I-007

GPUを用いた相互作用のある粒子の実時間描画

Real-time rendering of particles with the interaction by using GPU

三嶋 仁† 田中 敏光† 佐川 雄二†
Hitoshi Mishima Toshimitsu Tanaka Yuji Sagawa

1．はじめに

グラフィックハードウェア(GPU)の進歩により，パーティクル・システム[1]でも多数の粒子をリアルタイム表示できるようになった．しかし，近傍の粒子を探索するコストが大きいため，粒子間の相互作用は満足に表現できていない．GPU を利用して，空間分割で近傍の粒子を検出する方法[2]や，粒子とポリゴンとの衝突計算手法[3]などが開発されているが，十分とはいえない．

そこで本研究では，GPU で粒子間の相互作用を高速処理する方法を検討する．これを利用して，実在感のある煙，霧，水，炎をリアルタイム表示し，消火シミュレーションやドライブシミュレーションなどに応用する．

2．粒子の相互作用の計算

粒子間の距離が大きくなると，粒子の相互作用は急速に弱くなる．そこで，近傍の粒子だけから粒子の運動を計算する，近傍粒子は次の方法で求める．

2.1 空間分割

粒子が自由に動き回ることのできる状態で近傍粒子を求める必要があるので，構築や参照が簡単な三次元一様グリッドで空間を分割する．そして，粒子が入っているグリッドの番号を，その粒子のデータとして記録する．

2.2 ソート

ある粒子の近傍粒子は，同一のグリッド番号か，隣接するグリッドの番号を持つ．そこで，近傍粒子を効率的に探索するために，粒子をグリッド番号でソートする．グリッド番号は三次元座標(x,y,z)で与えられるので，グリッドの分割数 N を使ってグリッドインデックス $x+yN+zN^2$ を計算し，バイトニックソートで昇順に並べ直す．

2.3 グリッドテーブル

ソートされた粒子リストを先頭からスキャンし，それぞれのグリッドに含まれる粒子の先頭アドレスと最終アドレスを，グリッドインデックスを引数とするテーブルに格納する．図１にその例を示す．左図がソート済みの粒子リストである．同じグリッド番号を同じ色で示しているので，色が連続している．粒子は一次元のリストで記述されるが，

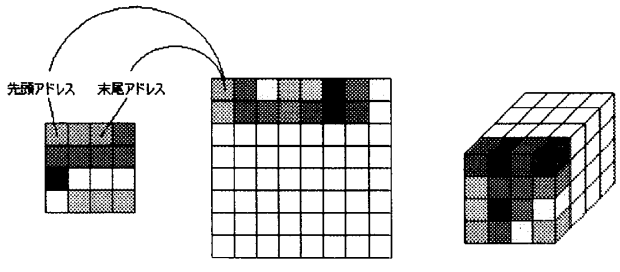

図 1. グリッドテーブル

GPU のテクスチャメモリに保存するために，２次元配列として領域を確保している．グリッドは右図のような三次元構造を持つが，同様の理由で，二次元配列に保存する．

粒子のグリッド番号からグリッドインデックスを計算し，グリッドテーブルを参照することで，近傍粒子をわずかなコストで参照できる．

3 任意形状との交差判定

粒子と一般の物体との衝突を判定するには，粒子の軌跡と任意形状との交差判定が必要となる．リアルタイム CG では物体形状は三角形パッチで記述されるので，三角形の各頂点を内包する外接直方体の木構造（AABB tree）で交差する三角形の絞りこみを行った後に，三角形との境界判定を行う．このアルゴリズムを GPU に実装することで，効率的な衝突判定を実現した．

4．メタボールによる形状表現

粒子の位置にメタボールを置くと，水の様な，表面が滑らかで，かつ，分離・融合する物質を表示できる．メタボールを描画するには，視線に沿ってポテンシャルを計算し，その値が閾値を超えた位置を表面と判定する処理が必要となる．本研究では，GPU を用いて，動的に変化するメタボールを実時間表示する．

4.1 従来研究

レイキャストを使う手法では，画素単位でレイを飛ばし，レイに沿って一定間隔でポテンシャルを計算して，表面を求める．このため，画像の品質は高いが，計算コストも高くなる．マーチンキューブを使う方法では，空間を任意の区画で分割し，各区画でポテンシャルを求める．その後，周囲の区画のポテンシャルを調べ，予め用意されたポリゴンパターンを割り当てて表面を生成する．このため，計算は高速であるが，画面の解像度に対して空間分割が十分でないと，画像品質が低下する．

4.2 提案手法

空間を等間隔にスライスし，始点から近い順に，スライス面と視点の交点でポテンシャルを計算する．正のポテンシャルが算出したら，その位置と１つ前のスライス面との間で表面の位置を算出する．ただし，空間を均一に分割したのでは，無駄な処理が多くなる．このため，次に示す高速化の工夫を行った．また，GPU で処理しやすいように，アルゴリズムを工夫した．

4.3 スライス面の設定

粒子の影響は，粒子からの距離が遠くなるにつれて急速に弱くなる．そこで，ポテンシャル関数にを用いて，粒子の影響範囲を制限する．粒子が影響を与える限界の距離を影響半径と呼ぶ．

粒子が単独で存在する場合には，粒子から一定の距離に表面が発生する．このときの半径を粒子半径と呼ぶ．もし，ある場所にメタボールの表面が存在すれば，それは，ある

†名城大学大学院理工学研究科

粒子の中心以上で，かつ，影響半径以下の距離となる．したがって，この範囲を調べればよい．

メタボールで作られる形状はソリッドとなる．しかし，描画に必要となるのは視点側の面だけである．そこで，先に決めた探索範囲を始点から見える側に限定する．

粒子の位置を画面に投影し，その位置を中心として影響半径の円を描く．この内側で，奥行きが，視点から粒子までの距離から影響半径を引いた値と視点からの距離の間が，探索範囲になる．

