

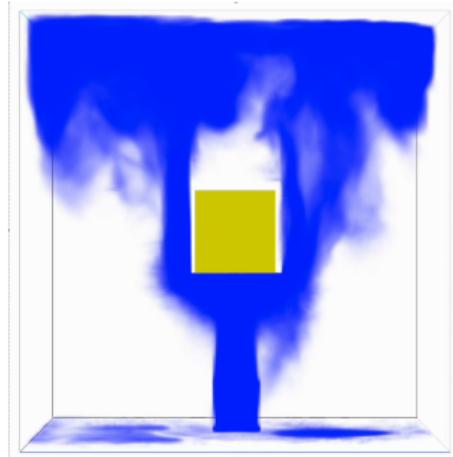
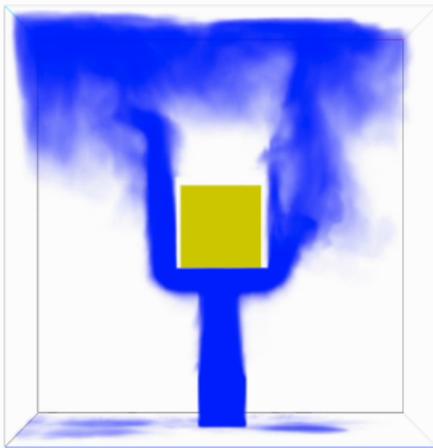
煙シミュレーションのための部分空間法の高速化

須之内 俊樹

中央大学理工学研究科 情報工学専攻
形状情報処理研究室 23N8100018B

2025年2月21日

概要



• 190.54 秒 → 33.43 秒

- 部分空間法：流体シミュレーションの高速化手法
- 煙のシミュレーションに部分空間法を適用
- 提案手法による部分空間法の高速化と、シミュレーション結果の変化を評価

研究背景

流体シミュレーション

CG 分野

- 流体の映像の生成に利用
- 計算負荷の軽減，流体の挙動が制御しやすさ
- 物理的な正確さよりも，それらしさ

流体シミュレーションの課題

- 希薄な流体や，水飛沫は手法によっては再現できない
- 近年は高品質な映像が求められ，計算負荷が大きい

研究背景

流体シミュレーション（煙，液体）の数理モデル

ナビエ・ストークス方程式の離散化

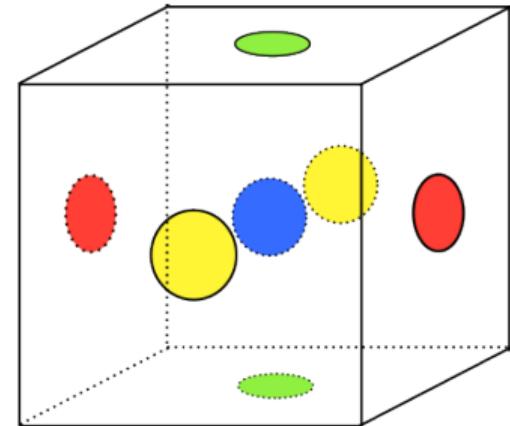
$$\mathbf{u}(t + \Delta t) = \mathbf{u}(t) - \Delta t((\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f})$$

$$\nabla \cdot \mathbf{u} = 0$$

- \mathbf{u} , \mathbf{f} : 流体の速度, 外力
- p , ρ , ν : 流体の圧力, 密度, 粘性
- $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$

スタッガード格子

- 速度と圧力の配置位置を工夫し, 計算の安定性を向上
- 非直交格子への適用が困難



- x軸方向の流速
- y軸方向の流速
- z軸方向の流速
- 圧力

研究背景

流体シミュレーション（煙，液体）の数理モデル

部分段階法

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

中間子 \mathbf{u}_0 から \mathbf{u}_3 を用いて，以下のように各項ごとに分割して計算する

$$\mathbf{u}_0 = \mathbf{u}(t) - \Delta t \mathbf{f}$$

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

$$\mathbf{u}_2 = \mathbf{u}_1 - \Delta t \nu \nabla^2 \mathbf{u}_1$$

$$\mathbf{u}(t + \Delta t) = \mathbf{u}_3 = \mathbf{u}_2 - \Delta t \frac{1}{\rho} \nabla p$$

