x} - \frac{\partial \psi_x}{\partial y} \right) \mathbf{e}_z \quad (14)$$

4. 解析方法

式 (10)、式 (11)、および式 (12) から成る系を解くにあたり、大きく二種類の解析方法が考えられる。一つは、有限要素法を使用して解く方法である。もう一つは、Biot-Savart の法則と境界要素法からなる積分形式で解く方法である。後者の方法では、まず Biot-Savart の法則を使用

して境界条件を考慮せずにベクトル Poisson 方程式 (15) の特解 $\boldsymbol{\psi}_1$ を求める。

$$-\Delta \boldsymbol{\psi}_1 = \boldsymbol{\omega} \quad (15)$$

$$\boldsymbol{\psi}_1 = G \star \boldsymbol{\omega}, \quad G = \frac{1}{4\pi|\mathbf{x}|} \quad (16)$$

次に、境界要素法を使用し、ベクトル Laplace 方程式 (17) の斉次解 $\boldsymbol{\psi}_2$ を、境界条件を考慮した式 (18) および式 (19) から計算する。

$$-\Delta \boldsymbol{\psi}_2 = 0 \quad (17)$$

$$(\nabla \times \boldsymbol{\psi}_2) \cdot \mathbf{n} = \mathbf{u} \cdot \mathbf{n} - (\nabla \times \boldsymbol{\psi}_1) \cdot \mathbf{n} \quad (18)$$

$$\boldsymbol{\psi}_2 \cdot \mathbf{n} = -\boldsymbol{\psi}_1 \cdot \mathbf{n} \quad (19)$$

最後に、一般解 $\boldsymbol{\psi} = \boldsymbol{\psi}_1 + \boldsymbol{\psi}_2$ の回転をとることで速度場を求める。

$$\mathbf{u} = \nabla \times \boldsymbol{\psi} \quad (20)$$

有限要素法では、計算領域全体のメッシュを作成する必要がある。一方、積分形式では、境界要素法の計算の際に境界表面にのみメッシュを作成すればよいので、渦法のメッシュレス性を比較的保持させることができると考えられる。しかし本研究では、実装上の難易度を考慮し、有限要素法を使用した。

5. 有限要素法

5.1 弱形式

有限要素法では、偏微分方程式の階数を一つ落とした弱形式を使って解く⁽⁷⁾。本研究ではベクトルポテンシャルを扱っているため、 $\boldsymbol{\psi}$ および $\boldsymbol{\omega}$ の各成分に関する弱形式 (21) を解く。

$$\begin{aligned} a(\psi_x, v) &= \langle \omega_x, v \rangle \\ a(\psi_y, v) &= \langle \omega_y, v \rangle \\ a(\psi_z, v) &= \langle \omega_z, v \rangle \end{aligned} \quad (21)$$

ここで、

$$\begin{aligned} a(\psi_x, v) &= \int_{\Omega} \nabla \psi_x \cdot \nabla v \, dx \\ \langle \omega_x, v \rangle &= \int_{\Omega} \omega_x v \, dx \end{aligned} \quad (22)$$

であり、他の成分に関しても同様である。

5.2 Galerkin 法

本研究では Galerkin 法を使用する。すなわち、ベクトルポテンシャルおよび渦度の x 成分に関して、弱形式

$$\sum_{i=0}^{N-1} c_i a(\phi_i, \phi_j) = \langle \omega_x, \phi_j \rangle \quad \forall \phi_j, \quad j = 0, \dots, N-1 \quad (23)$$

に基づき、解 ψ_x の近似解 ψ_{xh} を得る。

$$\psi_{xh}(\mathbf{x}) = \sum_{i=0}^{N-1} c_i \phi_i(\mathbf{x}) \quad (24)$$

ここで, c_i は係数, ϕ_i は形状関数である. また, N はメッシュ全体の格子点の数である. c_i を求めるために,

$$\begin{aligned} A &= [a_{ij}] = [a(\phi_i, \phi_j)] \\ b_j &= \langle \omega_x, \phi_j \rangle \end{aligned} \quad (25)$$

とすると, 連立一次方程式を得る.

$$Ac = b \quad (26)$$

ここで,

$$\begin{aligned} a(\phi_i, \phi_j) &= \int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j \, d\mathbf{x} \\ \langle \omega_x, \phi_j \rangle &= \int_{\Omega} \omega_x \phi_j \, d\mathbf{x} \end{aligned} \quad (27)$$

であり, ベクトルポテンシャルおよび渦度の y, z 成分に関しても同様である.

