

機関番号：12102

研究種目：研究活動スタート支援

研究期間：2009～2010

課題番号：2180007

研究課題名（和文） 粒子ベースシミュレーション結果の高速かつ高品質可視化手法の開発

研究課題名（英文） Development of Fast and High-Quality Visualization Methods for Particle-based Simulations

研究代表者

金森 由博 (YOSHIHIRO KANAMORI)

筑波大学・大学院システム情報工学研究科・助教

研究者番号：10551418

研究成果の概要（和文）：本研究では、粒子ベースの流体シミュレーション結果を高品質かつ高速に可視化する手法の開発を行った。具体的には、メタボールと呼ばれる陰関数曲面による液面の抽出手法を2つ提案した。2009年には、グラフィクスハードウェア（GPU）を効率的に用いて、申請者が2008年に提案した手法よりも高速な手法を開発し学会発表した。2010年には、従来の抽出手法では液面に粒子に由来する凹凸が生じるため、それを解消する手法を開発し学会発表した。

研究成果の概要（英文）：This study aimed at development of fast and high-quality methods for visualizing results of particle-based liquid simulations. Specifically, the applicant developed two methods for extracting liquid surfaces based on an implicit-surface technique called metaballs. In 2009, the applicant developed and published a faster method that utilizes graphics hardware (GPU) than the method developed by the applicant in 2008. In 2010, the applicant also developed and published a method that reduces particle-derived small bumps on the surface, which are inevitable when using traditional techniques.

交付決定額

（金額単位：円）

	直接経費	間接経費	合計
2009年度	1,080,000	324,000	1,404,000
2010年度	930,000	279,000	1,209,000
総計	2,010,000	603,000	2,613,000

研究分野：コンピュータグラフィクス

科研費の分科・細目：メディア情報学・データベース

キーワード：可視化、コンピュータグラフィクス

1. 研究開始当初の背景

粒子ベースシミュレーションは、科学技術分野からエンターテインメント分野まで広く普及している。粒子ベースの物理シミュレーションを行う際、流体の液面などを可視化するために、メタボールと呼ばれる陰関数曲面がよく利用される。各メタボールは密度分布を持ち、多数のメタボールからなる密度場の等値面が滑らかな曲面を表現する。こういった可視化では、メタボールが多数あり、径が小さく、また時間とともに移動する、とい

う特徴から、計算コストおよびメモリコストが膨大となる。

申請者が本研究を開始した当初、世界最速の可視化手法は、申請者らが2008年に開発した手法であった。NVIDIA社のグラフィクスハードウェア（GPU）である GeForce 8800 Ultra 搭載のPCで実験した場合、約10万粒子からなる結果を可視化する速度は2.3フレーム毎秒であった（図1）。粒子ベースシミュレーションでは数百万、数千万オーダーの粒子を用いてシミュレーションが行われるこ



図1. 申請者らが2008年に開発した手法による、約10万粒子からなる噴水シミュレーションの可視化結果。NVIDIA GeForce 8800 Ultra 搭載のPCでの描画速度は2.3フレーム毎秒。

ともしばしばあるため、GPUが約一年半で倍の性能になっている現状を考慮しても、残念ながらこの速度は不十分と言わざるを得ない。そこで、より多数の粒子に対して、より高速な可視化を可能にする手法を開発する必要があった。

一方、メタボールを用いた液面の可視化において、従来の手法では、速度面だけでなく品質面でも問題があった。メタボールを用いると、メタボールの径が大きい場合ははぶきや液面が不自然に膨らむ、逆にメタボールの径が小さい場合は液面が凸凹になる。そのため従来は、メタボールで表現された液面をまずポリゴンとして抽出したのち、平滑化を施す、ということが行われてきた。これによって品質面の問題が回避されていたが、アニメーションを作成する場合は、これをアニメーションの毎フレームで行う必要があるため、計算コストおよびメモリコストが問題となった。これは、速度が重視されるエンターテインメント分野、特にゲームなどの対話的アプリケーションにおいては致命的である。そこで、ポリゴン化を必要としない、高品質かつ高速な可視化手法を開発する必要があった。

