

中央大学大学院理工学研究科情報工学専攻
修士論文

煙シミュレーションのための部分空間法的高速化

Accelerated Subspace Method for 3D Smoke Simulation

須之内 俊樹
Toshiki SUNOUCHI
学籍番号 23N8100018B

指導教員 森口 昌樹 准教授

2025年3月

概 要

キーワード: 流体シミュレーション, 部分空間法, Snapshot 固有直交分解, Cubature

目次

第1章 序論	1
第2章 関連研究	2
2.1 流体シミュレーション	2
2.2 部分空間法	4
第3章 煙の流体シミュレーション	7
3.1 格子法による離散化	7
3.1.1 外力項計算	8
3.2 ボリュームデータの可視化手法	9
第4章 部分空間法	10
4.1 基底空間構築手法	10
4.2 流体ソルバへの部分空間法の適用	10
4.3 部分空間法の課題	10
第5章 部分空間法の高高速化手法の提案	11
5.1 Snapshot の分割による高速化	11
5.2 領域分割による高速化	11
5.3 計算機実験	11
5.3.1 計算条件	11
5.3.2 計算結果	11
5.4 課題	11
第6章 結論	12
謝辞	19
参考文献	20

第1章 序論

流体シミュレーションとは、コンピュータを用いて、流体の位置、速度、圧力などの物理量を計算するものである。表現できる流体は、空気、ガスなどの気体や、水、土砂の混ざった水、蜂蜜のような粘性がある液体、砂粒などの固体の粒など多岐にわたる。また、2次元と3次元を扱うことができる。2次元では、水面に広がる波紋や、異なる流体が作り出すマーブル模様などがシミュレーションでき、3次元では私たちの身の回りの現象を広くシミュレーションすることができる。計算した結果を人間が見やすいように可視化することで、流体力学の理論だけでは把握することが困難な、空間に広がる流体の物理量の分布を把握することができるようになる。

流体シミュレーションは、工学分野とコンピュータグラフィックスの分野で活用されている。工学分野では、流体に触れる製品の設計・開発において、製品と流体の相互関係を事前にシミュレーションすることで、実験のコスト削減などに役立っている。コンピュータグラフィックスの分野では、水や煙のそれらしいアニメーションを生成し、映像作品やゲームの演出に役立っている。コンピュータグラフィックスの分野では、現実の物理現象に忠実な計算よりも、それらしい振る舞いを高速に計算することや、ユーザーが表現を操作しやすいことが重要視されている。

近代的な高解像度の流体表現は計算負荷が高く、高速化の研究が盛んに行われている。コンピュータグラフィックスの分野における高速化において、大規模並列計算手法の適用性が重要視されている。流体シミュレーションの計算はGPUによる高速化と相性が良く、高速化手法もGPUに適用できるものが望まれている。また、前処理によってシミュレーション計算に利用する行列の次元を削減し計算負荷を削減する、部分空間法が存在する。これらの手法はシミュレーション計算は高速に行うことができるが、前処理の計算負荷や空間計算量が大きいことに注意が必要である。

本研究では、部分空間法の計算負荷削減により、前処理にかかる時間の短縮を目指す。

第2章 関連研究

2.1 流体シミュレーション

まず、流体シミュレーションの分野で一般的に使われている表記の説明をする。

- \mathbf{x} : 流体の位置を表すベクトル. シミュレーションする空間によって2次元, または3次元になる.
- t : 時刻を表す変数. シミュレーション開始の時刻を0とする.
- $\Delta x, \Delta t$: 離散化する計算格子の格子幅, 次の時刻までの時間幅.
- $\mathbf{u}(\mathbf{x}, t)$: 位置 \mathbf{x} , 時刻 t での流体の速度ベクトル.
- $p(\mathbf{x}, t)$: 位置 \mathbf{x} , 時刻 t での流体の圧力値.
- ρ, ν : それぞれ, 流体の密度, 流体の粘性. ここでは位置や時刻によらない定数とする.
- \mathbf{f} : 位置 \mathbf{x} , 時刻 t での流体にかかる外力ベクトル. 重力などはここに含める. シミュレーションの手法によっては外力を位置 \mathbf{x} , 時刻 t によって変化する外力を扱うことができるが, 今回は簡単のため, 重力のみを扱い, どの位置でも, どの時刻でも一定の値として考える.
- ∇ : 空間微分演算子ナブラ. スカラー場に作用させると勾配を表し, ベクトル場に作用させると発散を表す. 3次元空間なので, $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$
- $\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u}(\mathbf{x}, t) \cdot \nabla$: ラグランジュ微分, または物質微分. 流体力学のような, 連続体を扱う力学で用いられており, 流れに沿って移動する物体と同じように移動する観測者から見た, 物理量の時間変化率を表している.

下記の式 2.1 で表される式を, ナビエ・ストークス方程式とよび, これは流体力学の支配方程式である. 非線形二階微分方程式となっており, 代数的に一般解を求める事ができない. 下記の式 2.2 は流体の連続の式と呼ばれる, 質量保存や流量保存を表す式である.