5.3 有限要素による線形近似

弱形式 (21) を有限要素で線形近似し, Galerkin 法を適用したものは式 (28) のようになる. ここで T は要素である.

$$\begin{aligned} a(\phi_i, \phi_j) &= \sum \int_T \nabla \phi_i \cdot \nabla \phi_j \, d\mathbf{x} \\ \langle \omega_x, \phi_j \rangle &= \sum \int_T \omega_x \phi_j \, d\mathbf{x} \end{aligned} \quad (28)$$

また, 要素ごとの近似解は, 式 (29) のようになる.

$$\psi_{xh,T}(\mathbf{x}) = \sum_{i=0}^{n-1} c_i \phi_i(\mathbf{x}) \quad (29)$$

有限要素で線形近似した際, 要素ごとの要素行列や要素ベクトルの元の数は, 使用した要素の種類および近似の次数に依存する. 要素に四面体要素を使用する際, 近似の次数が一次の場合は $n=4$, 二次の場合は $n=10$ となる. それら要素行列および要素ベクトルをそれぞれ剛性マトリクスおよびシステムベクトルへとアセンブリし, 最終的に連立一次方程式 (26) を得る. ここで, メッシュ内のすべての格子点には番号が振られており, どの格子点がどの要素に属しているのかということが, 格子点の番号で管理される. 要素行列や要素ベクトルの元は要素の格子点に対応しており, アセンブリを行う際, 要素を構成する格子点の番号に基づいて剛性マトリクスおよびシステムベクトルの各元へと足し合わされる. y, z 成分に関しても同様である. ただし, 表面 Poisson 方程式の解の回転を境界条件として剛性マトリクスおよびシステム行列へ適用させる際に各成分をひとまとまりとして扱う必要があるため, 要素行列および要素ベクトルは最終的には, ベクトルポテンシャルの成分が連続して並ぶように拡張された $3N \times 3N$ 元の行列および $3N$ 元ベクトルとなる.

本研究ではアイソパラメトリック要素を使用した. すなわち, 自然座標系 ξ, η, ζ であらかじめ定義した形状関数を使用して計算を行い, その結果を各要素ごとに物理座標系 x, y, z へと写像する. 自然座標系におけるベクトルを ξ , 物理座標系におけるベクトルを \mathbf{x} , ξ から \mathbf{x} への写像を $\mathbf{x} = \chi(\xi)$ とすると, 式 (28) の一要素における積分は, 自然座標系における形状関数 $\hat{\phi}$ を使用して最終的に式 (30), 式 (31) のようになる.

$$\int_T \nabla \phi_i \cdot \nabla \phi_j \, d\mathbf{x} = \int_T \frac{\partial \hat{\phi}_i(\xi)}{\partial \xi} \cdot \left(\frac{\partial \chi(\xi)}{\partial \xi} \right)^{-1} \cdot \frac{\partial \hat{\phi}_j(\xi)}{\partial \xi} \cdot \left(\frac{\partial \chi(\xi)}{\partial \xi} \right)^{-1} \cdot \left| \det \frac{\partial \chi(\xi)}{\partial \xi} \right| \, d\xi \quad (30)$$

$$\int_T \omega_x \phi_j \, d\mathbf{x} = \int_T \omega_x(\chi(\xi)) \hat{\phi}_j(\xi) \cdot \left| \det \frac{\partial \chi(\xi)}{\partial \xi} \right| \, d\xi \quad (31)$$

自然座標系が直交座標系の場合, 四面体の格子点は, 重心座標系 $(1-\xi-\eta-\zeta, \xi, \eta, \zeta)$ を使用して簡潔に指定することができる. 自然座標系において, 四面体の格子点の一つが原点, 他の三つが原点から 1 単位長さ離れた各軸上に配置されるような四面体を定義する. そのときの任意の格子点は, 重心座標系の成分のいずれか一つを 1 とすることで指定する.

積分は, 四面体に関する求積法を使用して計算する. ここで, w_k は分点 k の重み, x_k は分点 k の座標である.

$$\int_T \nabla \phi_i \cdot \nabla \phi_j \, d\mathbf{x} = \sum_{k=0}^{N-1} w_k \left(\nabla \phi_i \cdot \nabla \phi_j \right) \quad (32)$$

$$\int_T \omega_x \phi_j \, d\mathbf{x} = \sum_{k=0}^{N-1} w_k \left(\omega_x(x_k) \phi_j(x_k) \right) \quad (33)$$

写像 $\mathbf{x} = \chi(\xi)$ は, 物理座標系における四面体の格子点の座標 \mathbf{x}_i および自然座標系における形状関数 $\hat{\phi}_i(\xi)$ を使用して, 式 (34) のようになる. ここで, n は形状関数の数である.

$$\mathbf{x} = \sum_i^n \hat{\phi}_i(\xi) \mathbf{x}_i \quad (34)$$

ここで, 一次近似の場合は $n=4$, 二次近似の場合は $n=10$ であり, それぞれの場合の形状関数は一次関数 (35), 二次関数を含む関数 (36) となる.