2. 研究の目的

上記の背景を踏まえ、本研究では、比較的安価に入手できるGPUを用いて、メタボールからなる曲面を高品質かつ高速に描画する手法の開発を目指した。速度については、その時点で利用可能なGPUを用いて、100万粒子を対話的な速度で描画できる手法を目指した。品質については、従来行われてきたポリゴンの平滑化と同様な処理を、メタボールの描画処理の流れを大きく変更することなく実現する手法を目指した。

3. 研究の方法

GPUを活用でき、高速に動作するアルゴリズムを考案し、プログラムとして実装、性能を評価した。以下、(1) 高速な描画を目指した手法、(2) 高品質な結果を目指した手法とに分けて、具体的な内容を述べる。

(1) 動的スライス分割によるメタボールのレンダリング

高速な描画を実現する方法として、多数の動くメタボールからなる等値面を、スライスを用いたボリュームレンダリングによって高速に描画する手法を提案した。まず、描画範囲を定義する視野を表す錐台（ビューボリューム）を、視線に垂直な平面（スライス）に分割し、そのメモリを確保したのち、各メタボールの密度分布をスライス上に累積していく。径の小さなメタボールを捕捉するにはスライス間隔を細かくとる必要があり、メモリコストが問題となる。そこで提案法では、密度値の累積に先立ち、シーン中のメタボールの配置を検査し、メタボールの存在する箇所のみスライスを確保する。提案法は従来法に比べ、径の小さい多数の動くメタボールを、より高速に描画することが可能である。提案法は以下の4つの処理からなる。

- ① シーンを粗くボクセル分割し、メタボールと交差またはメタボールを内包するボクセルを特定
 - ② メタボールを含むか交差するボクセルをスライスに分割し、メモリを確保
 - ③ スライスに密度場を書き込む
 - ④ 密度場を元に直接法等値面を描画する
- 以下、各処理について説明する。

① シーンの粗いボクセル分割

シーン中のメタボールの配置を調べるため、シーンを粗くボクセル分割し、メタボールを含むか交差するボクセルを特定する。この処理を効率的に行うために、1つのボクセルを1ビットで表現し、GPUのテクスチャ処理で高速に処理するEisemannらのボクセル化手法を応用する。彼らの手法はポリゴンメッシュからなる物体を対象とし、シーン中の各物体を1回ずつ描画するだけでボクセル化が完了する。さらに、ボクセルを1ビットで表現するので、メモリ消費が少ない。ただし、複数の物体が互いに交差する場合、ボクセル化に失敗する。

提案法では、Siggらの手法を用い、各メタボールを包含する楕円体を1回ずつ描画する。その際に楕円体を含むか交差するボクセルを特定し、そのボクセルに対応するビットを1にする。この結果を、OR演算を用いてテクスチャに書き込むことで、楕円体が互いに交差する場合も正しく計算できる。実験では、ボクセル用テクスチャのx、y方向の解像度

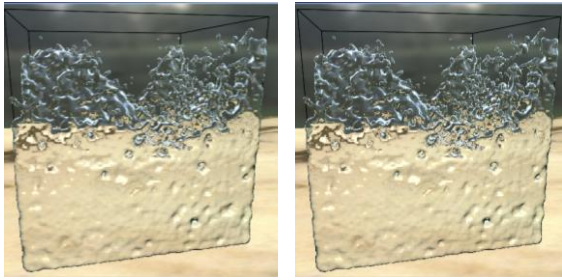


図2. 申請者らが2008年に開発した手法(左)と提案法(右)の比較。描画結果はほぼ同等であるが、1フレームあたりの描画時間はそれぞれ1063ミリ秒と92.2ミリ秒。

は8x6、32ビット整数フォーマットのテクスチャ1枚のRGBAの各チャンネルを用いて、z方向に32x1x4=128ビットを確保した。

② スライスメモリ確保

ボクセル用テクスチャをCPUに読み戻し、ビットが1のボクセルに対して、複数枚のスライスを割り当てる。ボクセル1つにつき複数枚のテクスチャを割り当て、テクスチャのRGBA各チャンネルをスライス1枚とする。実験では、1ボクセルあたり(テクスチャ2枚)x(4チャンネル)=8枚のスライスを割り当てた。z方向のボクセル数を128としたので、z方向の実質スライス数は128x8=1024枚となる。