水や砂粒のように圧縮されない流体を扱うときは、流体の密度が常に一定、つまり密度の時間微分が0になり、それに伴い式2.2の左辺の第二項も0になる。これにより、式2.3と、式2.4を得る。式2.1と式2.4または式2.3を連立することで非圧縮性流体の速度を求めることができる。ナビエ・ストークス方程式を扱う際は、コンピューターで近似解を解析的に求める手法が用いられる。

$$\frac{\partial}{\partial t} \mathbf{u}(\mathbf{x}, t) = -(\mathbf{u}(\mathbf{x}, t) \cdot \nabla) \mathbf{u}(\mathbf{x}, t) - \frac{1}{\rho} \nabla p(\mathbf{x}, t) + \nu \nabla^2 \mathbf{u}(\mathbf{x}, t) + \mathbf{f} \quad (2.1)$$

$$\frac{D}{Dt} \rho + \nabla \cdot \mathbf{u}(\mathbf{x}, t) = 0 \quad (2.2)$$

$$\frac{D}{Dt} \rho = 0 \quad (2.3)$$

$$\nabla \cdot \mathbf{u}(\mathbf{x}, t) = 0 \quad (2.4)$$

一般の非線形微分方程式は厳密な解析的解法が存在せず、様々な条件を設定した上で、数値解法により離散化し、近似解を求めるのが一般的である。ナビエ・ストークス方程式の数値解法は、中間の速度 $\mathbf{u}_i^*(\mathbf{x}, t)$ を用いて、右辺の項について個別に方程式の近似解を求める手法が一般的である。式(2.1)の右辺の第一項を移流項、第二項を圧力項、第三項を粘性項、第四項を外力項とよぶ。

Losasso らが提案した Marker And Cell 法 (MAC 法) [1] と呼ばれる手法は、時間微分の離散化に前進差分を用いてナビエ・ストークス方程式の数値解を求める代表的な手法である。MAC 法は仮の速度 \mathbf{u}^* を用いてナビエ・ストークス方程式を分割し、それぞれ個別に計算を行う。

$$\mathbf{u}(\mathbf{x}, t + \Delta t) = \mathbf{u}^* - \Delta t \left(\frac{1}{\rho} \nabla p(\mathbf{x}, t) + \nu \nabla^2 \mathbf{u}(\mathbf{x}, t) + \mathbf{f} \right) \quad (2.5)$$

$$\mathbf{u}^* = \mathbf{u}(\mathbf{x}, t) - \Delta t (\mathbf{u}(\mathbf{x}, t) \cdot \nabla) \mathbf{u}(\mathbf{x}, t) \quad (2.6)$$

ナビエ・ストークス方程式の空間の離散化については、大きく分けて格子法 (Eulerian) と粒子法 (Lagrangian) がある。格子法とは、流体を扱う空間を正方形や立方体の格子に区切り、格子の中心や辺、頂点などに物理量を配置して計算する方法である。区切る格子の数が多いほど、流体の詳細な動きが計算できるが、計算負荷は増加する。格子法の利点は、規則正しく並んだ格子で空間を離散化することで、微分演算を差分法などの方法で近似することができることや、境界条件の設定が容易であることが挙げられる。一方で、格子法の欠点は、格子の大きさよりも細かい表現が正確に行えないことや、格子内の流体の

粒子の運動を平均化することにより、個々の粒子の詳細な運動を追跡することができないことが挙げられる。これらの欠点により、薄く広がった流体の運動や、水飛沫などを表現することが困難である。

粒子法は、流体の粒子の運動を追跡する手法であり、薄く広がった流体の運動や水飛沫などを表現することが容易である利点がある一方で、計算精度の保証が不十分であることや、境界条件の設定が煩雑になるという欠点がある。

格子法と粒子法は固体にも適用することが可能であり、流体とは異なる支配方程式を用いて変形や移動を計算することができる。流体と固体の相互関係をシミュレーションする際は、格子法と粒子法をそれぞれ個別に適用することが可能である。また、流体の移流項のみを粒子法を用いて計算し、そのほかを格子法で計算するなど、各項の計算について個別に適用する手法が存在する。

2.2 部分空間法

部分空間法は、ベクトル空間をより低次元の部分空間に射影し、数値シミュレーションにおける計算負荷を軽減する手法である。流体シミュレーションにおける部分空間法は、モデル縮約やモデル縮退とも呼ばれている。扱うベクトルデータやシミュレーションの計算を低次元の部分空間に射影して行い計算負荷を軽減する。以下では、部分空間法の理論の概要を説明する。