$$\begin{aligned} \hat{\phi}_1 &= 1 - \xi - \eta - \zeta \\ \hat{\phi}_2 &= \xi \\ \hat{\phi}_3 &= \eta \\ \hat{\phi}_4 &= \zeta \end{aligned} \quad (35)$$

$$\begin{aligned} \hat{\phi}_1 &= (1 - \xi - \eta - \zeta) \{ 2(1 - \xi - \eta - \zeta) - 1 \} \\ \hat{\phi}_2 &= \xi (2\xi - 1) \\ \hat{\phi}_3 &= \eta (2\eta - 1) \\ \hat{\phi}_4 &= \zeta (2\zeta - 1) \\ \hat{\phi}_5 &= 4(1 - \xi - \eta - \zeta)\xi \\ \hat{\phi}_6 &= 4\xi\eta \\ \hat{\phi}_7 &= 4\eta(1 - \xi - \eta - \zeta) \\ \hat{\phi}_8 &= 4(1 - \xi - \eta - \zeta)\zeta \\ \hat{\phi}_9 &= 4\xi\zeta \\ \hat{\phi}_{10} &= 4\eta\zeta \end{aligned} \quad (36)$$

6. 境界条件

ベクトルポテンシャルは境界表面で以下の条件を満たすことが要求される.

$$(\nabla \times \boldsymbol{\psi}) \cdot \mathbf{n} = \mathbf{u} \cdot \mathbf{n} \quad \text{on boundary } \Omega \quad (37)$$

$$\boldsymbol{\psi} \cdot \mathbf{n} = 0 \quad \text{on boundary } \Omega \quad (38)$$

ベクトルポテンシャルの境界表面における値を剛性マトリクスに反映させるために, 以上の条件を満たすベクトルポテンシャルを境界条件として求める必要がある. ベクトルポテンシャルの, 境界表面に対して水平な成分は, 式 (37) から導出される, 表面 Poisson 方程式 (39) を μ に関して解き, 式 (40) でベクトルポテンシャルに変換することで具体的に求まる (6). ここで, Δ_s は Laplace-Beltrami 作用素である.

$$-\Delta_s \mu = \mathbf{u} \cdot \mathbf{n}, \quad \mathbf{B} = \mu \mathbf{n} \quad (39)$$

$$\boldsymbol{\psi}_s = \nabla_s \times \mathbf{B} \quad (40)$$

速度の値を式 (39) の右辺のように直接与えるので, 有限要素法では Dirichlet 条件として境界条件を適用する.

7. Nitsche 法

Nitsche 法 (9) とは, Dirichlet 境界条件の処理方法の一つである. 式 (40) で求めた境界条件 $\boldsymbol{\psi}_s$ の値を用い, 弱形式 (21) の x 成分に対して, 境界積分項を式 (41) のように加える. ここで, η は任意に与える 0 より十分に大きな値であり, 安定性に関わる.

$$\begin{aligned} a(\boldsymbol{\psi}_x, v) &= \int_{\partial\Omega} \partial_\nu \boldsymbol{\psi}_x v \, ds - \int_{\partial\Omega} \boldsymbol{\psi}_x \partial_\nu v \, ds + \eta \int_{\partial\Omega} \boldsymbol{\psi}_x v \, ds \\ &= \int_{\partial\Omega} \boldsymbol{\psi}_s \partial_\nu v \, ds + \eta \int_{\partial\Omega} \boldsymbol{\psi}_s v \, ds + \langle \boldsymbol{\omega}_x, v \rangle \end{aligned} \quad (41)$$

8. 解析の流れ

有限要素法によるベクトル Poisson 方程式の解析は, 図 1 ような流れになる. ここで, メッシュの生成以外はプログラムとして C++ で実装した. メッシュの生成には, 自動メッシュ生成ソフト gmsh を使用した.

最初に, メッシュファイルから幾何情報を読み込んで要素同士の関係を構築する. 本解析ではマルチグリッド法を使用するため, マルチレベル細分化アルゴリズム (8) によってメッシュの細分化を行う.

境界条件の計算では, まず表面 Poisson 方程式を解く際に使用する速度境界条件を与え, 表面 Poisson 方程式に基づいて μ を計算し, 最後にその回転をとることでベクトルポテンシャルとしての境界条件 $\boldsymbol{\psi}_s$ を得る.

ベクトルポテンシャルの計算では, まずベクトル Poisson 方程式を解く際に使用する渦度場を与え, 要素ベクトルおよび要素行列を要素ごとに構成し, それらのアセンブリを行う. そのように作成したシステム行列およびベクトルに対し, Nitsche 法で先ほどの Dirichlet 境界条件を適用し, マルチグリッド法を使用して線形行列式を解く. 最後に形状関数を用いた解の線形近似を要素ごとに行うことで最終的にベクトルポテンシャル場を得る. その後, 可視化や誤差評価を行う際に各格子におけるベクトルポテンシャルの回転をとって速度場を得る. 速度場の可視化には可視化ソフト ParaView を使用した.

