

# 曲率不変条件による陰関数曲面のメッシュベース移流法

## Velocity Calculation of an Implicit Surface by Curvature Invariance

藤澤誠<sup>†</sup> 萬立洋次郎<sup>‡</sup> 三浦憲二郎<sup>‡</sup>

Makoto Fujisawa<sup>†</sup>, Yojiro Mandachi<sup>‡</sup> and Kenjiro T. Miura<sup>‡</sup>

<sup>†</sup> 筑波大学 <sup>†</sup> University of Tsukuba <sup>‡</sup> 静岡大学 <sup>‡</sup> Shizuoka University

E-mail: [†fujis@slis.tsukuba.ac.jp](mailto:†fujis@slis.tsukuba.ac.jp), [‡{f0810138,tmkmiur}@ipc.shizuoka.ac.jp](mailto:‡{f0810138,tmkmiur}@ipc.shizuoka.ac.jp)

## 1 はじめに

コンピュータグラフィックスでは物体の境界面を描画に用いることが多く、移動・変形する物体表面の追跡は非常に重要なテーマである。特に表面が大きく変形する流体や粘弾性体のシミュレーションでは流体形状を表すために表面を抽出、追跡することは必須のタスクである。これまでの研究では、レベルセット関数などの陰関数形式で表された表面を Marching Cubes 法などで毎ステップ線形区分近似することが一般的であった。この方法ではステップ間の表面の関係性を規定することが難しく、表面のテクスチャや色を移流させることが難しい。この問題を解決するために、メッシュベースの表面追跡手法が研究されている。

メッシュベース法ではシミュレーションの最初に作成したメッシュを表面変化に追従する形で移流させる。メッシュベース法で問題となるのは、1) 表面移流方法、2) 表面トポロジ変化 である。2) に関してはトポロジ変化を検出、修正する手法が多く提案されている。一方で1) に関してはほとんどの場合、流体シミュレーションで得られた速度場、もしくは陰関数場の法線方向速度のどちらかが利用されているだけである。特に陰関数場のみしかわからない場合、法線方向にしか移流しないために表面のテクスチャや色の移流を正確に表現できず、また、メッシュの変形も大きくなり、細分割などのコストも大きくなる。Stam と Schmidt[12] は接線方向速度も考慮したメッシュ移流法を提案した。彼らは陰関数場の勾配 (法線方向) が保たれると仮定することで、接線方向速度を近似した Total Velocity を算出し、メッシュ移流に利用した。しかし、この方法では平行移動は考慮できるが、回転運動に対しては対応できない。本研究では平行移動とともに回転を局所的に把握するために、曲率不変条件を利用した移流速度算出法を提案し、回転や変形を含むシーンで有効に働くことを示す。

## 2 関連研究

水のような液体の振る舞いをコンピュータアニメーションで表現するために、移動・変形する液体表面を追跡する方法が多く提案されている。もっとも代表的なものとして、レベルセット法 [11, 5] があげられる。レベルセット法では液体表面でゼロとなる符号付距離場を移流させることで境界の場所を追跡する。Enright ら [4] は距離場だけでなく、パーティクルも同時に移流させるパーティクルレベルセット法を提案した。パーティクルを使って距離場を修正することで体積保存性を向上させた。Bargteil ら [1] はメッシュとレベルセット関数を組み合わせることで、体積損失を抑えた SLC 法を開発した。レベルセット法における表面移流を改良する方法として、このほかにも CLSVOF 法 [9]、BFEC[6]、USCIP 法 [7] などが提案されている。しかし、これらの方法では、実際にレンダリングする際にはレイトレーシングなどで陰関数場を直接レンダリングするか、Marching Cubes 法 [8] などで液体表面を近似する三角形ポリゴンを抽出する必要があり、表面のテクスチャや色の移流のために特別な処理を行わなければならない。

メッシュベース表面追跡法では、陰関数ではなくメッシュそのものを移流させる。そのときに問題となるのがメッシュのトポロジ変化である。Brochu と Bridson[3] はメッシュの自己交差情報を用いて、一貫性を持つ構造に再構築するアイデアを提案した。また、Marching Cubes 法でメッシュを作成する際にはグリッドを用いる。Wojtan ら [14] はこのグリッドを使ってトポロジ変化を局所的に検出、修正した。Müller[10] は Wojtan らの方法を拡張し、メッシュの自己交差にも対応し、グリッドの大きさよりも薄い特徴も捉えた。また、表面上の細かな特徴を捉えるために表面流を利用する方法 [15] も提案されている。