研究背景

流体シミュレーション（煙，液体）の数理モデル

部分段階法

$$\begin{aligned} \mathbf{u}_0 &= \mathbf{u}(t) - \Delta t \mathbf{f} \\ \mathbf{u}_1(x) &= \mathbf{u}_0(x - \Delta t \mathbf{u}_0) \end{aligned} \quad \dots \quad (1)$$

$$\begin{aligned} \mathbf{u}_2 &= \mathbf{V} \mathbf{u}_1 \\ \mathbf{b} &= \mathbf{W} \mathbf{u}_2 \end{aligned} \quad \dots \quad (2)$$

$$\mathbf{p} = \mathbf{A}^{-1} \mathbf{b} \quad \dots \quad (3)$$

$$\mathbf{u}(t + \Delta t) = \mathbf{u}_3 = \mathbf{u}_2 - \mathbf{Y} \mathbf{p}$$

- 式 1 は semi-Lagrangian 法 [Stam 1999]，式 2, 3 はコリンの射影法 [Chorin 1968]
- \mathbf{V} , \mathbf{A} は離散ラプラシアン行列を用いて ∇^2 を離散化
- \mathbf{W} , \mathbf{Y} は勾配演算子 ∇ を離散化

研究背景

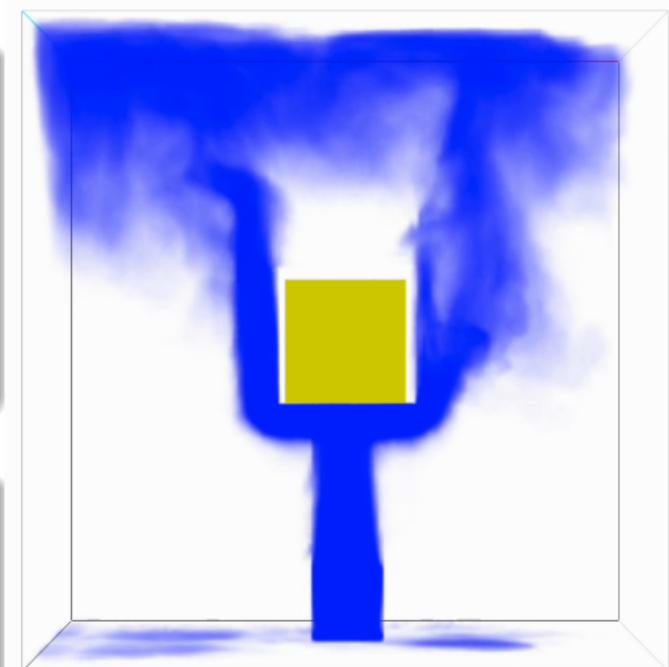
先行研究

Visual Simulation of Smoke [Fedkiw et al. 2001]

- CGにおいて広く用いられる煙のオフラインシミュレーション手法
- 密度を計算した後、ボリュームレンダリングを用いて描画
- 計算量は空間解像度に依存

計算機実験

- 解像度 128^3 , 200 フレーム
- シミュレーション 1800ms/f
- レンダリング 240ms/f



研究背景

部分空間法

部分空間法 (subspace method)

- 既存のシミュレーションの時系列データ（スナップショット）を主成分分析し、次元削減する
- 本研究の高速化の対象は、主成分分析

流体シミュレーションへの適用

- 空間解像度を保ったまま高速化可能
- タイムステップを小さくして再計算できる
- 前処理の計算負荷が大きい
- 次元削減後の流速 \tilde{u} は、流体の非圧縮条件 $\nabla \cdot \tilde{u} = 0$ を満たしていなければならない

研究背景

部分空間法

特異値分解を用いた直交基底の計算

- 既存のシミュレーションの時系列データ（スナップショット）を T 個用いて、部分空間の直交基底を作成する手法
- 空間解像度を n 、削減後の次元を r とすると、 $n \ll T \ll r$
- 時刻 t における速度ベクトル u_t を用いて、 $3(n+1) \times T$ 行列 S を定義する

$$S = [u_0 \quad u_1 \quad \cdots \quad u_{T-1}]$$

- S を特異値分解して得られる左特異行列 U を部分空間の直交基底とする
- 先に QR 分解を適用し、サイズを小さくすることができる

研究背景

先行研究

部分空間上のシミュレーション手法 [Treuille et al. 2006]

