

流体シミュレーションにおける FLIP 法の粒子半径の解析

19D8102020C 須之内 俊樹

中央大学理工学部情報工学科 形状情報処理研究室

2023 年 3 月

概要

流体シミュレーションは、流体と触れる工業製品の開発や、コンピュータグラフィックスによる流体の表現に役立っている。流体の種類や運動によって条件を設けることで、それらしい動きをシミュレーションすることができる。例えば水は非圧縮性の流体であるため、体積や密度が変化しないことが望ましい。シミュレーションでは、様々なパラメータが登場する。シミュレーションの内容によって適切な値は異なる。本研究では、水が重力に従って運動するシミュレーションにおいて、水の体積の変化が小さくなるようなパラメータの解析を行う。

キーワード: 流体シミュレーション, 数値流体力学, FLIP, 非圧縮性条件

1 序論

流体シミュレーションとは、コンピュータを用いて、流体の運動を表現する数値モデルの近似解を求めることであり、数値流体力学とも呼ばれる。計算結果を可視化することで、流体の物理量の分布を全貌を把握できる。計算は流体力学の式を離散化して行う。コンピュータグラフィックスにおいては、それらしい流体の運動が高速でシミュレーションできることが重要視される。精度良く計算できることはそこまで重要視されておらず、場合によってはそれらしさのために、現実より大袈裟な表現をすることもある。

1.1 表記法

シミュレーションで計算する対象は、時刻ごとの、空間に分布する流体の物理量である。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。

- $\mathbf{u}(\mathbf{x}, t)$: 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（ここでは定数）
- \mathbf{f} : 流体にかかる外力ベクトル。（簡単のため重力のみを扱い、一定の値。）

2 ナビエ・ストークス方程式

式 (1) で表されるナビエ・ストークス方程式は流体力学の支配方程式である。圧縮されない流体を扱うときは、式 (2) を同時に考える。

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

$$\nabla \cdot \mathbf{u}(\mathbf{x}, t) = 0 \quad (2)$$

式 (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。移流項は非線形項であり、その他は線形項である。仮の速度 \mathbf{u}^* を用いて、線形項を以下の式 (3)、非線形項を式 (4) のように分解して計算する手法がよく用いられている。

$$\mathbf{u}(\mathbf{x}, t+1) = \mathbf{u}^* - \Delta t \left(\frac{1}{\rho} \nabla p(\mathbf{x}, t) + \nu \nabla^2 \mathbf{u}(\mathbf{x}, t) + \mathbf{f} \right) \quad (3)$$

$$\mathbf{u}^* = \mathbf{u}(\mathbf{x}, t) - \Delta t (\mathbf{u}(\mathbf{x}, t) \cdot \nabla) \mathbf{u}(\mathbf{x}, t) \quad (4)$$

空間に分布する物理量を表現し計算する手法は、大きく分けて格子法と粒子法がある。

2.1 格子法と粒子法

格子法は、空間を立法体の格子に区切って離散化し、格子に物理量を配置して計算する方法である。格子が規則正しく並んでいるため、微分演算を差分法などの方法で近似することができる。移流項計算の際に誤差が生まれ、移流項計算の精度が悪い。

粒子法は、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。粒子の移動によって移流項が精度よく計算できるが、粒子の影響範囲内に他の粒子がないとき、移流項以外の計算が正確にできない。

3 FLIP (Fluid Implicit Particles)

Harlow らの手法 Particle In Cell (PIC) 法 [1] は、粒子法と格子法を組み合わせた手法である。移流項の計算

を粒子法を用いて行い、その他の計算を格子法を用いて行う。物理量を格子と粒子で受け渡して行うが、受け渡す際に物理量が平滑化されてしまい、粘性が高くなる。これを人工粘性と呼ぶ。Brackbill らは PIC のアルゴリズムを、粒子の速度の変化分のみを計算し、粒子の速度に加算するものに変更した、Fluid Implicit Particles (FLIP) 法 [2] を提案した。これにより人工粘性がなくなるが、数値誤差が蓄積し続けることや、粒子の勢いがほとんど減衰しないことなどの欠点がある。