メッシュのトポロジ変化に対してはこのように多くの手法が開発されている一方、メッシュを移流させる際の速度

場に関する研究はほとんどなく、多くの場合シミュレーションで得られた速度場が陰関数場の法線方向速度が用いられている。正確な表面速度が算出できれば、トポロジの変化も少なくすみ、流体表面の細かな特徴も保存される。Stam と Schmidt[12] は接線方向速度も考慮した速度場として Total Velocity を提案した。陰関数場に対してタイムステップの始めと終わりでの法線方向が保存されるという仮定の下で、接線方向成分も含めた表面速度を算出し、メッシュベース法に用いた。しかし、法線方向を用いた仮定では平行移動には対応できても、回転には対応できない。我々は Stam と Schmidt の方法を改良し、回転にも対応させる。

### 3 陰関数の移流

#### 3.1 法線方向速度

時間に依存する陰関数  $f = f(\mathbf{x}, t)$  を用いて曲面を次式で定義する。

$$\Gamma(t) = \{\mathbf{x} | f(\mathbf{x}, t) = 0\} \quad (1)$$

$f$  が 0 となる位置を表面とする。関数  $f$  を時間で全微分し、変形することで以下の式が得られる。

$$f_x x_t + f_y y_t + f_z z_t + f_t = 0 \quad (2)$$

この式より、陰関数で表された表面 ( $f = 0$ ) の時間変化は、陰関数の勾配  $\nabla f$  と表面速度  $\dot{\mathbf{x}} = (x_t, y_t, z_t)$  により決定されることがわかる。逆に陰関数値の変化が既知であり、そこから表面速度を算出することを考える。 $\mathbf{q} = (f_x, f_y, f_z)$  とすると式 (2) を以下のように変形できる。

$$\mathbf{q}^T \dot{\mathbf{x}} = -f_t \quad (3)$$

疑似逆行列  $\mathbf{q}^{-1} = \mathbf{q}^T / (\mathbf{q}^T \mathbf{q})$  を用いて  $\dot{\mathbf{x}}$  について解くと、

$$\dot{\mathbf{x}} = \frac{-f_t}{\mathbf{q}^T \mathbf{q}} \mathbf{q} = v_n \mathbf{n} \quad (4)$$

ここで、 $\mathbf{n} = \mathbf{q} / |\mathbf{q}|$ ,  $v_n = -f_t / |\mathbf{q}|$  である。上式から  $f = 0$  での表面速度  $\dot{\mathbf{x}}$  は陰関数  $f$  の法線方向にのみ働くということがわかる。つまり、接線方向成分  $v_t$  はゼロとなる。確かに表面形状の変化のみを考えた場合、 $v_t$  は必要でない。しかし、表面上の色やテクスチャの移流を表現するには法線方向速度のみでは不十分である。さらに、メッシュベース移流法では三角形ポリゴンに速度  $\dot{\mathbf{x}}$  に従って移流させるが、法線方向成分のみではポリゴンの大きな変形を引き起こす。図 1 に 2 次元で円を平行移動させたときの表面速

度とその法線方向速度を示す。移動方向と法線方向が同一である面では両者の速度は一致するが、間の角度が垂直に近づくほどその速度は大きく異なってしまう。よって、接線方向も考慮した速度場の算出が必要となる。

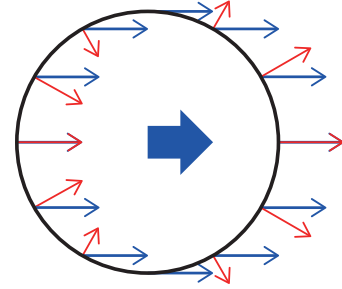


図 1: 表面移流速度 (青矢印) とその法線方向成分 (赤矢印)

この章の以下の節では、我々の提案する曲率不変条件 (3.2 節)、最小自乗法を用いた移流速度算出手法 (3.3 節)、そして、移流後のメッシュ修正方法 (3.4 節) について述べる。

