

再構築ポリゴン壁を用いたSPHの壁境界計算手法

笠 晃^{1,a)}

概要: 3次元コンピュータグラフィックスにおける流体の表現では、SPHによる物理シミュレーションが使用されることが多い。SPHは流体力学の分野で開発された手法であり、粒子により流体を表現する。SPHでは壁境界も粒子で表現するが、壁粒子は離散的に配置されるため平らな斜面であっても流体粒子が滑らかに流れないという現象が見られる。そこで、本研究では壁境界をポリゴンで表現し、さらに、粒子が壁に接近したときのみ、その近傍でポリゴン壁を再構築するという方法を考案した。この手法を用いれば、流体粒子から見て斜面は完全に平らになり、粒子は滑らかに流れるようになる。また、本手法は壁境界が移動や回転をした場合でも、再計算の負荷は小さい。

1. はじめに

流体力学における物理シミュレーションの一種に粒子法というものがある。これは粒子を使用して流体を表現するものであるが質量保存が容易に実現できるため、3次元コンピュータグラフィックスにおける流体の表現で頻繁に利用されている。現在までに開発されている粒子法として、SPH (Smoothed Particle Hydrodynamics) 法 [1] と MPS (Moving Particle Semi-implicit) 法 [2] がある。ただし、MPSに比べSPHの方が計算コストが低いので、コンピュータグラフィックスで使用されるのはほとんどSPHであり、本研究でもSPHを用いている。粒子法では固体との境界を形成する壁も粒子で表現する。しかし壁粒子は離散的に配置されるため離散化誤差が発生して、斜面を流体が滑らかに流れないという現象が見られるし、壁から受ける力を正確に計算できないといった不具合も生じる。

原田ら [3] は壁境界を表現するのに粒子ではなくポリゴンメッシュを使用することを提案した。これは流体粒子からポリゴンメッシュまでの距離を用いて壁から受ける力を計算するので、壁粒子を使用したときに生ずる問題は解決する。しかし、この手法は距離をいかにして計算するかという問題を新たに導入することになる。流体粒子からポリゴンメッシュまでの距離を計算する方法の一つに距離関数の利用があり、距離関数を高精度に求める手法も開発されているが計算コストが高く、ポリゴンメッシュが移動した場合の再計算の負荷が大きくなる。これに対し、原田ら [4]

は離散境界を用いて距離関数を計算する手法を提案している。この手法は従来の手法に比べ計算コストが低くなるが、離散化誤差が混入し、流体粒子が滑らかに流れないという現象を完全に除去することができない。また、依然として距離関数の値を3次元的に求める必要があるため、再計算の負荷は決して小さくない。そこで、我々はポリゴンメッシュの代わりに離散境界を用い、これにポリゴンメッシュの情報を持たせるという手法を新たに開発した。こうすることにより、ポリゴンメッシュの情報を高速に引き出し、流体粒子からポリゴンメッシュまでの距離を精度よく求めることが可能になる。また、この手法は壁境界が移動した場合でも、離散境界が持つ情報を更新するだけでよく、再計算のコストは低い。

2. 従来の研究

最初、SPHは天体物理学の分野で開発されたが、その後、圧縮率を小さくすることにより水などの液体を扱うことが可能になった。ただし、その際、壁境界も粒子で表現し、流体粒子が壁粒子から圧力や粘性力などを受けるものとして計算する [5], [6] ため離散化誤差が生ずる。そして、この離散化誤差のため、平らな斜面の途中で流体粒子が捕捉されて流体が滑らかに流れなくなるし、壁から受ける粘性力や圧力などが正確に計算できないといった問題も発生する。

原田ら [3] は壁粒子の代わりにポリゴンメッシュを用い、さらにポリゴンメッシュに対する距離関数を使用することで、このような問題を解決することを試みている。距離関数とは、空間内の任意の点から境界までのユークリッド距離を表すもので、符号付き距離関数の絶対値を取ったもの

