

セル投影型並列ボリュームレンダリングの Early Ray Termination による高速化

1. はじめに

本稿では、セル投影法を用いた並列ボリュームレンダリング処理 (以下、PVR と呼ぶ) に Early Ray Termination (以下、ERT と呼ぶ)¹⁾ を適用して高速化を行なう手法を提案する。

2. セル投影型並列ボリュームレンダリング

2.1 セル投影法

セル投影法³⁾でのボリュームレンダリング処理は以下の3つのフェーズで構成される。(図1参照)。なお、以下では議論を簡単化するため対象とする非構造格子は四面体を仮定する。

(1) 投影 (projection) : まず、与えられた三次元データをスクリーンに投影する。

(2) スキャン変換 (scan conversion) : 次に、投影された個々のセルに対して、スキャン変換処理を行う。具体的には、セルが投影されたスクリーン上のピクセル $P(x, y)$ を通る視線がセルに入射後、通過するまでに、 $P(x, y)$ に与える寄与値 (色値 (RGB), および 不透明度 (α)) と、入射点と通過点の奥行き座標値 (それぞれ z_{front}, z_{back}) を求める。以下では、これら4つのパラメータで構成されるデータを ray-segment と呼ぶ。その結果、個々のセルが投影されるスクリーン上の各ピクセルに対して1つの ray-segment が計算される。

このスキャン変換処理を全セルに対して行った結果得られた ray-segment を、対応するピクセル毎に分類し、かつ、奥行き順に並べたリストを作成する (図2)。

(3) 合成 (composition) : 全セルのスキャン変換処理が完了した時点で、各リストで奥行き順に並べられた ray-segment を手前から順次合成することで、スクリーン上のピクセル値を得る。全ピクセルの合成が終わったとき、最終的な結果として三次元数値データを Volume Rendering した画像が得られる。

なお、スキャン変換処理によって作成されたリストにおいて、隣接する ray-segment のうち、一方の通過点と他方の入射点が一致するものは奥行き順で畳み込み合成する (これを部分合成と呼ぶ) ことが可能である (図3)。部分合成した結果もまた ray-segment となり、結果として ray-segment リストを縮小することができる。したがって、スキャン変換時に新しく ray-segment をリストに挿入する際、可能であれば部分合成を行なった上でリストに挿入することで、リストの長さを短くするという最適化が可能である。また、部分合成を行なった後の ray-segment の不透明度は、合成前の2つの ray-segment の不透明度より小さくなることはない。この性質を利用することで 後述の ERT の効率を上げることが可能である。

2.2 セル投影法の並列化

本研究で採用する並列化アルゴリズムは基本的には文献2)で提案された方式に準じたものである。

まず、並列計算機の各プロセッサにデータ (セル) を分配する。各プロセッサは、割り当てられたセルを scan convert し、自身の ray-segment リストを構築する。全プロセッサ

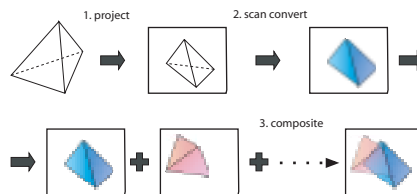


図1 セル投影法を用いたボリュームレンダリング処理



図2 ray-segment リスト

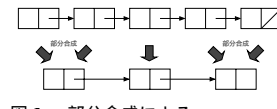


図3 部分合成による ray-segment リストの縮小

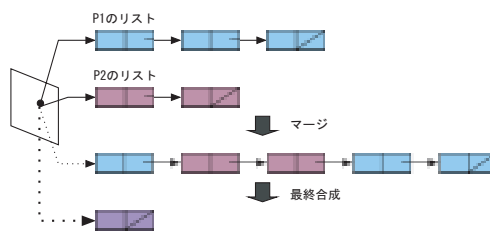


図4 ray-segment リストのマージ, 最終合成

が各々に割当てられたセルを全て scan convert し終わった後に、全プロセッサの ray-segment リストを、スクリーン上のピクセル毎に一つの ray-segment リストにマージする。その後、マージされたリストを先頭から順に合成していくことにより、リストを一つの ray-segment にする。この過程を最終合成と呼び、最終合成された ray-segment の色値が、そのピクセルでの色を表す。以上の結果、最終的に画像を得る (図4)。ray-segment リストのマージならびに最終合成では、スクリーンを複数の領域に分割し、各プロセッサに担当領域を割当てて並列処理を行なう。