#### 3.2 曲率不変条件

Stam と Schmidt[12] は陰関数  $f$  の勾配方向  $\nabla f / |\nabla f|$  が時間変化しないと仮定し、接線方向成分を含む速度場を算出した。

$$\frac{d}{dt} \left( \frac{\nabla f}{|\nabla f|} \right) = 0 \quad (5)$$

剛体の平行移動を考えたとき、法線方向が  $\Delta t$  後も保たれるというこの仮定はあっている。しかし、回転運動では法線方向が変化するため対応できない。図 2 は円形状を平行移動した場合と、回転移動した場合の法線方向の変化を示している。平行移動時では法線方向は全く変化しない (図 2 左) ため、式 (5) の条件は成り立つ。しかし、回転運動では法線方向が大きく変化する (図 2 右)。そのため、式 (5) の条件だけではこのような運動に対応できない。そこで、平行移動とともに回転を局所的に把握する方法として曲率を利用する。

陰関数曲面の法曲率は 2 階微分を用いて次式で計算できる。

$$\kappa(\mathbf{u}) = -\frac{(\mathbf{u}, \nabla^2 f \mathbf{u})}{|\nabla f|} \quad (6)$$

ここで、 $\mathbf{u}$  は接平面内の単位ベクトルを表す。曲率は曲率半径の逆数としても表され、ある点において表面がどのくらい曲がっているかということを定量的に評価できる。この曲率が時間変化に依存しないと仮定する。すなわち、

$$\frac{d\kappa}{dt} = 0 \quad (7)$$

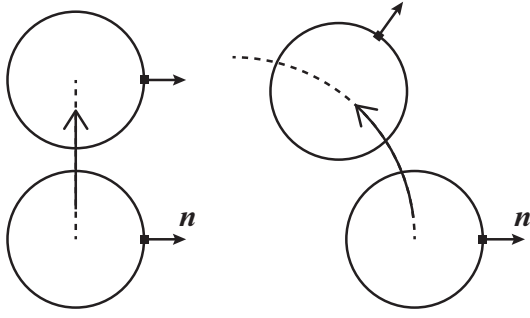


図 2: 平行移動 (左), 回転 (右) したときの法線方向の変化

式 (7) のことを曲率不変条件と呼ぶこととする. 3次元では接平面内の速度ベクトルを決定するために条件が2つ必要であるので, 以下の2条件を用いる.

$$\begin{aligned} \frac{dK}{dt} &= 0, \\ \frac{dH}{dt} &= 0 \end{aligned} \quad (8)$$

ここで,  $K, H$  はそれぞれガウス曲率, 平均曲率であり, 以下で与えられる [16].

$$\begin{aligned} K &= \frac{(\Delta f)^2 - \text{tr}(\nabla^2 f)^2}{2|\nabla f|^2} \\ &\quad + \frac{(\nabla f, (\nabla^2 f)^2 \nabla f) - \Delta f (\nabla f, \nabla^2 f \nabla f)}{|\nabla f|^4}, \quad (9) \\ H &= \frac{1}{2} \left( \frac{\Delta f}{|\nabla f|} - \frac{(\nabla f, \nabla^2 f \nabla f)}{|\nabla f|^3} \right) \end{aligned}$$

ここで  $\nabla^2 f$  はヘッセ行列,  $\Delta f$  はラプラシアンである.  $K, H$  を時間により全微分し, 展開することで表面速度  $\dot{\mathbf{x}} = (x_t, y_t, z_t)$  に関する2つの式を得る. 付録 A に  $K, H$  の全微分の展開式を示す. 式 (2) と式 (7) もしくは式 (8) (3次元の場合) を連立させ, 線形システムを解くことで接線方向も含まれる速度場が得られる. 剛体運動のみを行い, 形状が変化しないと仮定すると, これらの式から平行移動とともに回転運動を局所的に把握できる.

曲率不変条件は流体のような形状を変形する場合にも, 局所的に曲率が変化しないと仮定し適用することが可能である. ただし, 球面がその形状に沿って回転している場合や平面が法線方向に垂直に平行移動している場合など, 曲率が接平面上で変化しない場合には運動を把握することはできない. これらの形状が含まれていると線形システムを必ずしも解くことができない. このような状況に対応するために, 次節で述べるように最小自乗法により移流速度を算出する. さらに, 3.4節で述べるように, 周囲にある曲率が一定でない連続した曲面, つまり曲率不変条件が適用できる形状に追従させることでメッシュを移流させる.