スクリーンの解像度が640x480のとき、計算コストを下げるため、密度値を書き込むテクスチャの解像度は160x120とした。このテクスチャを、ボクセル化のときと同様に、x、y方向に8x6のタイルに分割する。各タイルが各ボクセルに対応し、必要なタイルを2Dテクスチャに敷き詰める。

③ スライスへの密度場の書き込み

Siggらの手法を用いて各メタボールを包含する楕円体を1回ずつ描画し、上記②で確保したスライスに密度値を累積する。楕円体が複数のタイルと交差する場合、1回描画するだけでは書き込みが行えないため、ジオメトリシェーダを用いて必要回数だけ描画する。

④ 等値面の描画

上記③で書き込まれた密度値を線形補間し、等値面を計算する。x、y方向についてはスライスのx、y方向について線形補間し、z方向についてはスライス間で線形補間を行う。

ここまで述べた手法について、C++、OpenGL、Cgを用いて実装を行い、Intel Core 2 Duo 2.66GHz、メモリ3GB、NVIDIA GeForce 9600MS搭載のノートPCで実験を行った。

2万粒子からなる流体シミュレーションを

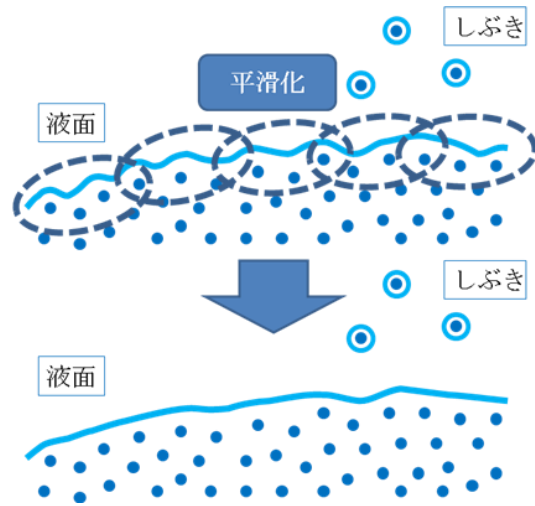


図3. 平滑化を施したメタボールのレンダリング手法の概要。液面に異方性フィルタリングと同等な処理を施し、凹凸を平滑化する。

可視化した例を図2に示す。申請者らが2008年に開発した手法を用いた場合(図2左)に比べ、提案法ではほぼ同等の品質でありながら、約11倍高速に描画が可能である。また、当初の目標として掲げた、100万粒子の対話的に描画についても、NVIDIA GeForce GTX 295で数フレーム毎秒の速度が出ることを確認しており、目標を達成することができた。

(2) 平滑化を施したメタボールのレンダリング

メタボールを用いた場合に生じる問題、すなわち、メタボールの径が大きい場合はしぶきや液面が不自然に膨らむ、逆にメタボールの径が小さい場合は液面が凸凹になる、といった問題を解決する手法を提案した。提案手法ではしぶきと液面の粒子を判別し、液面のみを異方性フィルタリングと同等な処理で平滑化する。すなわち、液面のメタボールの径を大きくし、近傍粒子の分布に応じて形状を変形したのち、変形に応じてメタボールの濃度値を正規化する。提案手法は、既存のリアルタイムレイトレーサに大幅な変更を加えることなく組み込むことが可能であり、対話的アプリケーションにおける液面表現を改善できる。

提案手法は、液面に沿った異方性フィルタカーネルを用いて平滑化すると同等な処理により、液面の凹凸を低減する(図1)。すなわち、液面の粒子に対して、メタボールの有効半径を大きくし、メタボールの形状を近傍の粒子分布に応じて変形させ(楕円体状にする)、そしてメタボールの有効半径内の濃度値の総和(積分値)が一定になるよう濃度値を正規化する。この拡張は、(1)式に当てはめると、重みの修正(正規化)と、濃度関