4 提案手法

水は非圧縮性の流体なので、シミュレーションの中で密度や体積が保存されることが望ましい。本研究では、立方体の内部の領域内で、ダム崩壊シミュレーションと呼ばれる、水を柱状に配置し、重力に従って崩れるシミュレーションを行う。関数値が粒子と格子の距離が 0 のとき $2/3$ 、距離が大きくなるにつれて 0 に収束するようなカーネル関数 $N(x)$ を用いて、周辺の粒子からの影響を表す重みを計算する。各粒子の重みは以下のように計算し、格子の重みは格子の付近にある粒子の重みの和である。 x_p は粒子の x 座標、 x_i は格子の x 座標を表す。

$$N(x) = \begin{cases} \frac{1}{2}|x|^3 - |x|^2 + \frac{2}{3} & (0 \leq |x| < 1), \\ \frac{1}{6}(2 - |x|)^3 & (1 \leq |x| \leq 2), \\ 0 & \text{else.} \end{cases} \quad (5)$$

$$w = N\left(\frac{1}{h}(x_p - x_i)\right) \cdot N\left(\frac{1}{h}(y_p - y_i)\right) \cdot N\left(\frac{1}{h}(z_p - z_i)\right) \quad (6)$$

重みに基づき閾値を定め、閾値以上の格子は液体として、液体の格子の体積を合計して流体の領域の体積を計算し、体積を保存できているかを確認する。閾値の等値面で、流体と気体の境界面を定める。

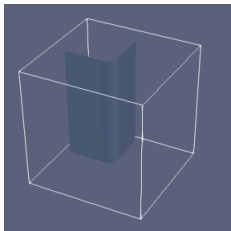


図 1: 運動開始時の様子

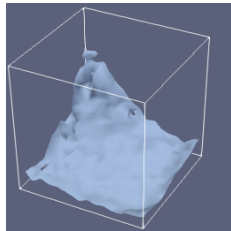


図 2: 流体の運動の様子

粒子の半径と、流体の非圧縮性を満たすように粒子位置の修正度を表す定数 γ を用いて、粒子間の距離が近すぎるときに粒子同士を離す計算をしている。また、境界面の等値面を計算するときに用いる閾値 θ が、流体の体積に大きく関わる。これら三つの定数を変えてシミュレーションを行う。シミュレーション開始時の体積

V_0 と、体積の最大値 \max と、体積の最小値 \min を用いて、どの程度の割合で体積が変化したのかを表す $(\frac{\max}{V_0}, \frac{\min}{V_0})$ を計算し、その差をとることで、体積の変動幅で体積が保存できているかを比較する。

5 実験結果と考察

実験では、変動幅が 25% 程度になることが確認できた。粒子半径や γ が小さいほど、体積の最小値が小さくなった。粒子位置の修正が小さく、非圧縮性を考慮していない速度の計算による移動の影響を強く受けていることが原因だと考えられる。体積が増加しているのは、シミュレーション開始から数フレーム後の、水の柱が崩れる間のみで、流体が拡散してしまっていることが確認できた。境界面での流体の拡散を防ぐ表面張力を考慮することができていないことが原因だと考えられる。

表 1: 半径を格子幅の 2 倍にしたシミュレーションでの、体積の変動幅の初期体積に対する割合

	$\gamma = 0.5$	$\gamma = 1$	$\gamma = 1.5$
$\theta = 0.9$	0.4519%	0.3688%	0.2550%
$\theta = 0.8$	0.5219%	0.4052%	0.2587%

6 結論

粒子半径や γ が小さいほど、粒子の密度を修正する方向に移動する距離が小さく、体積が小さくなってしまい、非圧縮条件をよく満たしているとは言えない結果が得られた、また、表面張力の影響を考慮していなかったため、水の柱が拡散してしまい、体積が増加してしまった。FLIP の抱える問題を解消した手法が生み出されているが、それらの手法も、体積や粘性に関する問題を抱えている。今後の研究では、それらの手法についての理解を深め、抱える問題点を解消する手法を提案したい。

参考文献

- [1] F.H. Harlow, (1964) The Particle-in-Cell Computing Method for Fluid Dynamics. *Methods in Computational Physics, Open Journal of Modelling and Simulation*, vol.4 no. 3, pp. 319–343, July 29, 2016.
- [2] J.U.Brackbill. D.B.KotheH.M.Ruppel. Flip: A low-dissipation, particle-in-cell method for fluid flow, *Computer Physics Communications*, vol.48 Issue 1, pp.25–38, January 1988.