### 3.3 移流速度の算出

曲率不変条件ではガウス曲率, 平均曲率の値のうちどちらからでも接平面上で大きさが0になると解が不定となる. また, 曲率の接平面上での変化がない場合は接線方向速度は算出できない. これらの算出可能条件を明確に定義することは難しく, 結果として不適当な解が移流速度として適用されてしまう. これを解決するために, 曲率不変条件の式と Total Velocity の式をあわせた6式の値の和をエラー値  $e$  として定義し, 未知数を接線方向速度として, 最小自乗法により解を近似する.

$$e = \sum_{i=1}^6 F_i \quad (10)$$

ここで,  $F_i$  は式 (2), 式 (8), および, Total Velocity の式の左辺を計算した値となる. ただし, 式 (2) は結局, 曲率不変条件, Total Velocity で共通となるので実質的には5式の値の和をエラー値として用いる. Total Velocity の式も含めることで解が算出できる点を増やすことができ, より正確な移流速度を求めることができる. 本研究では計算の簡素化のために最小値探索には関数値のみで実装できる Powell 法 [13] を用いたが, Total Velocity, 曲率不変条件式の勾配を算出することで BGFS 法などの準ニュートン法でより効率的に計算できる.

### 3.4 メッシュフェアリング

球体の回転や平面の接線方向への平行移動など陰関数値が変化せず, 解が正確に算出できない領域が存在する. そのため, 剛体においても移流後に三角形メッシュが大きくゆがむという現象が発生する. これを解決するために Stam ら [12] と同様にバネ-質点系を用いてメッシュフェアリングを行う. メッシュフェアリングの手順を以下に示す.

1. 各頂点において隣接頂点 (one-ring neighborhood) を接平面に投影する.
2. 隣接頂点を固定し, 初期エッジ長さをバネの初期長さとしてバネ-質点系により頂点位置を移動させる.
3. すべての頂点で 1, 2 を行ったのち, 以下の式により頂点を表面上にフィッティングする.

$$\mathbf{x}^{n+1} = \mathbf{x}^n - \frac{f(\mathbf{x}) \nabla f(\mathbf{x})}{\|\nabla f(\mathbf{x})\|^2} \quad (11)$$

これら一連の処理を反復することでメッシュをフェアリングする。このとき、バネ-質点系を使うことで解が不定となり、接線方向速度がほとんどでなかった点に引っ張られて、正しい接線方向速度で移流した点も修正されてしまうが、曲率不変条件を使うことでより多くの点を拘束点として使えるため、その影響を小さくすることができる。

## 4 結果

提案手法をパーティクルに対して実装した結果を示す。各パーティクルは中心座標と半径を持ち、以下の式により陰関数場を構成する [12]。

$$f(\mathbf{x}, t) = \sum_{i=1}^N w_i (1 - r^2)^3 - T \quad (12)$$

ここで、 $N$  は近傍パーティクル数、 $w_i$  は重み係数、 $T$  は閾値、そして、 $r = |\mathbf{x} - \mathbf{x}_i|/\sigma_i$  は近傍パーティクル中心までの距離を半径  $\sigma$  で正規化した値である。

まず、1つのパーティクルを左から右へ平行移動させた結果を図3に示す。図3(a)の法線方向速度では図1で示したように、移動方向に対して垂直に近い法線を持つ頂点が移動せずに取り残され、結果として形状が変化してしまっている。注意として、形状が大きく変形しても法線方向に移動しているため、頂点は球面上に位置している。一方、Total Velocity(図3(b))、曲率不変条件(図3(c))を用いた場合、メッシュの変形もほとんどみられない。この例では平行移動を行ったので、Total Velocityでも問題なく移流速度を算出できている。

次に、剛体回転として3つのパーティクルを原点を中心とした円上に配置し(図4(a))、図正面から見て原点中心で反時計回りに3回転させた結果を図4(b)-(d)に示す。フェアリング反復数は10とした。回転後すべての頂点が初期位置に戻るのが理想である。法線速度場を用いた図4(b)ではメッシュ移流が回転に全く追従しない。Total Velocityを用いた図4(c)では回転に対して追従するものの、かなりのずれが生じている。本手法による結果(図4(d))をみると、Total Velocityに比べて移流のずれが改善できていることがわかる。移流に対する誤差を定量的に評価するために、回転後の座標と元の座標の間の距離を誤差としてプロットしたものを図5に示す。横軸はフェアリングにおける反復数である。反復数を増やすと計算速度が大きく増大するが精度を向上させることができる。曲率不変条件ではTotal Velocityに比べて全体的な精度向上を実現しており、さら