¹ 福岡工業大学情報工学部
Faculty of Information Engineering, Fukuoka Institute of Technology

a) ryu@fit.ac.jp

として定義される。さらに、符号付き距離関数を ϕ で表せば、これは次のアイコナル方程式の解である。

$$|\nabla\phi| = 1, \quad \phi|_S = 0 \quad (1)$$

ここに S は境界表面を表している。原田らは、壁から受ける力を距離関数の関数として定義した。このため、理想的な距離関数であれば平面壁は粒子から見て完全な平面となり、平らな斜面で流体粒子が滑らかに流れないなどの問題は発生しない。

ところで、距離関数や符号付き距離関数は3次元空間を格子で分割して格子点においてのみ計算しておき、格子点以外の点の値は補間によって求める。各格子点に対し、境界面までの距離を計算することを距離変換 (distance transform) といい、境界面上の最近点を計算することを最近点変換 (closest point transform) という。格子点以外の点で距離関数を計算するとき、各格子点における距離関数の値を使用するよりも最近点の値を使用の方がより正確な値が得られる。

さて、ポリゴンメッシュの境界面に対して距離変換と最近点変換を求める手法として、さまざまなアルゴリズム [7], [8], [9] が提案されているが、これらはすべて多数の面を構築しなければならない。すなわち、各ポリゴンをその法線方向に押し出すことにより新しい面を生成し、これによって3次元空間を多数の領域に分割する必要がある。このため、高精度の距離関数は得られるものの多量の計算時間を消費する。これらの手法に対し、原田ら [4] は離散境界を使用する方法を提案している。すなわちポリゴンメッシュより離散境界を作成し (5.1 節を参照のこと)、各格子点に対し離散境界までの最小距離を計算することにより、距離変換と最近点変換を求めるのである。この手法は、ポリゴンを押し出す手法に比べ高速であるが、境界面を離散化するため離散化誤差の混入は避けられず、高精度の距離関数を得るのは困難である。このため、平らな斜面で流体粒子が滑らかに流れないという問題は依然として未解決であり、流体粒子が壁から受ける粘性力や圧力などにも誤差が混入する。

3次元コンピュータグラフィックスの観点より見たとき、流体粒子が壁から受ける力に混入する誤差は無視できるとしても、平らな斜面に粒子が停留するのは望ましくない。また、アニメーションにより境界面が移動した場合、距離関数を使用する方法は距離変換と最近点変換を再計算する必要がある。この処理は3次元空間内の各格子点に対して行わなければならないため計算コストが高く、たとえ離散境界を使用したとしても再計算のコストは決して小さくない。

3. SPH について

流体力学における支配方程式を数値的に解く場合、空間

を格子で分割して差分法を使用する方法と、粒子を使用する方法がある。粒子を使用する方法は粒子法と呼ばれ、SPH も粒子法の一つである。

3.1 非圧縮性流体の支配方程式

水などの非圧縮性流体に対する支配方程式は連続の式とナビエ・ストークス方程式であり、これらは次のように記述される。

$$\frac{\partial\rho}{\partial t} + \nabla \cdot (\rho\mathbf{v}) = 0 \quad (2)$$

$$\rho \frac{D\mathbf{v}}{Dt} = -\nabla p + \mu \nabla^2 \mathbf{v} + \rho \mathbf{g} \quad (3)$$

ただし、 ρ , \mathbf{v} , p , μ は、それぞれ流体の密度、速度、圧力、粘性率を表しており、 \mathbf{g} は重力加速度である。

3.2 カーネル近似と粒子近似

SPH の定式化はしばしば2段階に分割される。最初のステップは場の関数のカーネル近似である。場の関数を $A(\mathbf{r})$ で表すことにすると、 $A(\mathbf{r})$ のカーネル近似は次のように記述される。

$$\langle A(\mathbf{r}) \rangle = \int_{\Omega} A(\mathbf{r}') W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}' \quad (4)$$

ここに、 $W(\mathbf{r}, h)$ は平滑化カーネル関数、カーネル関数あるいは重み関数などと呼ばれるものである。 h は一般にカーネル半径と呼ばれ、カーネル関数はこの半径を持つ球の外側で0となる性質を持っている。したがって、積分領域 Ω も点 \mathbf{r} を中心とする球である。

定式化の次のステップは粒子近似である。すなわち、カーネル近似された場の関数を離散化し、有限個の粒子によって近似する。各粒子の位置を \mathbf{r}_i 、質量を m_i とし、さらに粒子の持つ関数の値を A_i とするとき、空間内の点 \mathbf{r}_i における平滑化された関数の値は次の式で与えられる

$$\langle A_i \rangle = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h) \quad (5)$$