9. マルチグリッド法とメッシュについて

本研究ではマルチグリッド法 (8) を使用する. マルチグリッド法は, 2 段グリッド法を再帰的に行って方程式の近似解を得る計算方法である. 2 段グリッド法では, 粗い格

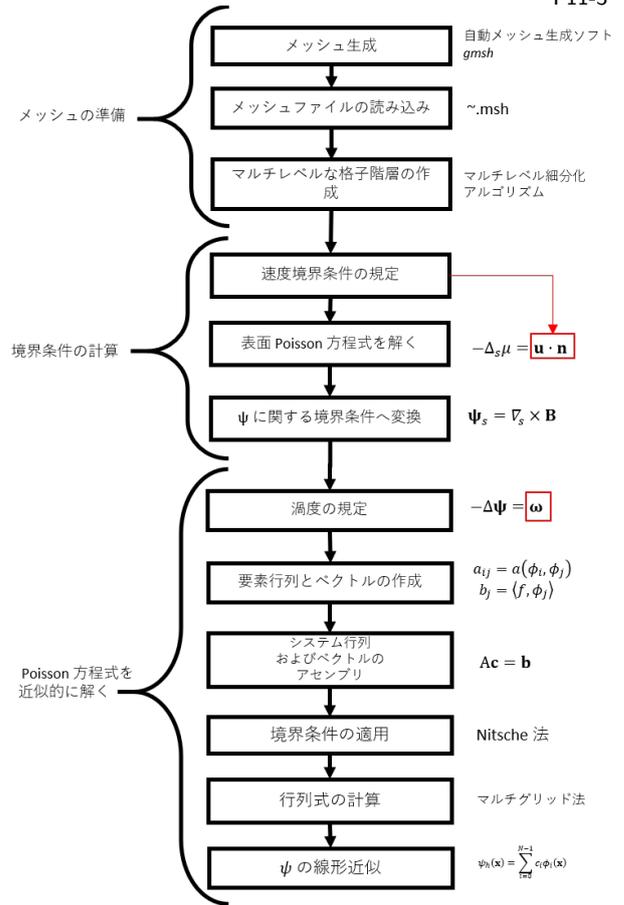


Fig. 1: 解析の大まかな流れ

子から得られる方程式を利用して近似解を効率的に改善する. 手法の主な流れとして, まず細かい格子上で方程式 (42) の近似解 \mathbf{c}_l を反復法で計算する. そこで発生する残差の制限 (restriction) \mathbf{r}_l を計算し, 粗い格子へと変換する (43). 粗い格子上で方程式 (44) を解く. その解の延長 (prolongation) \mathbf{p}_l を計算して細かい格子へと変換したものを使用し, 近似解を改善する (45).

$$A_l \mathbf{c}_l = \mathbf{b}_l \quad (42)$$

$$\mathbf{d}_{l-1} = \mathbf{r}_l (\mathbf{b}_l - A_l \mathbf{c}_l) \quad (43)$$

$$\mathbf{e}_{l-1} = A_{l-1}^{-1} \mathbf{d}_{l-1} \quad (44)$$

$$\mathbf{c}_l = \mathbf{c}_l + \mathbf{p}_l \mathbf{e}_{l-1} \quad (45)$$

細かい格子で式 (42) の近似解を計算するための反復法として, Gauss-Seidel 法を使用した.

マルチグリッド法で使用するメッシュには, 最も粗いメッシュから最も細分化されたメッシュまでを系統的に扱うためのマルチレベル格子階層を使用する. それを作成するにあたり, メッシュ要素の種類が正規か非正規かによって細分化の方法が異なるマルチレベル細分化アルゴリズム (8) を使用する.

メッシュ要素を扱う際, メッシュの細分化レベル, 要素の次元, 要素の位置, の三つの情報を指定する. プログラム内ではこれらの情報をひとまとめにし, 幾何学的 ID と称される構造体として使用する.

10. 解析条件

手法の検証を行うにあたり, 厳密解 \mathbf{u} を有する流れの渦度場 $\boldsymbol{\omega}$ および速度境界条件として, x および z 方向に