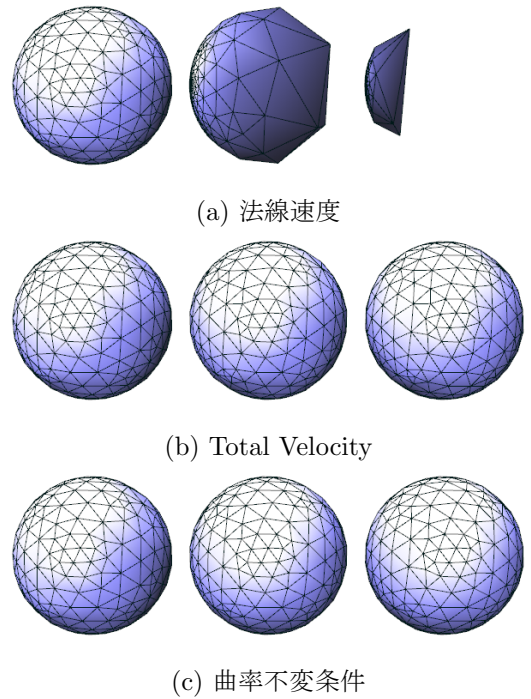


図3: パーティクルの平行移動

にフェアリング反復数が少ない場合でも精度良く頂点を移流できている。

提案手法が回転において有効に働くことを確かめるために、図6に示すように1つのパーティクルを30回転させた。結果を図7に示す。ただし、この例ではフェアリングを行っていない。Total Velocityでは球が変形してしまっているが、提案手法ではほとんど変化していない。1回転ごとに頂点誤差をプロットしたものを図8に示す。Total Velocityでは回転数に比例して誤差が蓄積されているのに対して、曲率不変条件では誤差の増加がほぼみられない。

最後に変形する場合の例として、SPH法による粘性流体シミュレーションに提案手法を適用した結果を図9に示す。階段状の固体境界上を表面が回転しながら滑り落ちるシーンを設定した。提案手法の結果である図9(b)では、図9(a)に比べて表面の回転をより強くとらえていることがわかる。ただし、最小自乗法において間違った局所解に収束し、不適当な位置に頂点が移流することがあり、それにより一部メッシュが乱れている。