上の式において和は \mathbf{r}_i から h 以内の距離にある粒子に対してのみ取ればよく、いわゆる近傍粒子探索を利用することが可能である。また、以下では平滑化された関数の括弧は省いて記述することにする。

3.3 ナビエ・ストークス方程式の SPH による近似

式 (5) を用いれば、 \mathbf{r}_i における密度 ρ_i は次の式で計算される。

$$\rho_i = \sum_j m_j W(\mathbf{r}_i - \mathbf{r}_j, h) \quad (6)$$

さらに、式 (3) の右辺の第1項と第2項はそれぞれ次のように離散化される。ただし、 i と j が対称になるように若干の補正がなされている。

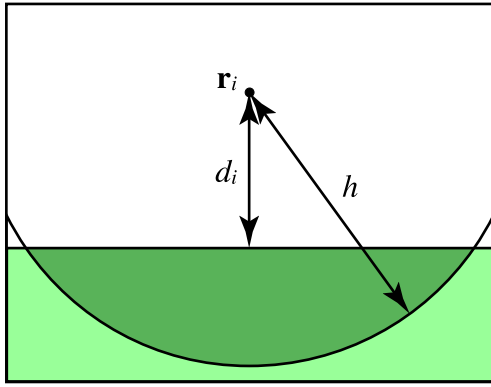


図 1 壁の積分体積

Fig. 1 Integral volume of the wall.

$$\mathbf{f}_i^p \equiv -\nabla p(\mathbf{r}_i) = -\sum_j m_j \frac{p_i + p_j}{2\rho_j} \nabla W(\mathbf{r}_i - \mathbf{r}_j, h) \quad (7)$$

$$\mathbf{f}_i^v \equiv \mu \nabla^2 \mathbf{v}(\mathbf{r}_i) = \mu \sum_j m_j \frac{\mathbf{v}_j - \mathbf{v}_i}{\rho_j} \nabla^2 W(\mathbf{r}_i - \mathbf{r}_j, h) \quad (8)$$

なお、一般に式 (7) は圧力項と呼ばれ、式 (8) は粘性項と呼ばれている。

カーネル関数は式によって最適なものを使用するのが普通である。たとえば、式 (6) に対するカーネル関数は次のようになる。

$$W_\rho(\mathbf{r}, h) = \frac{315}{64\pi h^9} \begin{cases} (h^2 - r^2)^3 & 0 \leq r \leq h \\ 0 & h < r \end{cases} \quad (9)$$

以下では $W_v(\mathbf{r}, h)$ も使用しているが、これは式 (8) に対するカーネル関数である。

4. 壁重み関数

原田らは粒子が壁から受ける力を距離関数の関数として求めるために、密度などの計算を流体粒子からの寄与分と壁粒子からの寄与分とに分割している [3]。本研究でも、このような分割を使用するが、壁粒子からの寄与分を壁からの寄与分へと変更する。すなわち、壁については SPH の定式化のカーネル近似のみを用い、粒子近似を使用しない。原田らは、壁粒子からの寄与分を壁粒子を発生させ数値計算によって求めているが、粒子近似を使用しなければ壁からの寄与分を解析的に求めることが可能になる。なお、ここではスペースの都合で圧力項の計算に関する議論を省略している。

4.1 密度の計算

点 \mathbf{r}_i における密度は式 (6) によって計算されるが、ここではこれを流体粒子からの寄与分と壁からの寄与分に分割し、次のように変形する。

$$\rho_i = \sum_{j \in \text{fluid}} m_j W_\rho(\mathbf{r}_i - \mathbf{r}_j, h) + \rho_0 \int_{\Omega_w} W_\rho(\mathbf{r}_i - \mathbf{r}', h) d\mathbf{r}' \quad (10)$$