シミュレーションの前処理として、 n 次元ベクトル \mathbf{x} を r 次元ベクトル \mathbf{y} に変換する、 $n \times r$ 直交行列 \mathbf{A} を考える。以下の 2 式が成り立つ。

$$\mathbf{y} = \mathbf{A}^T \mathbf{x}$$

$$\mathbf{x} = \mathbf{A} \mathbf{y}$$

また、 $\mathbf{y}' = \mathbf{A}^T \mathbf{x}'$ を満たす n 次元ベクトル \mathbf{x}' 、 r 次元ベクトル \mathbf{y}' について、以下の 2 式が成り立つような、任意の $m \times n$ 行列 \mathbf{F} 、任意の $m \times r$ 行列 $\tilde{\mathbf{F}}$ を考える。

$$\mathbf{x}' = \mathbf{F} \mathbf{x}$$

$$\mathbf{y}' = \tilde{\mathbf{F}} \mathbf{y}$$

$\mathbf{y}' = \mathbf{A}^T \mathbf{x}'$ に、 $\mathbf{x}' = \mathbf{F} \mathbf{x}$ と、 $\mathbf{x} = \mathbf{A} \mathbf{y}$ を代入することで、

$$\mathbf{y}' = (\mathbf{A}^T \mathbf{F} \mathbf{A}) \mathbf{y}$$

が得られる． $\mathbf{y}' = \tilde{\mathbf{F}}\mathbf{y}$ と比較すると，

$$\tilde{\mathbf{F}} = \mathbf{A}^T \mathbf{F} \mathbf{A}$$

が得られる．

以上により，任意のベクトルに対する行列積を， $n \times r$ 直交行列 \mathbf{A} を用いて部分空間に射影することができる．流体シミュレーションにおいて，部分空間に射影したベクトルを用いてシミュレーション計算をするためには，射影後のベクトルも流体の非圧縮性条件等の性質を満たさなければならない．そのような射影を達成する行列 \mathbf{A} を得る手法として，Snapshot 固有直交分解がある．行列 \mathbf{A} を既存の流体のデータを主成分分析を用いて計算することで，流体の性質を満たした射影が可能である．以降は Snapshot 固有直交分解の具体的な計算方法について説明する．

時刻 $t \{t \in N, 0 \leq t \leq m\}$ の n 次元ベクトルデータ \mathbf{u}_t を用いて， $n \times m$ 行列 \mathbf{S} を定義する．

$$\mathbf{S} = \begin{bmatrix} | & | & & | \\ \mathbf{u}_0 & \mathbf{u}_1 & \cdots & \mathbf{u}_{m-1} \\ | & | & & | \end{bmatrix}$$

次に，行列 \mathbf{S} 主成分分析を行うため， $n \times r$ 行列 \mathbf{A} について以下の最小化問題を考える．ここで r は抽出したい主成分の数である．

$$\min \|\mathbf{S} - \mathbf{A} \mathbf{A}^T \mathbf{S}\|_F^2$$

この最小化問題の解となる行列 \mathbf{A} は， $\mathbf{S} \mathbf{S}^T$ の固有ベクトル $\mathbf{v}_i (0 \leq i \leq r-1)$ を用いて，以下の行列になる．

$$\mathbf{A} = \begin{bmatrix} | & | & & | \\ \mathbf{v}_0 & \mathbf{v}_1 & \cdots & \mathbf{v}_{r-1} \\ | & | & & | \end{bmatrix}$$

ここで，固有ベクトル \mathbf{v}_i に対応する固有値を λ_i は， $\lambda_0 \geq \lambda_1 \cdots \geq \lambda_i \geq \lambda_{n-1}$ を満たすとする． $\mathbf{S} \mathbf{S}^T$ は対称行列になっているため，固有ベクトルは互いに直交し， $\mathbf{A}^{-1} = \mathbf{A}^T$ となる．

次に，行列 \mathbf{A} を用いて \mathbf{u} を \mathbf{y} に射影するとき， \mathbf{y} が流体の非圧縮性条件 $\nabla \cdot \mathbf{y} = 0$ を満たしていることを確かめる．行列 \mathbf{S} の各列ベクトル \mathbf{u}_i は既存のシミュレーションの速度データであるため，流体の非圧縮性条件 $\nabla \cdot \mathbf{u}_i = 0$ を満たす．このことから，

$$\nabla \cdot \mathbf{S} = \sum_{0 \leq i \leq n-1} \nabla \cdot \mathbf{u}_i = 0$$

が成り立つ. 行列 $\mathbf{S}\mathbf{S}^T$ の固有ベクトル \mathbf{v}_i と, それに対応する固有値を λ_i とすると, 固有値と固有ベクトルの定義から以下が成り立つ.

$$\mathbf{S}\mathbf{S}^T \mathbf{v}_i = \lambda_i \mathbf{v}_i$$

両辺に左から $\nabla \cdot$ を取ることにより,

第3章 煙の流体シミュレーション

流体シミュレーションを行う際、境界条件や外力項を適切に設定することで、流体の振る舞いを操作することができる。この章では、部分空間法を用いて高速化するシミュレーションを実装する具体的な手法を紹介する。この手法で得られたシミュレーションデータを用いて、部分空間法に用いる Snapshot 固有直交分解を行う。