## 5 まとめと今後の課題

本論文ではメッシュベース移流法において曲率不変条件を用いることで、従来手法の問題点であった回転に対応した移流速度の算出方法を提案し、実験結果により剛体回転



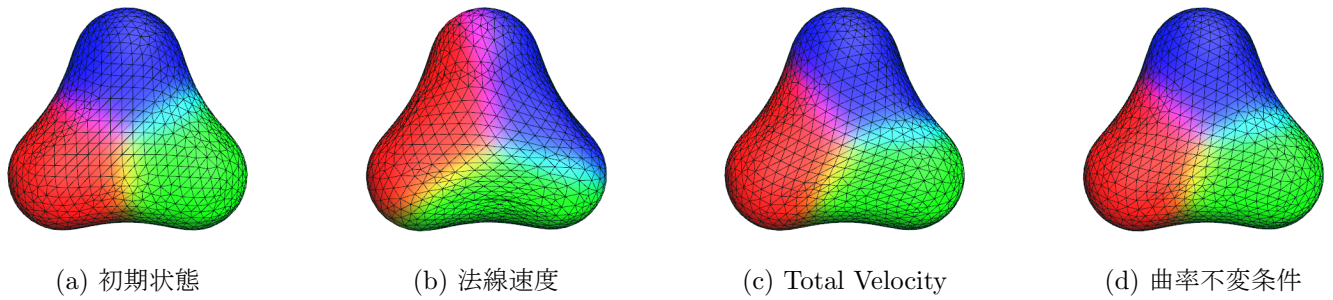


図 4: 剛体回転

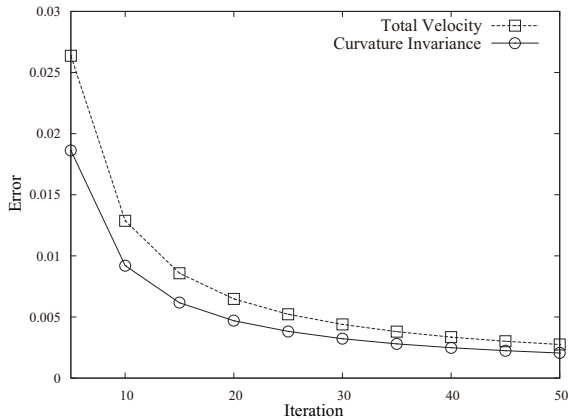


図 5: フェアリング反復数と頂点誤差

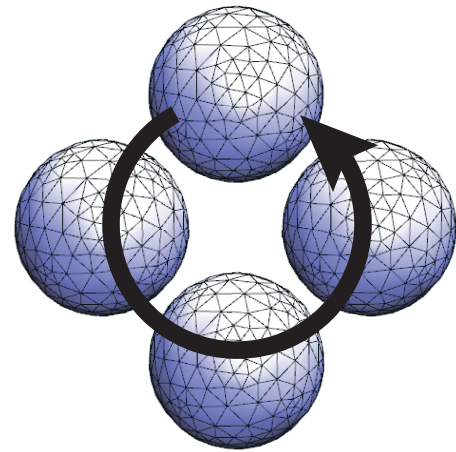


図 6: パーティクル回転

について従来手法と比べて高精度に移流できることを示した。また、精度をより高めるために最小自乗法を導入し、メッシュフェアリングの反復数が少ない場合でも、陰関数の移流速度により追従することが確認できた。さらに、パーティクル法を用いた粘性流体シミュレーション結果に適用し、変形が発生するシーンにおいても手法がある程度有効に働くことを示した。

今後の課題としては、図 9 にみられたようなフェアリングにおけるメッシュの乱れの改善があげられる。最小自乗法において間違った局所解に収束したことを検出し、それらの点をメッシュフェアリングにおける拘束点として用いないことでこの問題は解決できると考えている。さらに、Enright テスト ([2] など) に本手法を適用することで変形に対するロバスト性を確かめることができる。また、より複雑なシーンに対応するために Müller[10] や Wojtan ら [14] の方法を導入し、メッシュのトポロジ変化に対応させることも今後の課題である。

## 参考文献

- [1] Adam W. Bargteil, Tolga G. Goktekin, James F. O'Brien, and John A. Strain. A semi-lagrangian contouring method for fluid simulation. *ACM Transactions on Graphics*, Vol. 25, No. 1, pp. 19–38, 2006.
- [2] Haimasree Bhattacharya, Yue Gao, and Adam Bargteil. A level-set method for skinning animated particle data. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '11, pp. 17–24. ACM, 2011.
- [3] Tyson Brochu and Robert Bridson. Robust topological operations for dynamic explicit surfaces. *SIAM Journal on Scientific Computing*, Vol. 31, No. 4, pp. 2472–2493, 2009.
- [4] Douglas Enright, Stephen Marschner, and Ronald Fedkiw. Animation and rendering of complex water