右辺の第 2 項が壁からの寄与分であり、 ρ_0 は壁の密度を

表している。右辺第 2 項は容易に計算できて、 α_i の関数となる。ただし、 $\alpha_i \equiv d_i/h$ であり、 d_i は点 \mathbf{r}_i から壁までの距離である。すなわち、 α_i は d_i を h に関して正規化したものである。この様子を図 1 に示す。また右辺第 2 項を $F_\rho^w(\alpha_i)$ で表すことにすると、これは次のようになる。

$$F_\rho^w(\alpha_i) = \frac{\rho_0}{256} (1 - \alpha_i)^5 \times (35\alpha_i^4 + 175\alpha_i^3 + 345\alpha_i^2 + 325\alpha_i + 128) \quad (11)$$

これが壁重み関数 [3] と呼ばれるものである。この壁重み関数を用いれば、式 (10) は次のように単純化される。

$$\rho_i = \sum_{j \in \text{fluid}} m_j W(\mathbf{r}_i - \mathbf{r}_j, h) + F_\rho^w(\alpha_i) \quad (12)$$

なお、式 (11) は解析的に求まっているとはいえ複雑である。そこで、実装に当たっては区間 $[0, 1]$ を有限個の点で等間隔に分割し、各点における $F_\rho^w(\alpha_i)$ の値をあらかじめ求めておく。そして、流体計算において関数の値が必要なときは補間により求めることにする。これは、粘性項や圧力項の壁重み関数の計算においても同様である。

4.2 粘性項の計算

密度と同様、粘性項の計算も流体粒子からの寄与分 $\mathbf{f}_{i,\text{fluid}}^v$ と壁からの寄与分 $\mathbf{f}_{i,\text{wall}}^v$ に分割する。すなわち、粘性項は式 (8) によって計算されるが、これを次のように変形する。

$$\mathbf{f}_i^v = \mathbf{f}_{i,\text{fluid}}^v + \mathbf{f}_{i,\text{wall}}^v \quad (13)$$

ただし、壁からの寄与分は次のような式で表される。

$$\mathbf{f}_{i,\text{wall}}^v = \mu \int_{\Omega_w} (\mathbf{v}(\mathbf{r}') - \mathbf{v}_i) \nabla^2 W_v(\mathbf{r}_i - \mathbf{r}', h) d\mathbf{r}' \quad (14)$$

原田らは壁の速度をゼロにしているが、これでは壁が移動しているときに誤差が生ずる。我々は壁の速度を離散境界の速度で近似しており、これを $\hat{\mathbf{v}}$ で表せば、式 (14) は次のようになる。詳細は次章で議論する。

$$\mathbf{f}_{i,\text{wall}}^v = \mu (\hat{\mathbf{v}} - \mathbf{v}_i) \int_{\Omega_w} \nabla^2 W_v(\mathbf{r}_i - \mathbf{r}', h) d\mathbf{r}' \quad (15)$$

この式を積分することにより、次の結果を得る。

$$F_v^w(\alpha_i) = \frac{15\mu}{2h^2} (1 + \alpha_i)(1 - \alpha_i)^3 \quad (16)$$

$$\mathbf{f}_{i,\text{wall}}^v = (\hat{\mathbf{v}} - \mathbf{v}_i) F_v^w(\alpha_i) \quad (17)$$

ここに、 $F_v^w(\alpha_i)$ は粘性項に対する壁重み関数である。

5. ポリゴン壁の再構築

ここでは、離散境界を用いてポリゴン壁を再構築する手法について述べる。ポリゴン壁が再構築できたならば、これを使用して壁重み関数を計算することが可能となり、結果としてナビエストークス方程式の密度、粘性項および圧力項の値が求まる。これらの値はポリゴン壁を使用して計算したものであるため精度が高く、平らな斜面で流体粒子が滑らかに流れないという問題も発生しない。

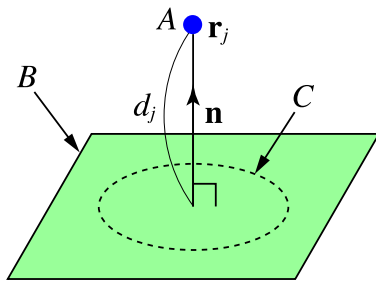


図 2 離散境界の情報と平面の再構築

Fig. 2 Information about the discrete boundary and reconstruction of the plane.