以上をまとめると，図 2 の橙色部分がメタボールの表面を探索する範囲となる．これを計算対象領域と呼ぶ．

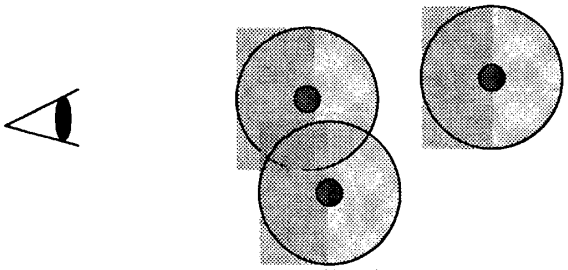

図 2. 粒子の計算対象領域

各画素で，視点に最も近い計算対象領域の前面を最初のスライス面とする． CPU の処理では，条件分岐のコストが高いため，1 つの粒子のスライス回数は常に一定にする．この回数で計算対象領域を等分割して，スライス面の間隔を決める．以後のスライス面は，現在のスライス面にこの間隔を足すことで求める．

4.4 スライス面の移動

一つの粒子を指定した回数だけ探索したら，次の粒子の探索範囲に移動する．このとき，図 3 の左側のように，現在の探索位置（黒丸で表示）が次の粒子の探索範囲と重なっていなければ，途中を飛び越して緑の線の位置にスライド面を移動する．図 3 右側のように探索範囲が重なっていれば，その位置にスライス面を置いて処理を継続する．

この場合も，処理回数を均一にする必要があるため，1

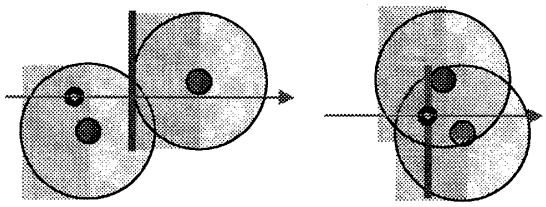

図3.スライス面の移動

画素で探索範囲を移動する回数を固定する．この回数を超えて奥行き方向に粒子を調べることはしない．

4.5 加算合成を利用したポテンシャル計算

画素ごとに，スライス面と視線との交点でポテンシャルを計算する．まず，粒子を一辺が影響半径の 2 倍の正方形で近似する．それを画像面に描画することで，その画素に影響を与える可能性のある粒子を求める．次に，粒子ごとに交点でのポテンシャルを計算し，加算合成することで，交点でのポテンシャルを計算する．GPU は，RGBA の 4 つの要素を一度に出力できるので， 4 つスライス面のポテンシャルを同時に計算する．GPU のマルチレンダーターゲット機能を使うと，n*4 個のポテンシャルを同時に出力することもできる．

4.6 境界面の計算

あるスライス面でポテンシャルが正になった場合，一つ前のスライス面との間に境界面があると推定できる．正しくは，この間で，ニュートン法のような数値演算手法を用いて，ポテンシャルが 0 になる位置を計算する必要がある．しかし，スライス間隔が十分に細かい場合には，前後のスライス面でのポテンシャルを線形補間して境界面を求めても，結果に大差は無い．そこで，処理が簡単な補間手法で計算する．

5. 実行結果

Geforce8800GTX ,Core2Duo E6600 , メモリ 1G のハードウェア環境で，DirectX9 と HLSL を使って提案手法を実装した．図 4 に示す実験結果では，粒子数を 4096 個，粒子半径と影響半径の比を 1 : 2.5 としている．一つの粒子のスライス面は 8 枚，探索範囲を移る回数は 10 回である．

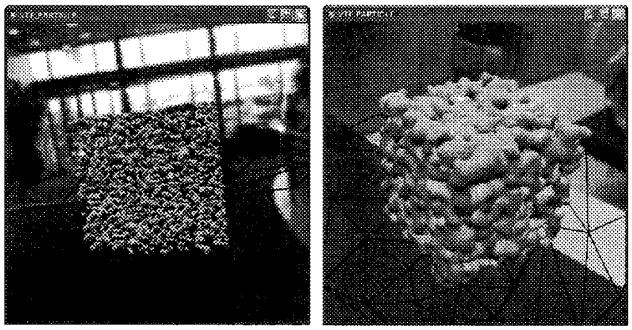

図 4. 実行例

図 4 左は影響半径を小さくした場合で，40FPS で描画できた．図 4 右は影響半径を大きくした場合で，メタボールの融合の様子がよりはっきりするが，ポテンシャル計算時の粒子の描画範囲が広がるため，速度が 20FPS まで低下した．どちらの場合も実時間表示できたが，画面での粒子領域が増えれば増えるほど処理速度が低下するので，現状ではこれが限界である．

6．今後の課題

空間分割などの方法で計算に不要な粒子を事前に取り除くことで，描画時間の短縮を図る．また，煙などの表示のために，関与媒質での光の減衰を考慮に入れた表示手法を検討する．

参考文献

(1) REEVES T., ACM Computer Graphics,pp. 91-108, 1983

(2) Kipfer, P., Graphics Hardware.ACM/Eurographics,pp. 115-122, 2004

(3) Kolb, A., Graphics Hardware.ACM/Eurographics,pp.123-131, 2004

(4) 山崎俊太郎, 他, “PC グラフィクスハードウェアを用いたスカラ場の等値面の高速描画法”,信学論 D-II, vol.J87, no.9, pp.1823-1833, 2004