

粒子法による Matlab CFD Toolbox 開発

情報科学科 小澤 旭弘

指導教員：代田 健二

1 はじめに

粒子法は、流体力学の問題を粒子の運動により離散化し、近似解を与える方法である。差分法、有限要素法などの従来法に比べると、解析対象領域の分割が必要なく、さらに粒子の運動にのみ着目していることから、大変形問題にも容易に対応可能である [1]。粒子法には MPS 法、SPH 法があり、これらは離散化の手法に違いがある。SPH 法は、カーネルと呼ばれる重み関数の重ね合わせにより物理量の場を近似し、物理量の偏微分を重み関数を元に近似する。それに対して MPS 法は、各粒子が持つ物理量の差を元に物理量の偏微分を近似計算する。SPH 法は主に圧縮性流れを陽的に計算するための手法であり、MPS 法は主に非圧縮性流れの解析をするための方法である。

本研究では、数値流体力学における粒子法の基礎研究促進を目的として、MPS 法による Matlab CFD Toolbox を開発する。Matlab[3]にはグラフィックスを表現する関数など、様々な機能が備わっており、複雑なコード生成を避けられるという利点がある。さらに並列化においては、マルチスレッドに対応した関数および Parallel Computing Toolbox[4]の関数を使うことで、自動的に並列処理可能なプログラムを作成することができる。さらに Matlab CFD Toolbox の開発によって、粒子法による数値流体シミュレーション研究の効率化に繋がることが期待できる。

2 MPS 法

MPS 法は、勾配、発散、ラプラシアンを粒子間相互作用モデルを用いて離散化し、それにより元の問題の近似解を得る手法である。勾配は、ある粒子 i とその近傍粒子 j との間の物理量の受け渡しによって、次のモデルで表現される。

$$\langle \nabla \phi \rangle_i = \frac{d}{n^0} \sum_{j \neq i} \left[\frac{\phi_j - \phi_i}{|\mathbf{r}_j - \mathbf{r}_i|^2} (\mathbf{r}_j - \mathbf{r}_i) w(|\mathbf{r}_j - \mathbf{r}_i|) \right].$$

ここで、 $\mathbf{r}_i, \mathbf{r}_j$ は粒子 i, j の位置ベクトル、 ϕ_i, ϕ_j は粒子 i, j の物理量、 n^0 は粒子数密度、 w は重み関数、 d は空間の次元数である。

また発散は、ベクトル変数 $\mathbf{u} = (u, v)$ を用いて、

$$\langle \nabla \cdot \mathbf{u} \rangle_i = \frac{d}{n^0} \sum_{j \neq i} \left[\frac{(\mathbf{u}_j - \mathbf{u}_i) \cdot (\mathbf{r}_j - \mathbf{r}_i)}{|\mathbf{r}_j - \mathbf{r}_i|^2} w(|\mathbf{r}_j - \mathbf{r}_i|) \right]$$

と表される。ここで、 $\mathbf{u}_i, \mathbf{u}_j$ は粒子 i, j のベクトル変数である。

ラプラシアンモデルは、物理量 ϕ_i の分散の増加を解析解と一致させるための係数 λ を用いて、以下のように表す。

$$\langle \nabla^2 \phi \rangle_i = \frac{2d}{\lambda n^0} \sum_{j \neq i} \left[(\phi_j - \phi_i) w(|\mathbf{r}_j - \mathbf{r}_i|) \right].$$

非圧縮性流れの数値シミュレーションを、MPS 法を用いて行う。ここで支配方程式は、次の質量保存則、運動量保存則で表される。

$$\frac{D\rho}{Dt} = 0, \quad \frac{D\mathbf{u}}{Dt} = -\frac{1}{\rho} \nabla P + \nu \nabla^2 \mathbf{u} + \mathbf{g}.$$

ここで、 \mathbf{u} は速度ベクトル、 ρ は流体の密度、 \mathbf{g} は外力 (重力加速度ベクトル)、 ν は動粘性係数、 P は圧力であり、 $\frac{D}{Dt}$ はラグランジュ微分である。MPS 法では、与えられた初期条件・境界条件の元に支配方程式を半陰的スキームにより計算する。陽的部分の計算として、時刻 k における各粒子の位置、速度、圧力が分かっているとして、新しい時刻 $k+1$ の粒子の位置、速度、圧力の値を計算し、さらに重力項と粘性項の計算、そして粒子の移動を行う。次に陰的部分の計算として、圧力ポアソン方程式の境界値問題を近似計算し、さらに、圧力勾配項の計算を行い、それにより速度と粒子位置の修正を行う。

3 数値実験と考察

空間次元数を 2、総粒子数を 2162 個、時刻幅 Δt を 0.001[s] とする。自由表面上の粒子においては圧力を 0 に固定し、壁境界には圧力計算をする内側壁粒子と圧力計算をしない外側壁粒子の 2 種類を配置する。Matlab CFD Toolbox は、先行研究 [2] の C プログラムを参考にして作成した。初期粒子配置を図 1 の通りにし、研究室計算機 (Intel Xeon E5620(Quad-Core)×2, 32GB DDR3 SDRAM) において Matlab 2014a で実行した結果は、図 2、図 3 である。計算結果は、先行研究との比較により妥当であると判断した。また、使用最大スレッド数を 1, 2, 4, 8 と変えて数値実験をした時の計算時間は、表 1 の通りである。実験結果より、使用最大スレッドの数を変えても計算時間は短縮されなかった。Toolbox 作成の際には、Matlab 関数のマルチスレッド対応を生かすべく行ったが、結果よりそれは不十分であると分かった。今後は計算時間がかかっている部分の並列化、また GPGPU の利用などを行い、Toolbox の高性能化を実現していきたい。

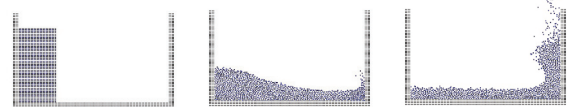


図 1 初期配置

図 2 0.4 [s]

図 3 0.8 [s]

表 1 計算時間

スレッドの数	1	2	4	8
経過時間 [s]	9631.72	9631.88	9630.45	9629.28

参考文献

- [1] 越塚誠一, 粒子法, 丸善, 2005.
- [2] 越塚誠一, 柴田和也, 室谷浩平, 粒子法入門—流体シミュレーションの基礎から並列計算と可視化まで, 丸善, 2014.
- [3] 小林一行, 最新 MATLAB ハンドブック第五版, 2014.
- [4] MathWorks, <http://jp.mathworks.com/products/parallel-computing/features.html>