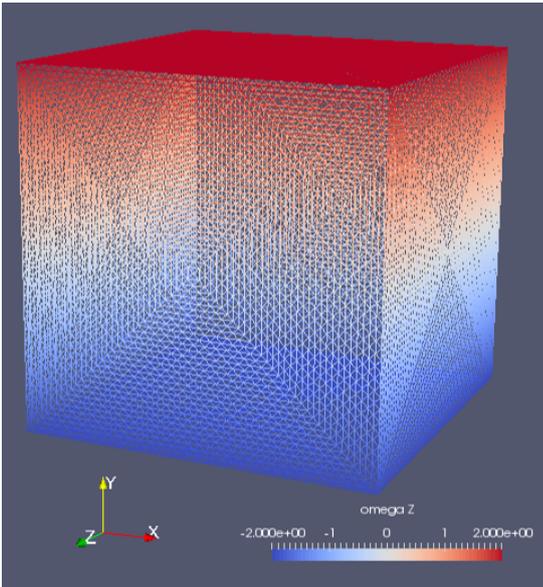


Fig. 2: メッシュの表面と渦度場の様子

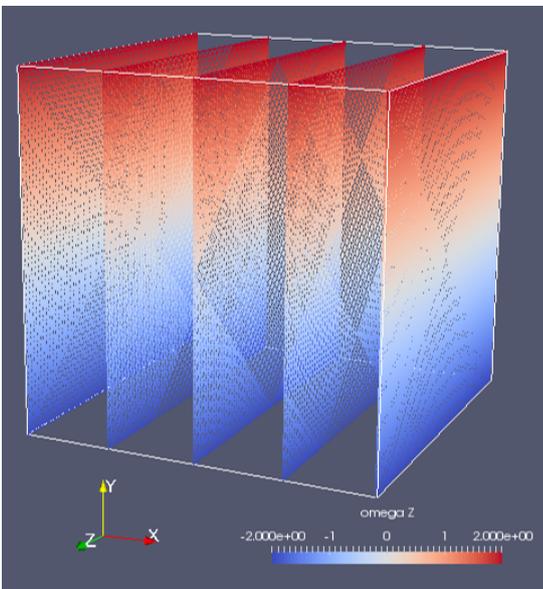


Fig. 3: メッシュの断面と渦度場の様子

一様な粘性 Channel 流のものを使用した.

$$\mathbf{u} = (1 - y^2) \mathbf{e}_x \quad (46)$$

$$\boldsymbol{\omega} = 2y \mathbf{e}_z \quad (47)$$

正六面体の計算領域を定義した. 速度境界条件は式 (39) より $\mathbf{u} \cdot \mathbf{n}$ であるため, x 軸に垂直な断面上では $\mathbf{u} \cdot \mathbf{n} = \pm 1 - y^2$, y 軸, z 軸に垂直な断面上では $\mathbf{u} \cdot \mathbf{n} = 0$ となる. 四つの格子点で構成される一次要素の四面体を用いて計算領域を分割したメッシュを作成した. メッシュはマルチレベル細分化アルゴリズムで第五段階まで細分化したものをを使用した. このときの要素の数は 786432 個, 格子点の数は 1073345 となった. 第五段階の細分化レベルでのメッシュの表面と渦度場の様子は図 2, メッシュの断面と渦度場の要素は図 3 のようになった.

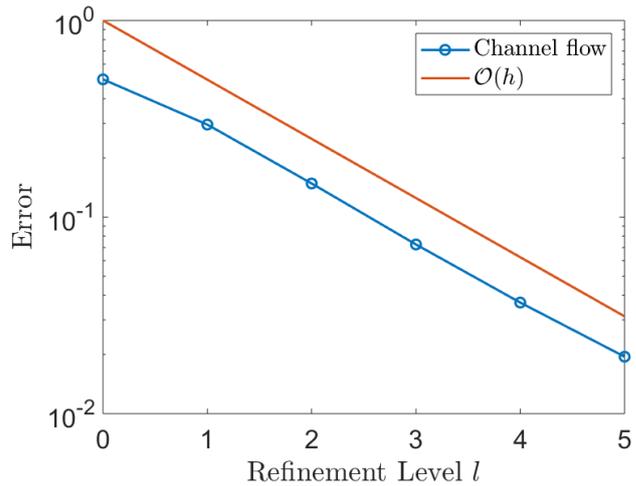


Fig. 4: 誤差のメッシュ精度依存性 一次精度

10.1 解析結果

10.2 誤差のメッシュ精度依存性

まず, 領域全体において, 厳密解に対する計算結果の誤差式 (48) を求めてメッシュ解像度の依存性を評価した. ここで \mathbf{u}_h は解析の結果得た近似解, \mathbf{u} は厳密解である.

$$\text{Error} = \|\mathbf{u}_h - \mathbf{u}\|_{L^2(\Omega)} \quad (48)$$

その結果, 一次精度が確認できた. この理由として, 境界において μ の関数として二次関数を使用していたことが挙げられる. このため μ は最大でも三次精度となるが, 式 (40), 式 (14) からわかるように μ の二回微分が速度なので, 結局速度は一次精度となる.

10.3 速度分布

次に, 解析によって得られたベクトルポテンシャル場と速度場を可視化し, それぞれ図 5, 図 6 のようになった. 特に速度場に関して, 計算領域の各角に特異なベクトルの存在を確認した.

また, 速度の x 成分に関して, y 軸に平行な直線に沿った分布を異なる位置 $(x, z) = (0, 0), (0, 1), (-1, 0), (-1, 1)$ で確認した. 角を含む位置 $(x, z) = (-1, 1)$ のものが厳密解から大幅に外れていることが図 10 からわかり, 角における誤差の発生を確認した. また, メッシュの精度を高めると, この誤差は増大していった.

これの主な要因として, 図 11, 図 13 からわかるように, メッシュの細分化に伴い角における ψ の勾配が急になるためだと考えられる. そのために速度を構成する $\frac{\partial \psi_x}{\partial y}$ および $-\frac{\partial \psi_y}{\partial z}$ の値も減少していく様子が図 12, 図 14 から確認できた. ただ, なぜそのようになるのかはまだ調査中である.