3.1 格子法による離散化

格子法による空間の離散化は、扱う格子によって様々な手法が存在する。例えば物理量を格子の中心に配置するコロケート格子や、速度成分を格子の面の中心に配置するスタaggered格子が存在する。コロケート格子は流体の格子の形状によらず適用できるが、境界条件の設定が煩雑である。スタaggered格子は適用が立方体格子などに限定される一方、境界条件の設定が容易である。本論文では、シミュレーション空間をスタaggered格子を用いて離散化する。スタaggered格子は下記の図 6.1 のように、格子の中心を圧力の位置に定義し、格子面の中心に、その格子面と垂直な流速の成分の位置を定義する。例えば xy 平面に並行な格子面には、その地点での流速の z 成分を配置する。シミュレーション空間の x , y , z 軸方向の分割数を n_x , n_y , n_z とする。スタaggered格子によってシミュレーションの物理量を、以下のベクトルに離散化する。

- \mathbf{v}_x : $(n_x + 1) \times n_y \times n_z$ 次元ベクトル.
- \mathbf{v}_y : $n_x \times (n_y + 1) \times n_z$ 次元ベクトル.
- \mathbf{v}_z : $n_x \times n_y \times (n_z + 1)$ 次元ベクトル.
- \mathbf{p} : $n_x \times n_y \times n_z$ 次元ベクトル.

\mathbf{v}_x , \mathbf{v}_y , \mathbf{v}_z は速度の x , y , z 成分, \mathbf{p} は圧力である。ここで、前進差分を用いた空間の偏微分の離散化の例として、スタaggered格子を用いた以下の偏微分の離散化を上記のベクトルを用いて表現することを考える。

$$\frac{\partial}{\partial x} \mathbf{u}(\mathbf{x}) = \frac{\mathbf{u}(\mathbf{x} + \Delta \mathbf{x}) - \mathbf{u}(\mathbf{x})}{\Delta x}$$

ここで、 $\mathbf{u}(\mathbf{x})$ は位置 \mathbf{x} での速度を表す三次元ベクトルである。スタaggerド格子の一辺の長さを $\Delta L = \Delta \mathbf{x}$ 、流体の位置ベクトル \mathbf{x} を自然数 i ($0 \leq i \leq nx-1$)、 j ($0 \leq j \leq ny-1$)、 k ($0 \leq k \leq nz-1$) を用いて、 $\mathbf{x} = (i\Delta L, j\Delta L, k\Delta L)$ とすると、以下のようになる。

$$\frac{\mathbf{u}(\mathbf{x} + \Delta \mathbf{x}) - \mathbf{u}(\mathbf{x})}{\Delta \mathbf{x}} = \begin{bmatrix} \frac{v_x(i+1, j, k) - v_x(i, j, k)}{\Delta L} \\ \frac{v_y(i, j+1, k) - v_y(i, j, k)}{\Delta L} \\ \frac{v_z(i, j, k+1) - v_z(i, j, k)}{\Delta L} \end{bmatrix}$$

このような離散化を用いて、以降は気体の流体シミュレーションにおけるナビエ・ストークス方程式の計算方法を説明する。

$$\mathbf{u}(\mathbf{x}, t + \Delta t) = \mathbf{u}^* - \Delta t \left(\frac{1}{\rho} \nabla p(\mathbf{x}, t) + \nu \nabla^2 \mathbf{u}(\mathbf{x}, t) + \mathbf{f} \right) \quad (3.1)$$

$$\mathbf{u}^* = \mathbf{u}(\mathbf{x}, t) - \Delta t (\mathbf{u}(\mathbf{x}, t) \cdot \nabla) \mathbf{u}(\mathbf{x}, t) \quad (3.2)$$

上記の式 3.1 と式 3.2 を以下のように各項ごとに分割して計算する。

$$\mathbf{u}_0(\mathbf{x}) = \mathbf{u}(\mathbf{x}, t) - \Delta t \mathbf{f} \quad (3.3)$$

$$\mathbf{u}_1(\mathbf{x}) = \mathbf{u}_0(\mathbf{x}) - \Delta t (\mathbf{u}_0(\mathbf{x}) \cdot \nabla) \mathbf{u}_0(\mathbf{x}) \quad (3.4)$$

$$\mathbf{u}_2(\mathbf{x}) = \mathbf{u}_1(\mathbf{x}) - \Delta t \nu \nabla^2 \mathbf{u}_1(\mathbf{x}) \quad (3.5)$$

$$\mathbf{u}(\mathbf{x}, t + \Delta t) = \mathbf{u}_3(\mathbf{x}) = \mathbf{u}_2(\mathbf{x}) - \Delta t \left(\frac{1}{\rho} \nabla p(\mathbf{x}) \right) \quad (3.6)$$

ここで、式 3.3 は外力項、式 3.4 は移流項、式 3.5 は粘性項、式 3.5 は粘性項、式 3.6 は圧力項を計算する式であり、これらの両辺を全て足し合わせると元のナビエ・ストークス方程式が得られる。

3.1.1 外力項計算

$$\mathbf{u}_0 = \mathbf{u}(\mathbf{x}, t) - \Delta t \mathbf{f} \quad (3.7)$$

外力項では、再現したい現象になるような外力を与えなければならない。コンピュータグラフィックスにおける気体の外力は、気体にかかる重力と、温度差による対流を発生させ

る力，気体が渦巻くように与える力などをユーザーがパラメータによって調整できるようにする計算方法が一般的である．よりそれらしい振る舞いを再現するため，他の計算では定数として扱う密度や温度を，ここでは位置と時刻によって変化するデータとして計算する．位置 x における気体の密度，温度をそれぞれ $\rho(x)$ ， $T(x)$ とし，