- 中間子 u_i から得た基底を U_i , p から得られた基底を P とする
- $n \times n$ 行列が $r \times r$ 行列に. 各ステップが高速化
- 前処理に \tilde{V} , \tilde{W} , \tilde{A} , \tilde{Y} の計算も含める

$$u_2 = Vu_1$$

$$U_2^T u_2 = (U_2^T V U_1) U_1^T u_1$$

$$\tilde{u}_2 = \tilde{V} \tilde{u}_1$$

$$b = W u_2$$

$$P^T b = (P^T W U_2) U_2^T u_2$$

$$\tilde{b} = \tilde{W} \tilde{u}_2$$

$$p = A^{-1} b$$

$$P^T p = (P^T A^{-1} P) P^T b$$

$$\tilde{p} = \tilde{A} \tilde{b}$$

$$u_3 = u_2 - Y p$$

$$U_3^T u_3 = U_2^T u_2 - (U_3^T Y P) P^T p$$

$$\tilde{u}_3 = \tilde{u}_2 - \tilde{Y} \tilde{p}$$

部分空間法の課題

空間計算量

一般に,

$$64^3 \leq n \leq 1024^3$$

$$30 \leq r \leq 150$$

- $n = 256^3$, $r = 50$ のとき, 行列 A のために 10GB ほど必要
- 廉価な GPU ではメモリが足りない
- 削減後の次元の数に制限. $r \leq T$

前処理の計算時間

- $3(n + 1) \times T$ 行列の特異値分解, QR 分解の計算量は $O(nT^2)$
- シミュレーションに用いる行列を, 部分空間へ射影する計算時間

提案手法

スナップショットの分割

$$\mathbf{S} = [\mathbf{u}_0 \quad \mathbf{u}_1 \quad \cdots \quad \mathbf{u}_{T-1}]$$

二分割すると、

$$\mathbf{S}_0 = [\mathbf{u}_0 \quad \cdots \quad \mathbf{u}_{T/2-1}]$$

$$\mathbf{S}_1 = [\mathbf{u}_{T/2} \quad \cdots \quad \mathbf{u}_{T-1}]$$

$O(nT^2)$ の計算負荷が、 $O(2n(T/4))$ に削減できる

評価方法

- 前処理の計算時間を計測
- 累積寄与率を計測
- 相対誤差により精度を評価

実験結果

基底計算の解像度と分割数ごとの実行時間 (秒)

解像度	分割なし	分割数 2	分割数 4	分割数 10
64^3	23.43	14.56	8.98	3.78
128^3	190.54	118.49	69.62	33.43

行列の射影の解像度と分割数ごとの実行時間 (秒)