マージする際、プロセッサ間ではスクリーンの解像度に応じた量の ray-segment リストが通信される。したがって、部分合成によってリストを短くできれば、通信量の削減につながり最終合成の高速化が図れる。

3. セル投影法への ERT の応用

Early Ray Termination (ERT) は、ボリュームレンダリング処理の高速化手法の一つで、レンダリングの対象の性質やレンダリング時に使用する伝達関数に依存するが、高速化に極めて有効であることが知られている¹⁾。ERT はレイキャスティング型ボリュームレンダリングにおいて多く用いられてきた手法であり、ピクセル毎に計算されるボクセルの寄与値累積計算過程において、ある点までの累積計算結果として得られる不透明度がある閾値を越えた場合 (以下では、この状態を terminated と呼ぶ) に、その点より奥にあるボクセルからの寄与は十分小さいと考え、それ以降の累積計算を打ち切ることで高速化を図る手法である。

今、セル投影型 PVR への ERT の応用について考えてみる。本来 ERT はスクリーン上のピクセル単位で行われる最適化であるため、セル投影型 PVR において ERT を活用

できるのは単純に考えると ray-segment の合成処理だけとなる。しかしながら、セル投影型 PVR に要する時間のうち、大部分を占めるのはスキャン変換処理である。したがって、スキャン変換処理を省くことができなければセル投影型 PVR において本質的な高速化につながらない。

そこで我々は、スキャン変換処理を ERT によって省略するアルゴリズム ERT-table 法を提案し、さらに、並列処理への応用について検討を行なった。

3.1 ERT-table 法

3.1.1 提案手法の概要

いま、図5のようなあるピクセルにおける ray-segment リストを考える。

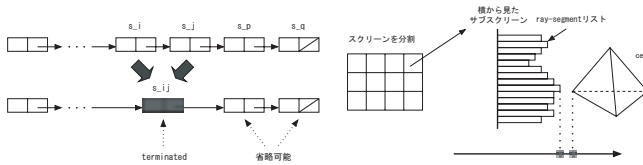


図5 部分累積不透明度による ERT

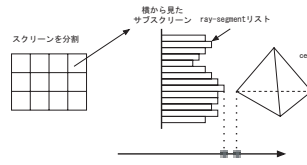


図6 ERT-table 法

ここで、ray-segment s_i, s_j は部分合成可能であり、部分合成を行った結果である ray-segment s_{ij} の不透明度が、十分不透明であるとする。このとき、 s_{ij} よりも後ろに生成される ray-segment は処理を省略することが可能である。この省略可能性判定は、 s_i よりも前にある ray-segment の状態とは無関係に行なうことが可能である。すなわち、あるピクセルでの ERT は、部分累積不透明度の値をもって行うことが可能である。

次に、あるセル c のスキャン変換処理を省略できる場合がどのような場合が考える。 c が投影された領域におけるスクリーン上の全ピクセルに対して、それまでにスキャン変換された、 c よりも前方に存在するセルを変換した ray-segment リストが存在し、なおかつそれらのリスト全てが terminated になっているならば、 c の投影された領域は既に全て terminated な状態であることになる。したがって、この場合に c のスキャン変換処理は省略可能である。

しかし、このようにピクセル単位で ERT 判断を行うことには、問題がある。各セルを投影した領域にある全ピクセルでの ERT 判断を行うには、スキャン変換を行うと同様の反復を繰り返さなくてはならない。ERT 判断を行うには、部分合成済みの ray-segment リストを先頭からたどる必要があるため、この方法ではスキャン変換処理をしているのと変わらない処理量がかかってしまう。すなわち、ERT のためのオーバーヘッドが大きすぎる。