3.2 ボリュームデータの可視化手法

第4章 部分空間法

4.1 基底空間構築手法

4.2 流体ソルバへの部分空間法の適用

4.3 部分空間法の課題

第5章 部分空間法的高速化手法の提案

5.1 Snapshot の分割による高速化

5.2 領域分割による高速化

5.3 計算機実験

5.3.1 計算条件

5.3.2 計算結果

5.4 課題

第6章 結論

$$\frac{\partial}{\partial t} \mathbf{u}(\mathbf{x}, t) = -\mathbf{u}(\mathbf{x}, t) \frac{\partial}{\partial \mathbf{x}} \mathbf{u}(\mathbf{x}, t) \quad (6.1)$$

格子法における非線形項の計算

式 (3.1) の粘性項は先述の通り省略し、外力項は事前に流速に与えて扱うことができるため省略する。残った圧力項の計算について考える。ただし、移流項を先に計算し、 \mathbf{u}^* は既知のものとする。

$$\mathbf{u}(\mathbf{x}, t+1) = \mathbf{u}^* - \frac{\Delta t}{\rho} \nabla p(\mathbf{x}, t) \quad (6.2)$$

両辺の発散をとると、

$$\nabla \cdot \mathbf{u}(\mathbf{x}, t+1) = \nabla \cdot \mathbf{u}^* - \frac{\Delta t}{\rho} \nabla^2 p(\mathbf{x}, t)$$

が得られ、流体の非圧縮性条件、 $\nabla \cdot \mathbf{u}(\mathbf{x}, t) = 0$ により、左辺は0になるため、

$$\nabla \cdot \mathbf{u}^* = \frac{\Delta t}{\rho} \nabla^2 p(\mathbf{x}, t) \quad (6.3)$$

が得られる。この式を満たすような圧力を求め、上記の式 (6.2) に代入することによって、次の時刻の流速 $\mathbf{u}(\mathbf{x}, t+1)$ が計算できる。式 (6.3) のような微分方程式はポアソン方程式と呼ばれ、境界条件として、扱う領域の境界での $p(\mathbf{x}, t)$ の値を計算前に定めることで、式を満たすような変数を求めることができる。

図 6.1: スタッガード格子

図 6.2: 圧力計算のモデル

圧力項の計算を、スタッガード格子を用いて離散化するため、先ほどの式 (6.3) を考える。

$$\nabla \cdot \mathbf{u}^* = \frac{\Delta t}{\rho} \nabla^2 p(\mathbf{x}, t)$$

スタaggerド格子では圧力と流速が配置してある位置がずれている．従って，空間微分に対して同じ差分を適用すると，同じ幅で差分を取ったとしても離散化される空間の位置がずれてしまうため，空間微分に対して同じ差分は適応できない．そこで差分を取るのではなく，両辺をセルの領域で積分して考える．

$$\int_V \nabla \cdot \mathbf{u}^* dV = \int_V \frac{\Delta t}{\rho} \nabla^2 p(\mathbf{x}, t) dV$$

両辺にガウスの発散定理を適用すると，周回積分の形で表せる．

$$\oint_V \mathbf{u}^* \cdot \mathbf{n} dV = \oint_V \frac{\Delta t}{\rho} \nabla p(\mathbf{x}, t) \cdot \mathbf{n} dV$$

この式はスタaggerド格子において，周回積分は格子の各面で境界条件を考えて，全ての面の物理量を足すことで表すことができる．ここで離散化のため， x 軸方向に i 番目， y 軸方向に j 番目， z 軸方向に k 番目の格子を $g_{i,j,k}$, $g_{i,j,k}$ の圧力を $p_{i,j,k}$ と定義し，図 6.2 のようなモデルを考える．

まず，整数 $n = 0, 1, 2, \dots, 5$ までは，それぞれ格子の各面の方向に割り振る．図 6.2 の場合， $n = 0$ は x 軸正の方向， $n = 1$ は y 軸正の方向， $n = 2$ は x 軸負の方向， $n = 3$ は y 軸負の方向， $n = 4$ は z 軸正の方向， $n = 5$ は z 軸負の方向に割り振る．ここで，この整数 n を使って，いくつかの配列や変数を定義する．

- F_n 格子の面の各方向に隣接しているものが流体なら 1，壁面なら 0 を取るブーリアン値．

図 6.2 の場合， $n = 0$ に対応する x 軸正の方向と， $n = 5$ に対応する z 軸負の方向は壁に隣接していて格子がないため， $F_0, F_5 = 0, F_1, F_2, F_3, F_4 = 1$ となる．

- D_n 格子の面の各方向に垂直で，格子の外側を向く単位ベクトルを表す係数．向いている方向が座標軸の正の向きなら +1，負の向きなら -1 を取る整数値．

図 6.2 の場合，座標軸の正の方向に向いているのは $n = 0, n = 1, n = 4$ に対応する方向で，それ以外は負の方向に向いているため， $F_0, F_1, F_4 = 1, F_2, F_3, F_5 = -1$ となる．