解像度	分割なし	分割数 2	分割数 4	分割数 10
64^3	3.51	2.82	0.85	0.67
128^3	5.21	3.06	3.05	3.98

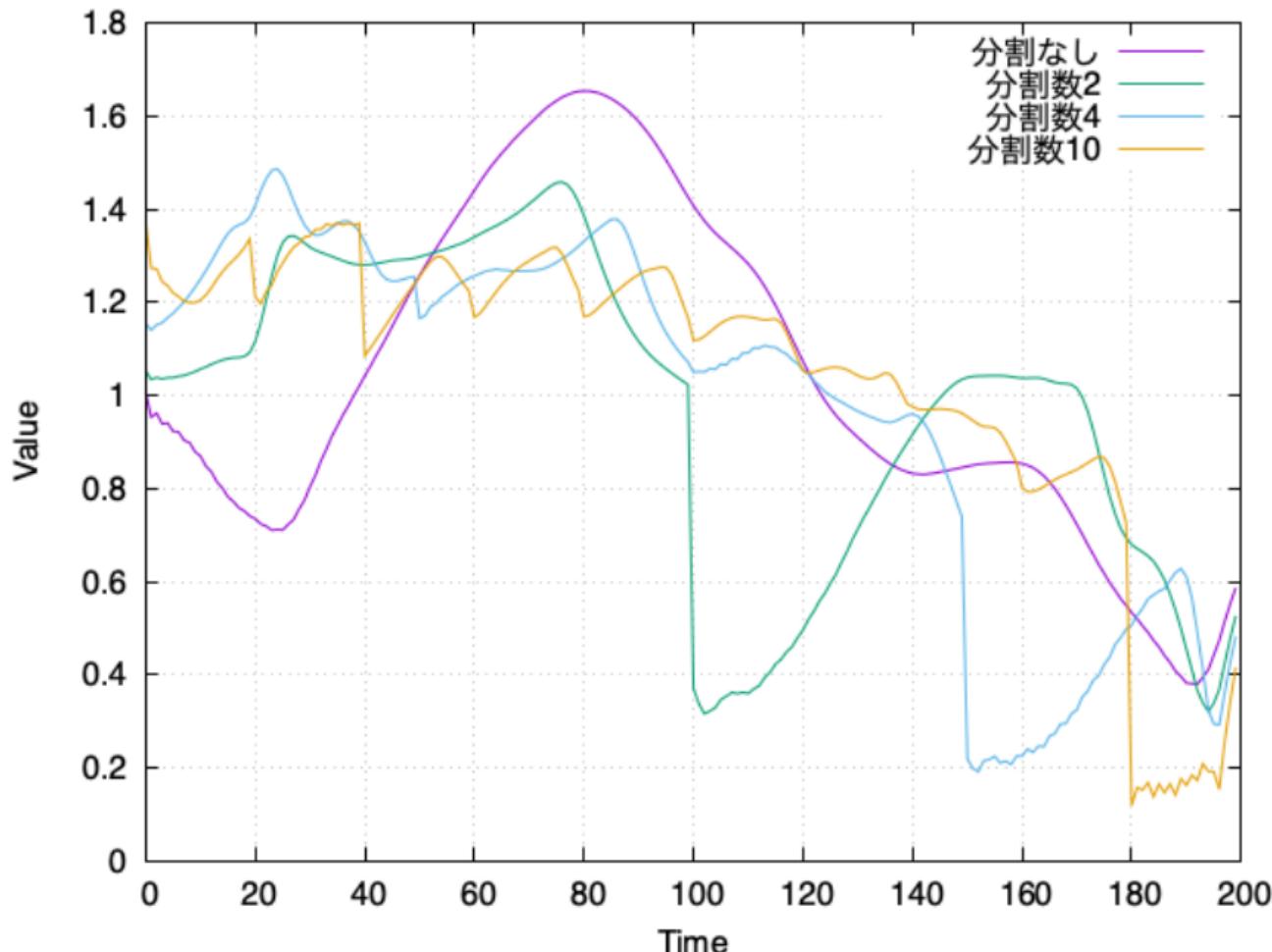
基底の累積寄与率の最小値（上）と最大値（下）

解像度	分割なし	分割数 2	分割数 4	分割数 10
64^3	0.970	0.972	0.98	0.993
		0.986	0.990	0.997
128^3	0.940	0.955	0.971	0.989
		0.965	0.976	0.996