11. 結言と今後の展望

本研究では, 三次元有界領域の渦法解析を想定した, 渦度場を使用した速度場の計算において, ベクトルポテンシャルを解とするベクトル Poisson 方程式を境界条件と併せて解く有限要素法のアプローチを整理し, 検証を行った. 一様な粘性 Channel 流れの渦度および速度境界条件を使用した検証では, メッシュ精度依存性に関しては一次精度を確認した. しかしながら, 正六面体のメッシュを使用したところ, 角を含む直線に沿った速度分布に比較的大きな誤差を確認し, メッシュの細分化レベルの上昇とともにその角の箇所だけ誤差が増大していく様子を確認した.

本研究では, 正規化の必要がない, 滑らかな渦度場がすでに得られていることを想定していた. そのため, 三次元有界領域における粒子の正規化手法 Smooth PUFEM と

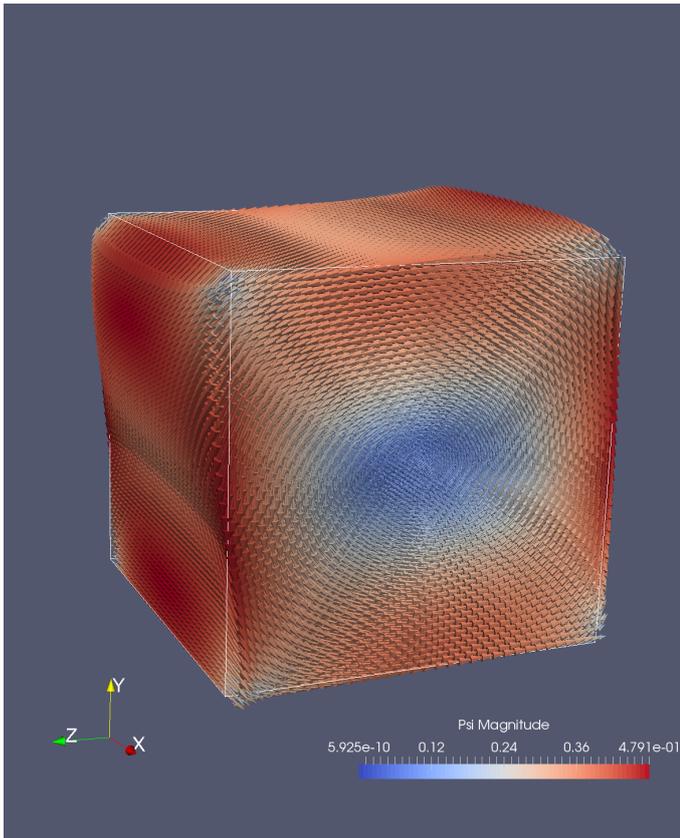


Fig. 5: ベクトルポテンシャル場 ψ

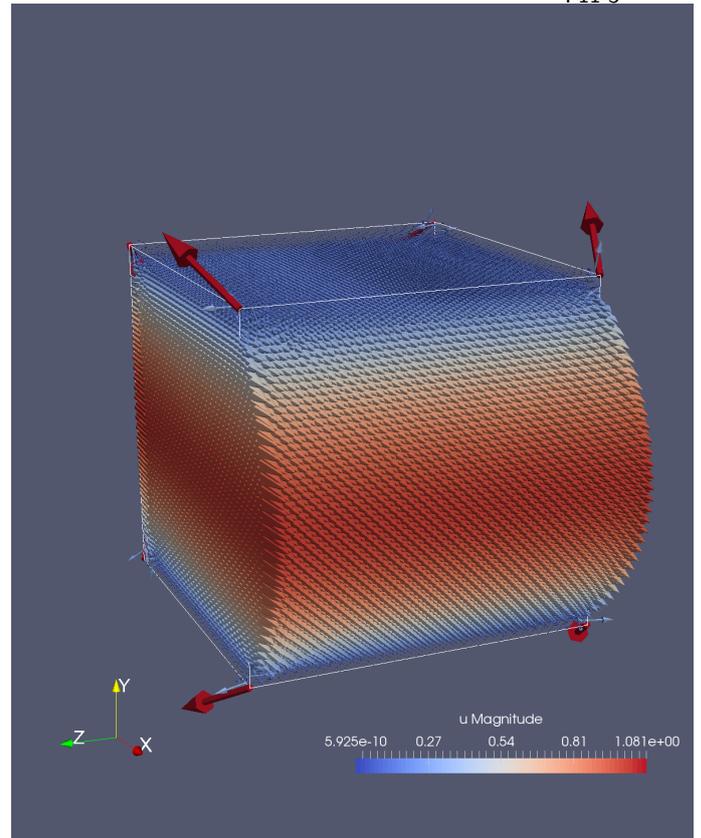


Fig. 6: 速度場 $\mathbf{u} = \nabla \times \psi$

併せることで、実用的な渦度変換手法として完結させることが今後の課題の一つとして挙げられる。

また、本研究ではベクトル Poisson 方程式の解法として有限要素法を使用した。しかし、有限要素法では計算領域全体のメッシュを作成する必要がある、渦法のメッシュレス性を保持することができない。一方、Biot-Savart の法則と境界要素法からなる積分形式を使用した場合、境界要素法の計算の際に境界表面のみメッシュを作成すればよいので、渦法のメッシュレス性を比較的保持させることができると考えられる。しかし、有限要素法と比べて実装面での複雑さは大幅に増す。特異積分の計算⁽¹⁰⁾が必要になるが、そのアルゴリズムが長い。また、実用レベルにまで高速化させるために高速多重極展開法 (FMM) と組み合わせる必要がある。