- p_n 割り振られた方向に隣り合っている格子の圧力値．図 6.2 の場合， $p_0 = p_{i+1,j,k}$, $p_1 = p_{i,j+1,k}$, $p_2 = p_{i-1,j,k}$, $p_3 = p_{i,j-1,k}$, $p_4 = p_{i,j,k+1}$, $p_5 = p_{i,j,k-1}$ となるが，境界条件を考えると，壁に隣接している格子との圧力値の勾配を 0 にするため， $p_0 = p_{i,j,k}$, $p_5 = p_{i,j,k}$ である．

- ∇p_n 割り振られた方向の圧力勾配. 前方差分によって, $\nabla p_n(\mathbf{x}, t) = \frac{p_n - p_{i,j,k}}{\Delta x}$ と近似できる.
- u_n $g_{i,j,k}$ に配置されている流速値で, 割り振られた方向の格子面に配置されている流速値. 壁に隣接している方向は流速を 0 にするため, $u_0 = 0$, $u_5 = 0$ である.

定義した配列や変数を用いて, 周回積分の形で表された式を離散化することで,

$$\sum_n \mathbf{u}_n^* D_n F_n \Delta x = \sum_n \frac{\Delta t}{\rho} \nabla p_n(\mathbf{x}, t) F_n \Delta x$$

が得られ, 圧力勾配を $\nabla p_n(\mathbf{x}, t) = \frac{p_n - p_{i,j,k}}{\Delta x}$ で近似すると,

$$\sum_n \mathbf{u}_n^* D_n F_n \Delta x = \sum_n \frac{\Delta t}{\rho} \frac{p_n(\mathbf{x}, t) - p_{i,j,k}(\mathbf{x}, t)}{\Delta x} F_n \Delta x$$

が得られる. この式を整理すると,

$$\sum_n \mathbf{u}_n^* D_n F_n = \sum_n \frac{\Delta t}{\rho} \frac{p_n(\mathbf{x}, t) - p_{i,j,k}(\mathbf{x}, t)}{\Delta x} F_n \quad (6.4)$$

が得られる. 式 (6.4) は p についての連立一次方程式となっている. 全ての格子に対して式 (6.4) を考えることになるため, 連立方程式は規模が大きくなる. 従って, 直接法ではなく反復法を用いて, 必要な精度で計算を打ち切って計算する. 連立方程式の解法としては, 前処理付き共役勾配法がよく用いられている.

圧力項の計算を離散化した式 (6.4) は, 求めたい圧力分布の式 (6.2) を満たすような解が得られるが, 得られた解がシミュレーション内で, 良い圧力として振る舞うかどうかは考えなければならない. ディリクレ境界条件は, 扱う領域を流体と気体の領域に分割し, 流体の物理量のみを考える際に用いる境界条件で, 流体が存在しない格子の圧力値を 0 とする. ディリクレ境界条件を考えずに流体の圧力のみを計算すると, 気体の領域とされている場所にも流体の圧力が計算できてしまう. そのため, 気体と液体の領域に分割するようなシミュレーションで, 気体の圧力が液体の圧力に及ぼす影響が小さい場合は, 圧力にディリクレ境界条件を追加した方が, より良い圧力分布になると言える.

粒子法とは, 流体をいくつかの小さな粒子の集まりとして考え, 各粒子に物理量を与えて計算をする方法である. 本来は流体は, アボガドロ数 ($N_A = 6.02 * 10^{23}$) 単位の子が動いているが, これら全てを計算機で扱うのは現実的でないため, いくつかの分子の動きを一つの粒子にまとめて扱う. 粒子の数が多いほど実際の現象に近づき, 精度が良くなるが, 計算負荷は増加する. 利点として, 以下の計算によって, 移流項の計算に数値拡

散が発生しないことが確認できるため、移流項が精度良く計算できる。また、格子法では水飛沫の表現が困難であるが、粒子法は粒子が存在する場所に液体が存在すると考えるため、粒が飛び散りさへすれば水飛沫のような表現になる。欠点として、粒子が飛び散ったときに粒子の影響範囲内に他の粒子がないとき、計算がうまくできないことから、正確な物理量の分布が計算できる訳ではない。また、対象の流体以外は計算しないことが多いため、例えば水と空気が入り混じるようなシミュレーションを考える際、水の計算のみをすることで、空気の影響が無視されてしまう。簡単のため、ナビエストークス方程式から粘性項と外力項を取り除いた、オイラー方程式と呼ばれるものについて考える。

$$\frac{\partial}{\partial t} \mathbf{u}(\mathbf{x}, t) = -(\mathbf{u}(\mathbf{x}, t) \cdot \nabla) \mathbf{u}(\mathbf{x}, t) - \frac{1}{\rho} \nabla p$$