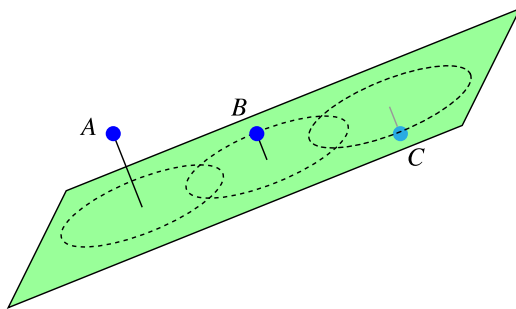


図 3 離散境界の複数の点による平面の再構築

Fig. 3 Reconstruction of the plane using multiple points in the discrete boundary.

5.1 離散境界の作成

離散境界とは、ポリゴンメッシュの境界面を点によって離散化したものであり、ポリゴンメッシュをボクセル化し、各ボクセルの中心に点を作成することにより構築される。ポリゴンメッシュのボクセル化は、空間を格子で分割し、ポリゴンメッシュと交差する格子セルをボクセルとすることにより実行される。

5.2 離散境界による壁の再構築

ポリゴンを離散境界に変換する際、離散境界に元のポリゴンの情報を持たせることが可能である。図 2 において A は離散境界を構成する点の 1 つを表し、B はポリゴン平面を表している。ここで、平面 B の単位法線ベクトルを \mathbf{n} とし、点 A の位置ベクトルを \mathbf{r}_j 、点 A の平面 B に関する符号付き距離を d_j で表すことにし、三つ組 $\langle \mathbf{n}, \mathbf{r}_j, d_j \rangle$ を点 A の持つ情報として記憶しておく。すると、これらの情報よりポリゴン平面 B の方程式が次のように得られる。

$$(\mathbf{r} - \mathbf{r}_j) \cdot \mathbf{n} + d_j = 0 \quad (18)$$

これは、点 A の持つ三つ組情報からポリゴン平面を局所的に再構築可能なことを意味している。図 2 の C が局所的に再構築されたポリゴン平面または壁を表している。また、図 3 は離散境界を構成する複数の点によって再構築されたポリゴン壁を示している。

ナビエストークス方程式に関する壁重み関数は、この

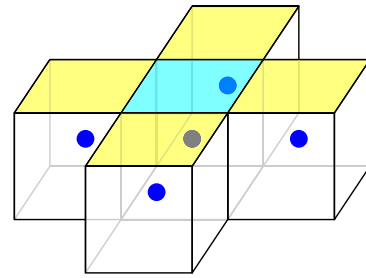


図 4 離散境界の点の最も単純な配置パターン

Fig. 4 The simplest configuration pattern of points in discrete boundary.

再構築されたポリゴン壁を用いて計算する。すなわち、壁重み関数はすべて流体粒子から壁までの距離 d_i によって計算されるが、ここでは離散境界のうち流体粒子に最も近い点を求め、この点の三つ組情報 $\langle \mathbf{n}, \mathbf{r}_j, d_j \rangle$ を用いて次の式により d_i を計算することにする。

$$d_i = (\mathbf{r}_i - \mathbf{r}_j) \cdot \mathbf{n} + d_j \quad (19)$$

ただし、 \mathbf{r}_i は流体粒子の位置座標である。なお、ポリゴン壁が移動や回転をしたときは離散境界の持つ情報のうち \mathbf{n} と \mathbf{r}_j を再計算する必要がある。しかしながら、この計算は行列計算であり、しかも、距離関数の計算が 3 次元空間内の格子点に対して実行しなければならないのに対し、離散境界はポリゴンオブジェクトの表面付近にしか存在しないので高速な計算が可能である。

5.3 離散境界探索の範囲

流体粒子に関する壁重み関数を計算するとき、離散境界を構成する点のうち流体粒子に最も近いものを求めなければならないが、流体粒子に関しどの範囲を探索すれば十分であろうか。ここで鍵となるのが、流体粒子から境界壁までの距離が h 以上のとき壁関数がゼロになるという事実である。したがって、壁境界から h 以内の距離にある流体粒子から離散境界までの最短距離の最大値を l_{\max} とすれば、各流体粒子に対し半径 l_{\max} の球の内部を探索すればよいことになる。ただし、 l_{\max} は離散境界を構成する点の配置パターンによって変化するので、配置パターン P に対する l_{\max} を l_{\max}^P と表記することにする。なお、配置パターンが決定しても壁境界のポリゴン平面の位置によって l_{\max} は変化するので、 l_{\max}^P を求めるには、ポリゴン平面を動かして最大の l_{\max} を求める必要がある。