数の引数の修正（形状の変更）に相当する。これらの修正は既存のメタボールのレンダリング手法の枠組みに大幅な変更なく組み込むことができる。

提案手法での、各粒子に対する具体的な計算手順は次のようになる：

- ① しぶきか液面（液体内）かの判定
- ② メタボールの有効半径の決定
- ③ メタボールの形状の決定
- ④ メタボールの濃度値の正規化

以下、これらの各処理について説明する。

- ① しぶきか液面（液体内）かの判定

粒子 i がしぶきか液面（液体内）かを、粒子密度 ρ_i に基づき、ユーザ指定の閾値 ρ_{splash} および $\rho_{surface}$ を用いて次のように判定する。

- (i) $\rho \leq \rho_{splash}$: しぶき
- (ii) $\rho_{splash} < \rho < \rho_{surface}$: しぶきと液面の中間状態
- (iii) $\rho_{surface} \leq \rho$: 液面（液体内）

以下、条件 (i) を満たす粒子を「しぶき粒子」、条件 (iii) を満たす粒子を「液面粒子」と呼ぶことにする。

- ② メタボールの有効半径の決定

粒子 i に割り当てるメタボールの有効半径 R_i を、上記①で計算した粒子密度 ρ_i に基づき線形補間によって決定する：

$$R_i = (1-s)R_{splash} + sR_{surface},$$

$$s = \text{smoothstep}(\rho, \rho_{splash}, \rho_{surface})$$

ここで R_{splash} および $R_{surface}$ はユーザ指定の定数である。smoothstep 関数は $[0, 1]$ の範囲の値を返す関数であり、次式で定義される：

$$\text{smoothstep}(t, t_{\min}, t_{\max}) = \begin{cases} 0 & (t < t_{\min}) \\ \left(\frac{t-t_{\min}}{t_{\max}-t_{\min}} \right)^2 \left\{ 3 - 2 \left(\frac{t-t_{\min}}{t_{\max}-t_{\min}} \right) \right\} & (t_{\min} \leq t \leq t_{\max}) \\ 1 & (t_{\max} < t) \end{cases}$$

- ③ メタボールの形状の決定

液面を平滑化するため、液面のメタボールの形状を、液面に沿うよう扁平な楕円体に変形する（図 2）。これを実現するため、粒子 i の近傍の粒子分布を表す、重み付き共分散行列 $\Sigma_i \in \mathbf{R}^{3 \times 3}$ を計算する：

$$\Sigma_i = \frac{1}{\sum_j g_j} \left(\sum_j g_j \frac{(\mathbf{c}_j - \boldsymbol{\mu}_i)^T (\mathbf{c}_j - \boldsymbol{\mu}_i)}{\|\mathbf{c}_j - \boldsymbol{\mu}_i\|^2} \right),$$

$$\boldsymbol{\mu}_i = \frac{\sum_j g_j \mathbf{c}_j}{\sum_j g_j}, \quad g_j = g(\mathbf{c}_j - \mathbf{c}_i, h)$$

ここで、 g は重み関数である。本研究ではメタボールの濃度関数と同じ 6 次多項式を用いた。 h はサポート半径である。

- ④ メタボールの濃度値の正規化

液面のメタボールに異方性フィルタカーネルを畳み込むと、メタボールの濃度は薄くなる。すなわち、有効半径を大きくし、楕円体に変形したメタボールの濃度値は、小さくしなければならない。これを実現するために、各メタボール内の濃度値の総和（積分値）が、基準となる大きさのメタボールの値と等しくなるよう、濃度値を正規化する。