参考文献

- (1) Cottet, G. H., & Koumoutsakos, P. D. Vortex methods: theory and practice. Cambridge university press, 2000.
- (2) Russo, G., & Strain, J. A. Fast triangulated vortex methods for the 2D Euler equations. Journal of Computational Physics, 111(2), 1994, 291-323.
- (3) Cottet, G. H., & Poncet, P. Advances in direct numerical simulations of 3D wall-bounded flows by Vortex-in-Cell methods. Journal of Computational Physics, 193(1), 2004, 136-158.
- (4) Kirchhart, M., & Obi, S. A Smooth Partition of Unity Finite Element Method for Vortex Particle Regularization. SIAM Journal on Scientific Computing, 39(5), 2017, A2345-A2364.
- (5) Melenk, J. M., & Babuka, I. The partition of unity finite element method: basic theory and applica-

tions. Computer methods in applied mechanics and engineering, 139(1-4), 1996, 289-314.

- (6) Hirasaki, G. J., & Hellums, J. D. A general formulation of the boundary conditions on the vector potential in three-dimensional hydrodynamics. Quarterly of Applied Mathematics, 26(3), 1968, 331-342.
- (7) Bang, H., & Kwon, Y. W. The finite element method using MATLAB. CRC press, 2000.
- (8) Gross, S., & Reusken, A. Numerical methods for two-phase incompressible flows (Vol. 40). Springer Science & Business Media, 2011
- (9) Nitsche, J. ber ein Variationsprinzip zur Lsung von Dirichlet-Problemen bei Verwendung von Teilrumen, die keinen Randbedingungen unterworfen sind. In Abhandlungen aus dem mathematischen Seminar der Universitt Hamburg (Vol. 36, No. 1, pp. 9-15). Springer-Verlag. (1971, July).
- (10) Sauter, S. A., & Schwab, C. Boundary element methods. In Boundary Element Methods (pp. 183-287). Springer, Berlin, Heidelberg, 2010