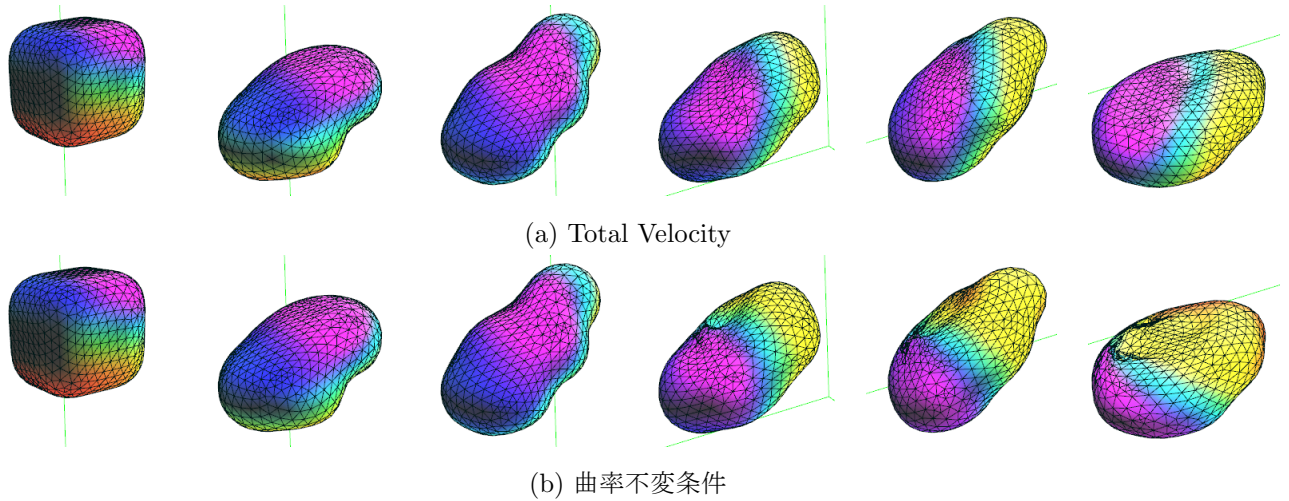


図 9: 階段上を流れる粘性流体

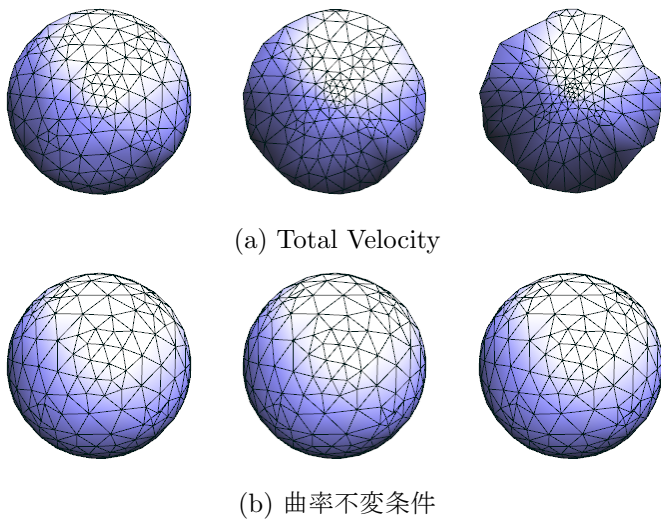


図 7: パーティクルを回転させたときのメッシュ変化. 左から 10 回転, 20 回転, 30 回転時の結果

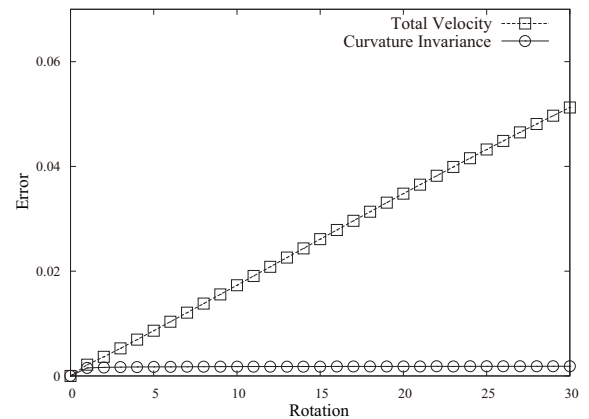


図 8: 回転数と頂点誤差

surfaces. In *Proceedings of SIGGRAPH 2002*, pp. 736–744, New York, NY, USA, 2002. ACM Press.

- [5] Nick Foster and Ronald Fedkiw. Practical animation of liquids. In *Proceedings of SIGGRAPH 2001*, pp. 23–30, New York, NY, USA, 2001. ACM Press.
- [6] ByungMoon Kim, Yingjie Liu, Ignacio Llamas, and Jarek Rossignac. Advections with significantly reduced dissipation and diffusion. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 13, No. 1, pp. 135–144, 2007.
- [7] Theodore Kim, Nils Thürey, Doug James, and Markus Gross. Wavelet turbulence for fluid simu-

lation. In *SIGGRAPH '08: ACM SIGGRAPH 2008 papers*, pp. 1–6, New York, NY, USA, 2008. ACM.

- [8] W. E. Lorensen and H. E. Cline. Marching cubes: a high resolution 3d surface construction algorithm. *Computer Graphics (SIGGRAPH '87 Proceedings)*, Vol. 21, No. 4, pp. 163–169, 1987.
- [9] V. Mihalef, B. Unlusu, D. Metaxas, M. Sussman, and M. Y. Hussaini. Physics based boiling simulation. In *Proc. SCA2006*, pp. 317–324, 2006.
- [10] Matthias Müller. Fast and robust tracking of fluid surfaces. In *SCA '09: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 237–245, New York, NY, USA, 2009. ACM.