そこで、厳密にセルの投影面のピクセル単位で ERT 判断を行うのではなく、セルを囲む程度の大きさのサブスクリーン単位で ERT 判断を行う方法を提案する。すなわち、各セル毎に、投影した全ピクセルに対して ERT 判断を行う代わりに、セルを囲むサブスクリーンの全ピクセルが terminated に達しており、なおかつサブスクリーンの全ピクセルのうちで最も Z 値が大きなもの (deepest) とセルの頂点のうちで最も Z 値が小さなもの (nearest) を比較することにより ERT 判断を行う。ERT 判定に必要なサブスクリーン単位の情報を表で管理する⁴⁾ことから、本手法を ERT-table 法と読んでいる (図6)。

3.1.2 利点・欠点

この方法では、あるセルの ERT 判断に要する処理は、1) セルを囲むサブスクリーンを見つけること、2) 当該サブスクリーン内の全ピクセルが terminated に達しているかの 1bit の真理値を調べること、3) サブスクリーンの Z 値とセルの Z 値と比較、といった単純な処理だけで済み、ERT 判定に要するオーバーヘッドが極めて少ない。

一方で、サブスクリーン内の全ピクセルが terminated でないとセルのスキャン変換を省略できないため、ERT によって省略できるセルの数はピクセル単位で ERT 判定を行う場合と比較して減少する。また、 Z 値が大きな ray-segment が先にスキャン変換されてしまうと、その ray-segment よりも手前にあるセルのスキャン変換が省略できない状況も発生する。ただし、後者に関してはセルを予め大雑把にソートしておくことで状況を改善することは可能である。

このように提案手法では、ERT 判定を緩めに (保守的に) 行なう代償として ERT 判定のオーバーヘッドを軽減し、これによりスキャン変換時における ERT 判定を可能とすることでセル投影型 PVR の高速化を図っている。

3.2 ERT 情報の共有

セル投影型並列 Volume Rendering に単純に ERT-table 法を組み合わせただけでは、ERT による効果が十分には発揮できない。これは、スキャン変換が各プロセッサで独立かつ並行して行なわれており、あるピクセルが terminate 状態になったことが他のプロセッサに伝わらず、本来省略できる領域について処理を継続して行うことによる無駄が生じるためである。

そこで本論文では、各プロセッサが ERT 情報の更新を非同期かつ独立して行うことを基本としつつ、定期的に各プロセッサの ERT 情報を交換することで ERT 情報を共有する手法を提案する。

4. まとめ

動的負荷分散機能を実装したセル投影型 PVR に提案手法を実装し、307,565 の要素から成る非構造格子データを、8 台の PC クラスタ (Pentium4 3GHz, GbE 接続) を使い、300² のスクリーン解像度でレンダリングした結果、提案手法を用いない場合に比べて、最良で 4.32 倍の性能向上を達成できることが分った。

参考文献

- 1) Marc Levoy. Efficient ray tracing of volume data. *ACM Transactions on Graphics*, 9(3):245-261, 1990.
- 2) Kwan-Liu Ma and Thomas W. Crockett. Parallel visualization of large-scale aerodynamics calculations: A case study on the cray T3E. In *Proc. IEEE Parallel Rendering Symposium*, pages 95-104, 1997.
- 3) Nelson Max, Pat Hanrahan, and Roger Crawfis. Area and volume coherence for efficient visualization of 3D scalar functions. *Computer Graphics (San Diego Workshop on Volume Visualization)*, 24(5):27-33, 1990.
- 4) 松井学, 竹内彰, 伊野文彦, 荻原兼一. 累積不透明度の伝播による並列ボリュームレンダリングの計算量削減. 電子情報通信学会 信学技報 (CPSY), 103(249):13-18, 2003.