たとえば、最も単純な例として離散境界の点が図 4 に示すような配置パターン P_1 の場合を考察しよう。図内に描画されている立方体は、離散境界を作成したときに利用したボクセルである。ただし、着色された面は、この面を介して隣接するボクセルが存在しないことを意味している。さらに、着色された面のうちシアン色の面は、この面を持つボクセル内の点に現在注目しているということを表して

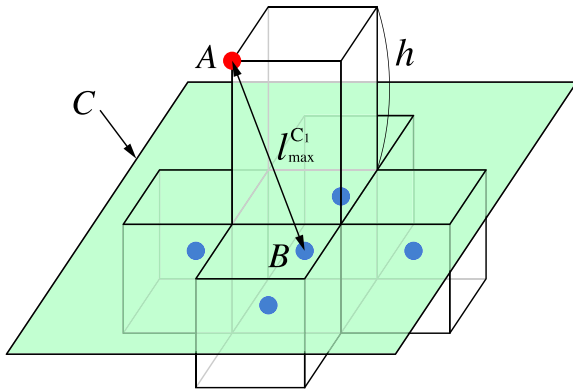


図 5 l_{\max} が最大になる壁境界の位置

Fig. 5 The wall boundary's position where l_{\max} is maximum.

いる。

さて、壁境界のポリゴン平面は図 4 のすべてのボクセルと交差するが、 l_{\max} が最大になるのは、壁境界のポリゴン平面と着色された面の距離が無限小のときである。このときの様子を図 5 に示す。この図において、A は流体粒子を、B は現在注目している離散境界の点を、C は壁境界をそれぞれ表している。また、図 4 のシアンに着色された面に関し、図 5 では、この面の上に直方体が配置されているが、これは壁境界から h 以内の距離にある流体粒子の内、離散境界点 B に最も近いものの存在範囲を表している。したがって、流体粒子が A の位置にあるときの A と B の間の距離が $l_{\max}^{P_1}$ になる。この値を求めるのは容易であり、次のように表される。

$$l_{\max}^{P_1} = \sqrt{4h^2 + 4ah + 3a^2}/2 \quad (20)$$

ただし、 a は離散境界を作成したときに利用したボクセルの 1 辺の長さである。

詳細は別の機会に譲るが、離散境界の点の配置パターンは全部で 10 種類あり、それぞれのパターンについて $l_{\max}^{P_1}$ を求めた。これらの最大値 L_{\max} を計算すれば、これが離散境界探索範囲を表す球の半径である。すなわち、次のような式で表される。

$$L_{\max} = \max_i l_{\max}^{P_i} = \sqrt{6h^2 + 5a^2}/2 \quad (21)$$

6. 実験

平らな斜面を用いた実験と水柱崩壊実験を実施した。両実験において、液体の密度は水と同じ 1000 kg/m^3 を使用し、表面張力はないものとした。また、重力加速度として 9.8 m/s^2 を用いた。実験に使用したコンピュータは Apple Mac Pro 2013 (クロック周波数 3.5GHz) で、32GBByte の主記憶を搭載している。

6.1 平らな斜面を用いた実験

15 度の傾斜を持つ平らな斜面に立方体状の流体粒子を配置し、重力により崩壊させた後、斜面を滑り落とさせるという実験を行った。立方体の 1 辺の長さは 4cm とした。実験結果を図 6 に示しているが、(a) が流体粒子の初期配置である。また、(b) は境界壁として 3 層粒子壁を用いた場合の 0.9sec 後の流体粒子の様子を表しているが、粒子が斜面の途中で停滞しているのが観測される。これは、時間がさらに経過しても状況は変化しなかった。これに対し、(c) は再構築されたポリゴン壁を用いた場合の 0.9sec 後の流体粒子の様子を表している。粒子が斜面に沿って滑らかに流れ落ちているのがよく分かる。