粒子法はラグランジュ微分を用いることで、移流項の計算が単純になる。時刻 t_0 に位置 \mathbf{x}_0 にある流体の速度を $\mathbf{u}(\mathbf{x}_0, t_0)$ とする。 Δt 秒後の位置は $\mathbf{x}_0 + \mathbf{u}(\mathbf{x}_0, t_0) \Delta t$ になっているので、時刻 $t_0 + \Delta t$ の流体の速度は、 $\mathbf{u}(\mathbf{x}_0 + \mathbf{u}(\mathbf{x}_0, t_0) \Delta t, t_0 + \Delta t)$ となる。この間の速度変化 $\Delta \mathbf{u}$ をテイラー展開を用いて表すと、

$$\Delta \mathbf{u} = \mathbf{u}(\mathbf{x}_0 + \mathbf{u} \Delta t, t_0 + \Delta t) - \mathbf{u}(\mathbf{x}_0, t_0) = \frac{\partial \mathbf{u}(\mathbf{x}_0, t_0)}{\partial \mathbf{x}} \mathbf{u}(\mathbf{x}_0, t_0) \Delta t + \frac{\partial \mathbf{u}(\mathbf{x}_0, t_0)}{\partial t} \Delta t$$

両辺を Δt で割ると、

$$\begin{aligned} \frac{\Delta \mathbf{u}}{\Delta t} &= \frac{\partial \mathbf{u}(\mathbf{x}_0, t_0)}{\partial \mathbf{x}} \mathbf{u}(\mathbf{x}_0, t_0) + \frac{\partial \mathbf{u}(\mathbf{x}_0, t_0)}{\partial t} \\ \frac{\Delta \mathbf{u}}{\Delta t} &= (\mathbf{u} \cdot \nabla) \mathbf{u} + \frac{\partial \mathbf{u}(\mathbf{x}_0, t_0)}{\partial t} \end{aligned}$$

ラグランジュ微分、つまり、粒子を流れに沿って動かしたときの変化量が、オイラー方程式の移流項になっている。粒子法では、物理量をもった粒子自体を動かすだけで、移流項の計算をすることができる。

$$\sum_n \mathbf{u}_n^* D_n F_n = \sum_n \frac{\Delta t}{\rho} \frac{\mathbf{p}_n(\mathbf{x}, t) - \mathbf{p}_{i,j,k}(\mathbf{x}, t)}{\Delta \mathbf{x}} F_n \quad (6.5)$$

圧力項は圧力を変数とした、上記の連立方程式を解く。格子の総数を n とすると、全ての格子で連立方程式を考えるため、合計 n 本の連立方程式を解くことになる。そこで、 $\mathbf{A}\mathbf{x} = \mathbf{b}$ に対応させたものを解く。 \mathbf{A} 、 \mathbf{b} それぞれのサイズは、 \mathbf{A} は $n \times n$ であり、 \mathbf{b} は n である。この際、3次元に分布する圧力をベクトルにうつす。具体的には、圧力の (i, j, k) 成分は、ベクトルの $iN_y N_z + jN_x + k$ 成分にうつす。また、 \mathbf{A} の $(iN_y N_z + jN_x + k, lN_y N_z + mN_x + n)$ の成分には、位置 (i, j, k) の格子についての連立方程式の、圧力の (l, m, n) 成分にかか

る係数を保存すれば良い．通常の行列のままだと空間計算量や時間計算量をが非常に大きくなってしまいますが， \mathbf{A} のほとんどの成分は 0 であることを利用し，Eigen の疎行列ライブラリを用いることで，空間計算量や時間計算量を削減できる．連立方程式の解法は，未知数が多いため，直接法で計算するのは現実的ではない．反復法では，必要な精度に解が収束したら計算を打ち切るため，解の収束が速い方が望ましい．前処理をすることで解の収束が速くなるため，前処理付き共役勾配法を採用し，Eigen の前処理付き共役勾配法を計算する関数を用いて連立方程式を解く． \mathbf{A} と \mathbf{b} の各成分は以下のように計算する．

- \mathbf{A} , \mathbf{b} を初期化する．この際， \mathbf{b} の成分は全て 0 にしておく．
- 全粒子について，ディリクレ境界条件を考え，粒子が存在しない空間の圧力を 0 で固定する．これは，`standerdgrid` の (i, j, k) に格納されている粒子の数を見て，0 なら \mathbf{A} の $(iN_yN_z + jN_x + k, iN_yN_z + jN_x + k)$ 成分を 1 にして，処理を終了すれば良い．
- 粒子が入っていたセル (i, j, k) について，式 (6.4) に従って係数を計算する．セル (i, j, k) の周囲の 6 方向に格納されている流速について，シミュレーションの境界で，壁に隣接していない方向についての流速の周回積分を計算する．周回積分の値を， \mathbf{b} の $(iN_yN_z + jN_x + k)$ 成分に代入する．
- セル (i, j, k) の周囲の 6 方向のセルについて，粒子が入っていないセルの方向，または，壁に隣接している方向なら疎行列には何もせず，それ以外なら疎行列の $iN_yN_z + jN_x + k$ 行の，それぞれの方向に対応した対応した成分に $\frac{\Delta t}{\rho \Delta x^2}$ を代入する．