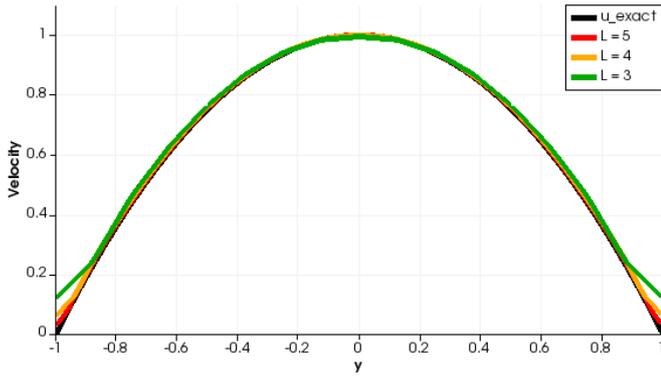


Fig. 7: $(x, z) = (0, 0)$ における速度分布

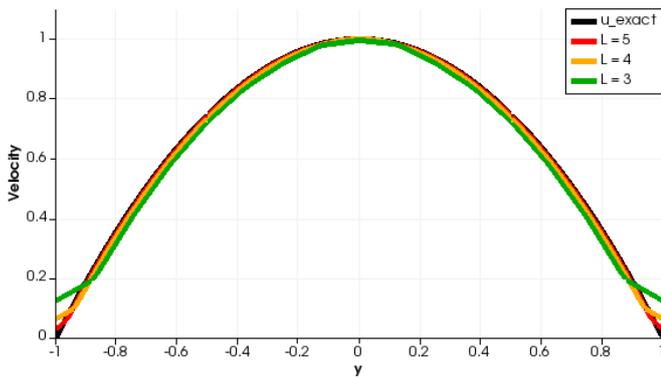


Fig. 8: $(x, z) = (0, 1)$ における速度分布

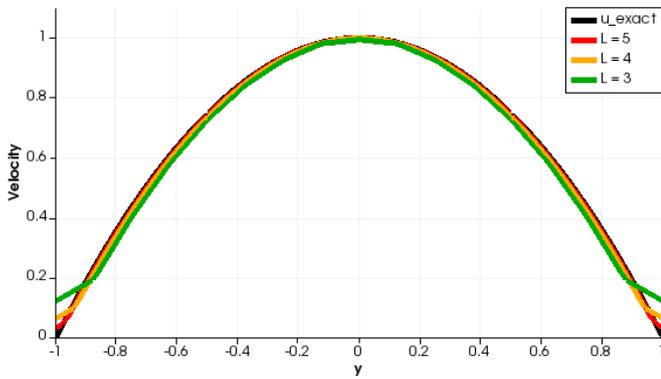


Fig. 9: $(x, z) = (-1, 0)$ における速度分布

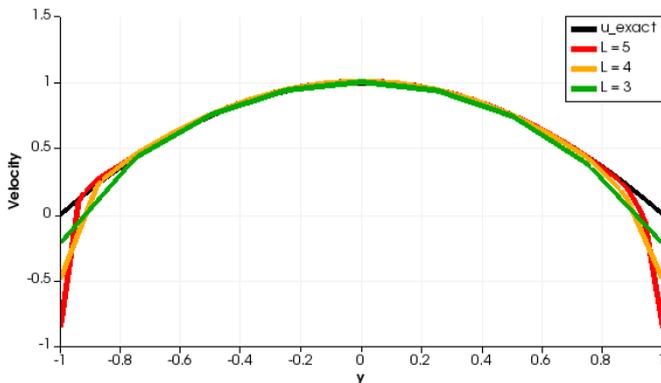


Fig. 10: $(x, z) = (-1, 1)$ における速度分布

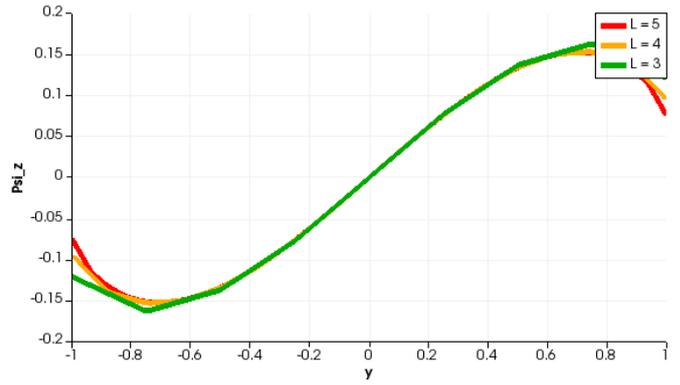


Fig. 11: ψ_z の $(x, z) = (-1, 1)$ における分布

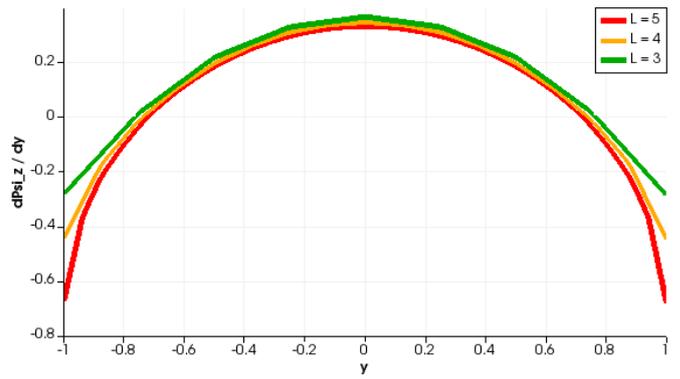


Fig. 12: $\frac{\partial \psi_z}{\partial y}$ の $(x, z) = (-1, 1)$ における分布

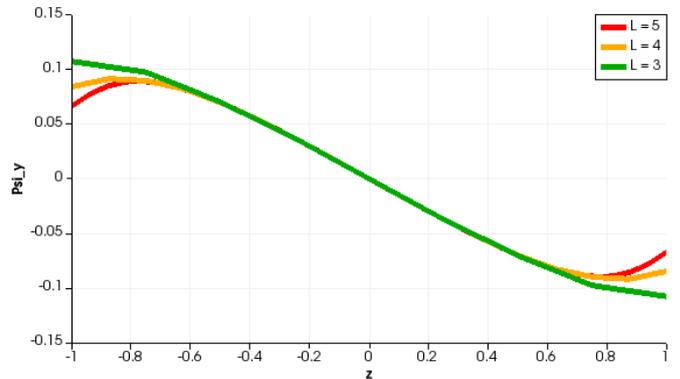


Fig. 13: ψ_y の $(x, y) = (-1, 1)$ における分布

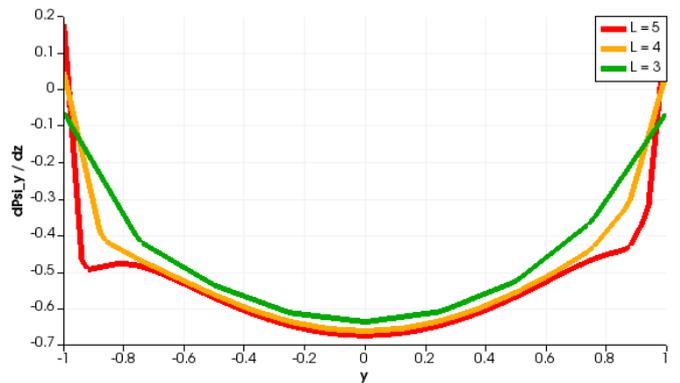


Fig. 14: $\frac{\partial \psi_y}{\partial z}$ の $(x, z) = (-1, 1)$ における分布