6.2 水柱崩壊実験

水槽内に直方体状の流体粒子を配置し、重力により崩壊させた。これは計算流体力学における標準実験の 1 つであり、水柱崩壊実験やダム崩壊実験と呼ばれている。水槽の大きさを、幅 × 高さ × 奥行きが $40 \text{ cm} \times 30 \text{ cm} \times 20 \text{ cm}$ となるようにし、水柱の大きさを $12 \text{ cm} \times 20 \text{ cm} \times 20 \text{ cm}$ とした。

実験結果を図 7 に示すが、(a) は流体粒子の初期配置を表している。(b) と (c) はそれぞれ境界壁として 3 層粒子壁および再構築されたポリゴン壁を用いた場合の崩壊後の流体粒子の様子であるが、どちらの場合も右側の壁に衝突した粒子が全体的に最も高い位置に来る瞬間として物理時間 $t = 0.467 \text{ sec}$ を選択した。(b) の粒子が右側の壁より離反したのち散開しているのに対し、(c) の粒子は右側の壁に沿って高く上昇している。実際の水を用いた実験では、水が右側の壁に衝突した後、壁に沿って高く上昇するので、(c) の方がより実際の水に近い挙動を再現していると考えられる。そして、これは 3 層粒子壁よりも再構築されたポリゴン壁の方が、壁の近傍において流体粒子により適切な力が働いているということを示唆している。

さらに、物理時間 $t = 0.467 \text{ sec}$ に達するまでに要する処理時間を 5 回計測し、平均値を算出した。3 層粒子壁の場合は 2122 ステップ 457.2sec であり、再構築されたポリゴン壁の場合は 1997 ステップ 214.2sec であった。したがって、1 ステップ当たりの処理時間は、3 層粒子壁の場合が 0.215sec、再構築されたポリゴン壁の場合が 0.107sec ということになる。すなわち、この実験に限って言えば、再構築されたポリゴン壁を使用したとき、3 層粒子壁を使用したときの 2.01 倍の処理速度になっている。

7. おわりに

離散境界を使用してポリゴン境界を再構築する手法について述べた。境界として 3 層粒子壁を使用した場合、流体粒子が平らな斜面の途中で停留することがあるが、再構築されたポリゴン壁ではそのようなことは起こらない。また、水槽を用いたダム崩壊実験でも、流体粒子の動きは 3

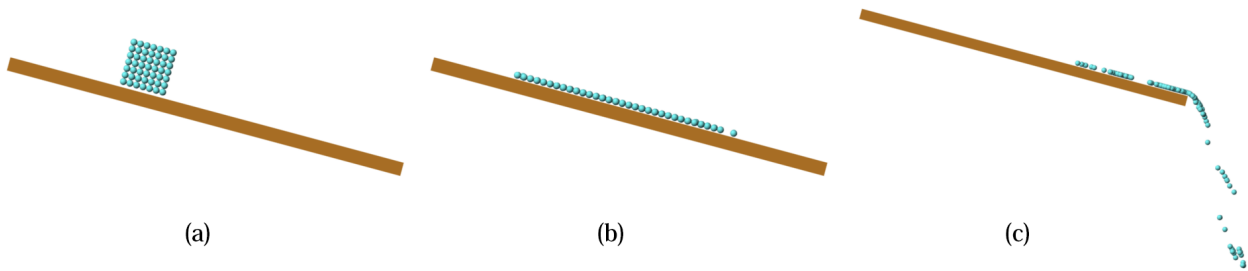


図 6 平らな斜面を用いた実験 (a) 初期配置 (b) 3層粒子壁 (c) 再構築壁

Fig. 6 Experiments using a flat slop. (a)initial configuration (b)3 layered particle wall (c)reconstructed wall