本手法では，シミュレーションは立方体の内部の領域内で行う．立方体の一辺の長さ L は，スタガード格子の水平方向の数を N ，スタガード格子の一辺の長さを dx とすると， $L = Ndx$ である．8 点 $(0, 0, 0)$, $(L, 0, 0)$, $(0, L, 0)$, $(0, 0, L)$, $(L, L, 0)$, $(0, L, L)$, $(L, 0, L)$, (L, L, L) を頂点とするような立方体の領域を扱うレベルセットサーフェス $\varphi(x, y)$ を，

表 6.1: 表のタイトル

項目 1	項目 2	項目 3
データ 1	データ 2	データ 3
データ 1	データ 2	データ 3
データ 1	データ 2	データ 3

以下のように定義する.

$$\varphi(x, y, z) \left\{ \begin{array}{ll} (x, y, z) & x \leq 0, y \leq 0, z \leq 0, \\ (L - x, y, z) & L \leq x, y \leq 0, z \leq 0, \\ (x, L - y, z) & x \leq 0, L \leq y, z \leq 0, \\ (L - x, L - y, z) & L \leq x, L \leq y, z \leq 0, \\ (x, y, L - z) & x \leq 0, y \leq 0, L \leq z, \\ (L - x, y, L - z) & L \leq x, y \leq 0, L \leq z, \\ (x, L - y, L - z) & x \leq 0, L \leq y, L \leq z, \\ (L - x, L - y, L - z) & L \leq x, L \leq y, L \leq z, \\ (0, 0) & \text{else.} \end{array} \right. \quad (6.6)$$

このようにして定義されるレベルセットサーフェスは、粒子の座標を引数とし、シミュレーションする立方体から外に出ていた場合は、立方体の領域内に最短距離で戻すようなベクトルを返り値として返す。粒子に対して移流計算や粒子位置の補正を行うと、シミュレーション領域から粒子が出ていってしまうことがあるため、そのような計算を行った後に全粒子に対して領域内部に戻す処理を行う。

図・表には通し番号と見出し (caption) を付け、本文中で当該の図・表に言及する。また、単位や目盛を正確に記す。図のタイトルは図の下に、表のタイトルは表の上に書く。

例を図 6.3 と表 6.1 に示す。第 2 章の図には図 2.1, 図 2.2, 図 2.3, …のように番号が振られ、第 2 章の表には表 2.1, 表 2.2, 表 2.3, …のように、図とは独立に番号が振られる。

一例として、和文の著書 [?], 和文の論文誌 [?], 英文の編書 [?], 英文の論文誌 [?], 国際会議 [?], 修士論文 [?], 電子雑誌 [?], Web ページ [?] を、2 ページの参考文献の節に載せる。参考文献には信頼性が高く、後世に残るものを載せるように注意せよ。

図 6.3: 情報工学科のロゴ

書くべき情報は以下のとおりである。

- 和文の著書: 著者, 書名, シリーズ名 (あれば), 発行所, 都市, 年.
- 和文の論文誌: 著者, 題名, 誌名, 巻, 号, 頁, 年.
- 英文の編書: 編者, 書名, 発行所, 都市, 年.
- 英文の論文誌: 著者, 題名, 誌名, 巻, 号, 頁, 年.
- 国際会議: 著者, 題名, 予稿集名, 都市, コード等, 頁, 年.
- 修士論文: 著者, 題名, 機関名, 年.
- 電子雑誌: 著者, 題名, 誌名, 巻, 号, 頁 (オンライン), DOI, 西暦年.
- Web ページ: 著者, Web ページの題名, Web サイトの名称 (オンライン) (ただし, 著者と同じ場合は省略可), 入手先 <URL> (参照日付).

英語の文献はすべて半角文字で書く. 参考文献には本文で引用した文献のみ載せる. 情報処理学会の論文誌の原稿執筆案内 [?] も参考になる.

通し番号は, 引用順または著者名のアルファベット順に付ける. 文献の引用のしかたは分野ごとに異なるので, **自己流では書かず, 当該分野の論文誌などを参考にすること.**

結論には, 研究の成果や意義その他を総括的に**過去形**で述べる.

謝辞

本研究を進めるにあたり，大変多くのご指導，ご助言を頂いた中央大学理工学部情報工学科の森口 昌樹 准教授に深く感謝いたします．また，多大なるご助言，ご協力を頂いた〇〇研究室の皆様には大変お世話になりました．心から感謝いたします．

参考文献

- [1] F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface. *The Physics of Fluids*, 8 (12) : 2182–2189, 1965.
- [2] R. Fedkiew, J. Stam, H. Jensen. Visual simulation of smoke. In *Proceedings of SIGGRAPH 01*, 15–22, 2001.
- [3] T. Kim, J. Delaney. Subspace Fluid Re-Simulation. *ACM Transactions on Graphics*, 32 (4) : 62:1–62:9, 2013.
- [4] A. Jones, P. Sen, and T. Kim. Compressing fluid subspaces. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 77–84, 2016.
- [5] M. Wicke, M. Stanton, and A. Treuille. Modular Bases for Fluid Dynamics. *ACM Transactions on Graphics*, 39:1–39:8, 2009.