正規化のための係数の導出を簡単にするために、メタボールの濃度関数としてガウス関数を考える。基準となるメタボールは有効半径が R_0 の球とする。すると、有効半径 R_i 、共分散行列 Σ_i のメタボール i のための正規化係数 C_i は、次式で与えられる：

$$C_i = \frac{\int \exp\left(-\frac{(\mathbf{x} - \boldsymbol{\mu}_i)^T (\mathbf{x} - \boldsymbol{\mu}_i)}{R_0^2}\right) dV}{\int \exp\left(-\frac{(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)}{R_i^2}\right) dV}$$

$$= \left(\frac{R_0}{R_i}\right)^3 \frac{1}{\sqrt{\det \Sigma_i}}$$

なお、有効半径の比 (R_0 / R_i) は 3 次元の場合は 3 乗されるが、2 次元の場合は 2 乗となる。また、本研究で用いる濃度関数は、ガウス関数ではなく 6 次多項式であるが、この多項式がガウス関数をよく近似しているものとして、上式を正規化係数として用いる。

ここまで述べた手法について、2 次元の場合についてプログラムを作成し、実験した。図 4 に結果を示す。これは水柱崩壊シミュレーション（粒子数 2720）のレンダリング結果である。図下段は、図上段の赤枠部分の拡大図を示しており、従来の手法で問題となっていた、しぶきや液面が不自然に膨らむ、という問題が見られないことがわかる。なお、研究期間内では 2 次元の結果までしか実験できなかったため、3 次元の結果および GPU を用いた実装については、今後の課題である。

4. 研究成果

メタボールと呼ばれる陰関数曲面による液面の抽出手法を 2 つ提案した。2009 年には、グラフィクスハードウェア (GPU) を効率的に用いて、申請者が 2008 年に提案した手法よりも高速な手法を開発し学会発表した。2010 年には、従来の抽出手法では液面に粒子に由来する凹凸が生じるため、それを

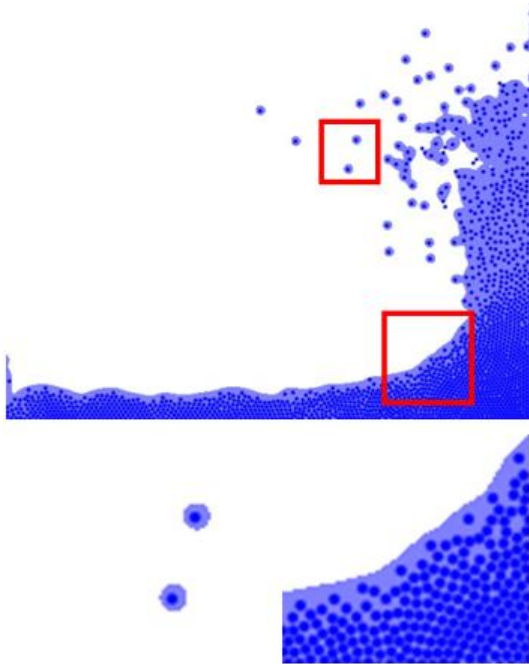


図4. 水柱崩壊シミュレーションのレンダリング結果。図下段は図上段の赤枠部分の拡大図を示しており、従来の手法で問題となっていた、しぶきや液面が不自然に膨らむ、という問題が見られない。

解消する手法を開発し学会発表した。

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[学会発表] (計 3 件)

- ① 金森由博, 西田友是, “平滑化を施したメタボールによる粒子ベース流体シミュレーションの可視化”, Visual Computing Workshop 2010, 静岡, 2010/11/26-27.
- ② 金森由博, “GPU-based Fast Ray Casting for a Large Number of Metaballs” (CG 国際大賞・優秀賞受賞講演), NICOGRAPH 2010 春季大会, 2010/3/27.
- ③ 金森由博, 西田友是, “動的スライス分割によるメタボールのレンダリング”, Visual Computing/グラフィクスと CAD 合同シンポジウム 2009 (GCAD 賞受賞), 2009/6/26.

[その他]

ホームページ等

<http://kanamori.cs.tsukuba.ac.jp/>

6. 研究組織

(1) 研究代表者

金森 由博 (YOSHIHIRO KANAMORI)

筑波大学・大学院システム情報工学研究科・

研究者番号 : 10551418

(2) 研究分担者
なし ()

研究者番号 :

(3) 連携研究者
なし ()

研究者番号 :