- [11] S. Osher and J. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, Vol. 79, pp. 12–49, 1988.
- [12] J. Stam and R. Schmidt. On the velocity of an implicit surface. *ACM Trans. Graph.*, Vol. 30, pp. 21:1–21:7, 2011.
- [13] S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C++: The Art of Scientific Computing*. Cambridge University Press, 2002.
- [14] C. Wojtan, N. Thürey, M. Gross, and G. Turk. Deforming meshes that split and merge. In *SIGGRAPH 2009*, pp. 1–10, 2009.
- [15] C. Wojtan, N. Thürey, M. Gross, and G. Turk. Physics-inspired topology changes for thin fluid features. In *SIGGRAPH 2010*, pp. 1–8, 2010.
- [16] 金谷健一. 形状 CAD と図形の数学. 共立出版, 1998.

次に  $H$  に関する曲率不変条件式は以下となる.

$$\begin{aligned} \frac{dH}{dt} &= \{|\nabla f|^4(\mathbf{d}f_{xx} + \mathbf{d}f_{yy} + \mathbf{d}f_{zz}) \\ &\quad - |\nabla f|^2 \Delta f (f_x \mathbf{d}f_x + f_y \mathbf{d}f_y + f_z \mathbf{d}f_z) - \\ &\quad \{2(\mathbf{d}f_x, \mathbf{d}f_y, \mathbf{d}f_z)^T \nabla^2 f \nabla f \\ &\quad + (\nabla f)^T \begin{bmatrix} \mathbf{d}f_{xx} & \mathbf{d}f_{xy} & \mathbf{d}f_{xz} \\ \mathbf{d}f_{xy} & \mathbf{d}f_{yy} & \mathbf{d}f_{yz} \\ \mathbf{d}f_{xz} & \mathbf{d}f_{yz} & \mathbf{d}f_{zz} \end{bmatrix} \nabla f\} |\nabla f|^2 + \\ &\quad 3(\nabla f)^T \nabla^2 f \nabla f (f_x \mathbf{d}f_x + f_y \mathbf{d}f_y + f_z \mathbf{d}f_z)\} \cdot \mathbf{V} = 0 \end{aligned}$$

## A 曲率式の時間による全微分

ガウス曲率と平均曲率の時間による全微分を速度  $\mathbf{v} = (x_t, y_t, z_t)$  について展開した式を示す. この付録では式を簡潔に表現するために以下の表記法を用いる.

$$\begin{aligned} \frac{df}{dt} &= \frac{\partial f}{\partial x} \frac{dx}{dt} + \frac{\partial f}{\partial y} \frac{dy}{dt} + \frac{\partial f}{\partial z} \frac{dz}{dt} + \frac{\partial f}{\partial t} \\ &= (f_x, f_y, f_z, f_t) \cdot (x_t, y_t, z_t, 1) \\ &= \mathbf{d}f \cdot \mathbf{V} \end{aligned}$$

式(9)のガウス曲率, 平均曲率式[16]を時間により全微分し, 上記表記法を用いて整理した結果を以下に示す. まず,  $K$  に関して,

$$\begin{aligned} \frac{dK}{dt} &= \{\Delta f |\nabla f|^4 (\mathbf{d}f_{xx} + \mathbf{d}f_{yy} + \mathbf{d}f_{zz}) \\ &\quad - (\Delta f)^2 |\nabla f|^3 (f_x \mathbf{d}f_x + f_y \mathbf{d}f_y + f_z \mathbf{d}f_z)\} \cdot \mathbf{V} - \\ &\quad \frac{1}{2} |\nabla f| \left\{ \frac{d}{dt} \{tr(\nabla^2 f)^2\} |\nabla f|^3 \right. \\ &\quad \left. - 2tr(\nabla^2 f)^2 (f_x \mathbf{d}f_x + f_y \mathbf{d}f_y + f_z \mathbf{d}f_z) \cdot \mathbf{V} \right\} + \\ &\quad \frac{d}{dt} \{(\nabla f)^T (\nabla^2 f)^2 \nabla f\} |\nabla f|^2 \\ &\quad - 4(\nabla f)^T (\nabla^2 f)^2 \nabla f (f_x \mathbf{d}f_x + f_y \mathbf{d}f_y + f_z \mathbf{d}f_z) \cdot \mathbf{V} - \\ &\quad \frac{d}{dt} \{\Delta f (\nabla f)^T \nabla^2 f \nabla f\} |\nabla f|^2 \\ &\quad + 4(\nabla f)^T \nabla^2 f \nabla f (f_x \mathbf{d}f_x + f_y \mathbf{d}f_y + f_z \mathbf{d}f_z) \cdot \mathbf{V} = 0 \end{aligned}$$