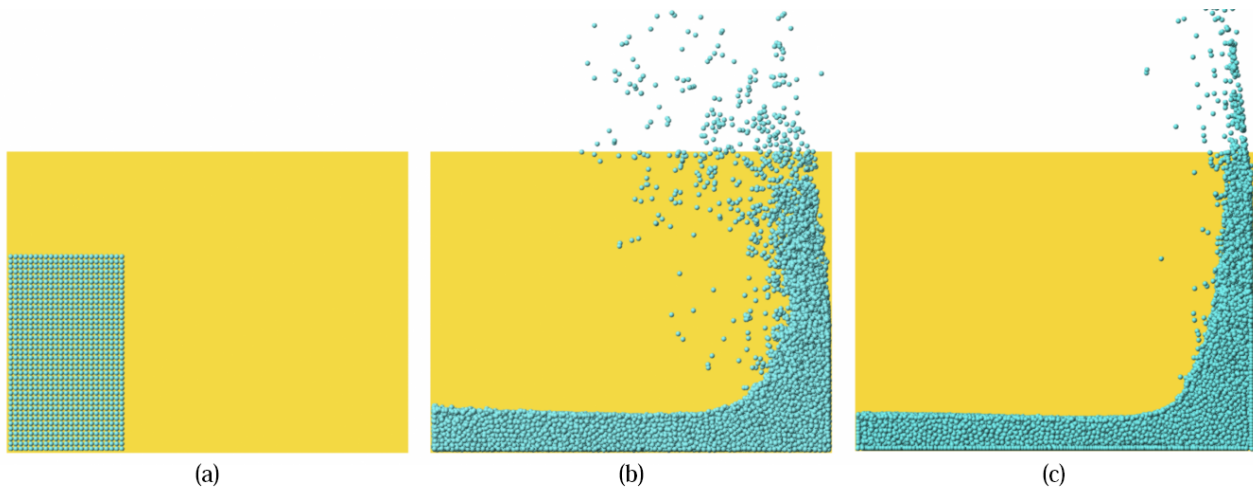


図 7 水柱崩壊実験 (a) 初期配置 (b) 3層粒子壁 (c) 再構築壁

Fig. 7 Water column collapse experiments. (a)initial configuration (b)3 layered particle wall (c)reconstructed wall

層粒子壁よりも再構築されたポリゴン壁の方が実際の実験をよく再現していた。

さらに、本手法を用いる利点として、壁境界が移動した場合の処理コストが小さいことが挙げられる。すなわち、壁境界が移動した場合でも離散境界の位置ベクトルに対して移動変換と回転変換を適用し、各離散境界が持つ平面の単位法線ベクトルに対して回転変換を適用するだけでよく、簡単な処理で対応可能である。

今後の改良点として複数個の CPU や GPU を用いた並列処理が考えられる。特に GPU では 3 次元幾何変換が高速に実行可能なので、壁境界が移動したときの移動変換や回転変換を短時間で実行できるものと思われる。

参考文献

[1] Monaghan, J.: An introduction to SPH, *Computer Physics Communications*, Vol. 48, No. 1, pp. 89–96 (1988).
[2] Koshizuka, S. and Oka, Y.: Moving-particle semi-implicit method for fragmentation of incompressible fluid, *Nuclear Science and Engineering*, Vol. 123, No. 3, pp. 421–434 (1996).

[3] 原田隆宏, 越塚誠一: SPH における壁境界計算手法の改良, *情報処理学会論文誌*, Vol. 48, No. 4, pp. 1838–1846 (2007).
[4] 原田隆宏, 越塚誠一: 離散境界を用いた距離関数の構築手法, *情報処理学会論文誌*, Vol. 48, No. 4, pp. 1820–1828 (2007).
[5] Monaghan, J. J.: Simulating Free Surface Flow with SPH, *Journal of Computational Physics*, Vol. 110, pp. 399–406 (1994).
[6] Libersky, L. D., Petscheck, A. G., Carney, T. C., Hipp, J. R. and Allahdadi, F.: High Strain Lagrangian Hydrodynamics: A Three Dimensional SPH Code for Dynamic Material Response, *Journal of Computational Physics*, Vol. 109, pp. 67–75 (1993).
[7] Mauch, S.: A Fast Algorithm for Computing the Closest Point and Distance Transform, *Technical Report, California Institute of Technology* (2000).
[8] Sigg, C., Peikert, R. and Gross, M.: Signed Distance Transform using Graphics Hardware, *Proc. IEEE Visualization*, pp. 83–90 (2003).
[9] Sud, A., Govindaraju, N., Gayle, R. and Manocha, D.: Interactive 3D Distance Field Computation using Linear Factorization, *Proc. ACM symposium on Interactive 3D graphics and games*, pp. 117–124 (2006).