実験結果

相対誤差

$$L_2 = \frac{\|\mathbf{u} - \tilde{\mathbf{u}}\|_2}{\|\mathbf{u}\|_2}$$

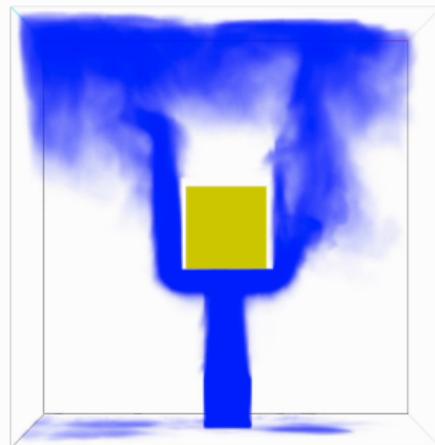


実験結果

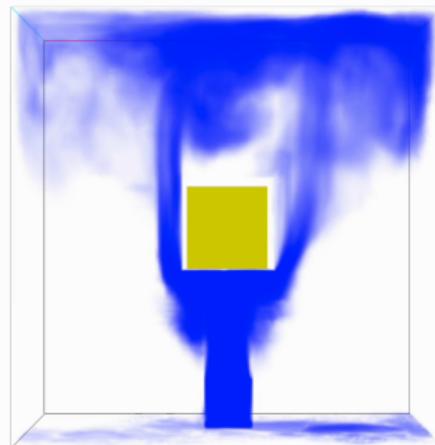
解像度 128^3 , 200 フレーム

CPU:Apple M4 Pro, メモリ:64GB

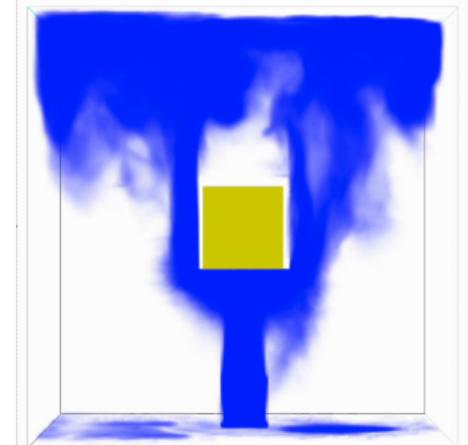
分割前後で流れが変化しているのが確認できる。



オリジナルのシミュレーション



既存手法



提案手法

おわりに

結論

- スナップショット行列の分割による手法を提案した.
- 部分空間の前処理の計算負荷の軽減に成功した.
- 流れの様子が急激に変化し、見た目に影響する場合がある.

今後の研究

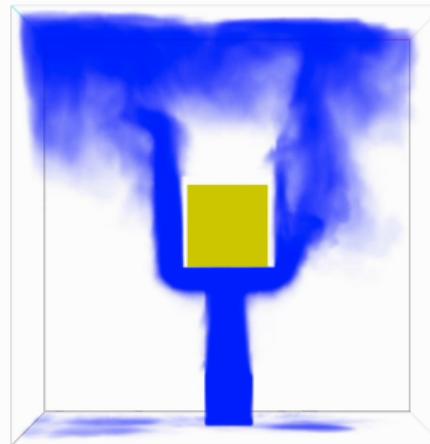
- 分割の最適化
- 分割前後の流速の変化を軽減
- 離散コサイン変換を用いた圧縮手法 [Jones et al. 2016]への適用

- [1] A. D. Jones, P. Sen, and T. Kim. Compressing fluid subspaces. *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 77–84, 2016.
- [2] A. J. Chorin. Numerical solution of the navier-stokes equations. *Mathematics of Computation*, 22(104):745–762, 1968.
- [3] A. Treuille, A. Lewis, and Z. Popović. Model reduction for real-time fluids. *ACM Transactions on Graphics*, 25(3):826–834, 2006.
- [4] J. Stam. Stable fluids. In *SIGGRAPH 99 Conference Proceedings, Annual Conference Series*, pages 121–128, 1999.
- [5] R. Fedkiew, J. Stam, and H. Jensen. Visual simulation of smoke. In *Proceedings of SIGGRAPH 01*, 15–22, 2001.

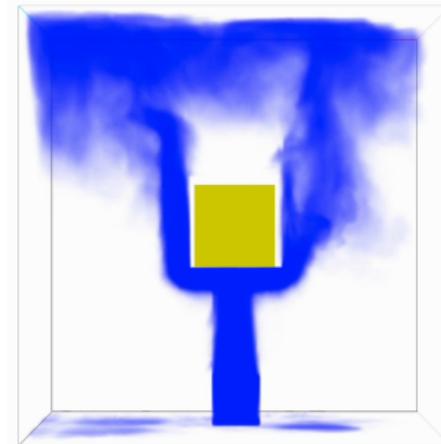
質疑応答

シミュレーションの条件変更

- タイムステップを変更
- 障害物を変更



タイムステップを変更



大きな分割数

質疑応答

その他の計算時間

- 圧力項 $p = A^{-1}b$ は反復回数 20 回, $330\text{ms}/f \rightarrow 6\mu\text{s}/f$
- 粘性項 $440\text{ms}/f \rightarrow 4\mu\text{s}/f$
- 部分空間から全次元へ射影 1.34ms