

# FPGA を用いたフォトンマッピング法高速化の研究

久原 拓也

Takuya KUHARA

## 1 はじめに

コンピュータを用いて 3 次元空間を画像として描画する手法を 3 Dimensional Computer Graphics(3DCG) という。代表的な 3DCG 手法の一つに、視点から光線を逆追跡するレイトレーシング法がある。レイトレーシング法は、屈折光や反射光などの光環境を表現できる一方で、集光模様など、光源から間接的に光が届くことによって生じる現象は表現できない。これらの表現が可能な 3DCG 手法として、レイトレーシング法を拡張したフォトンマッピング法がある。フォトンマッピング法を用いることで、レイトレーシング法よりも更にフォトリアスティックな画像の描画が可能となるが、一方で、高品質な CG 画像の生成には膨大な計算必要となる。そのため、フォトンマッピング法をリアルタイムレンダリングなどの高速な画像生成が必要な場面で用いるには、何らかの高速化を行う必要がある。近年では、Graphics Processing Unit(GPU) や、Cell Broadband Engine(Cell/B.E.) などのメニーコアアーキテクチャも登場し、複数コアを利用したフォトンマッピングの高速化も研究されている<sup>1)</sup>。本研究では、Field Programmable Gate Array(FPGA) を用いた、フォトンマッピング法の高速化について検討を行った。

## 2 フォトンマッピング法

### 2.1 概要

フォトンマッピング法は 1996 年に H.K.Jensen によって考案されたレンダリング手法である<sup>2)</sup>。フォトンマッピング法では、視点からの光と、光源からの光を両方追跡することで、レイトレーシング法よりも複雑な光環境を表現することができる。フォトンマッピング法では、以下の 2 段階の処理手順で 2 つの追跡を実現する。

1. 光源から、ランダム方向にフォトンを放射する。放射されたフォトンが物体に衝突し、反射、屈折などを繰り返した後に、吸着する。この吸着した座標をフォトンマップと呼ばれる分布図に記録する。
2. レイトレーシング法により、各画素の色を計算する。この時、フォトンマップを基に各画素の明るさを計算する。

このように処理を 2 段階で行うことで、処理の複雑さを軽減している。第 2 段階における画素の明るさ決定は、

視線の到達地点(クエリ)と近傍フォトンを探査することで行う。フォトンマッピング法において高品質な画像を生成するには、多くのクエリ、フォトンについて計算を行わなければならないが、これには膨大な計算時間が必要である。そのため、フォトンマップから高速に近傍フォトンを探査する手法の研究が多くなされている。

### 2.2 kd 木

フォトンマップ探索処理を高速に行うには、3 次元空間中に拡散した無数のフォトンの中から与えられたクエリの近傍フォトンを探査できるようにフォトンマップのデータ構造を考える必要がある。このようなデータ構造として、広く採用されているものに、kd 木がある。kd 木は多次元空間の 2 分探索木である。空間の各軸を各ノードに対応させて、末端ノードを分割した空間に対応する。座標情報と、軸情報でマップを構成できるため、メモリ効率が良い。フォトンの探索は、クエリと各ノードの座標を比較し、クエリが kd 木内のどの領域に属しているかを調べることで行う。この処理において、クエリと、フォトンはそれぞれ独立であり、並列処理に向いているといえる。一方で、メモリ上のデータにアクセスする回数が非常に多く、CPU や GPU などのメモリアクセスの遅いアーキテクチャでは、高速化が図りづらい。よって、フォトンマップ探索処理の高速化には、並列処理が得意であり、独自のメモリアーキテクチャを定義できる専用ハードウェアが望ましい。本研究では、これらの条件を満たすアーキテクチャとして、FPGA を採用する。

## 3 FPGA を用いたフォトンマッピング法実装の検討

### 3.1 FPGA とフォトンマッピング法

FPGA は、内部構成を書き換え、柔軟に専用ハードウェアを再構成できるデバイスである。プログラムにより、内部に複数の演算コアを実装することが可能であり、並列処理に適している。そのため、並列性の高いアプリケーションの高速化に多く用いられている。一方で、メモリ容量では、数 GB の CPU に比べ、数 MB 程度と劣っているため、メモリリソースを多く消費するアプリケーションには不向きである。

FPGA へのフォトンマップ探索実装において問題と

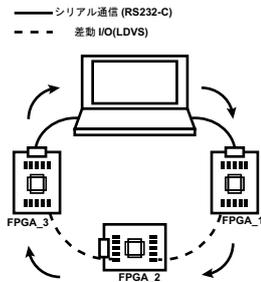


Fig. 1 FPGA システムの構成

なるのがkd木のメモリ容量である。kd木はメモリ効率に優れたデータ構造であるが、一般的なフォトンマッピング法では、数100万～数1000万ものフォトンを用いるため、多くのメモリリソースが必要となる。現在流通しているFPGAでは1台あたり50万～100万フォトン程度の格納が限界であるため、FPGAをフォトンマッピング法に用いるには、複数台のFPGAを用いる必要がある。

### 3.2 FPGA システムの実装

複数台のFPGAを必要とするアプリケーションを対象にFPGAシステムの設計と実装を行った。Fig.1に、FPGAシステムの構成図を示す。実装したシステムでは、通信モジュールを受信、送信で分離して動作させることで、高スループットでのデータ処理を実現している。また、送信、受信モジュールをロードしたFPGAをシステムに接続することで、FPGAの台数を拡張できるため、システムで扱う問題のサイズにあわせて、FPGAの台数を調整することが可能である。

### 3.3 フォトンマップ探索の検討

フォトンマップ探索処理をFPGAで実装した際、どの程度の効率化が図れるかを検討した。探索処理では、クエリとフォトンとの距離比較と、フォトンのソートが主な処理内容となる。検討方法として、CPU実装したkd木プログラムにおいてフォトンの取り出し、比較、ソートを単位時間で繰り返し、何個のフォトン进行处理することができるかを計測した。計測環境をTable.1に示す。実行パラメータとして、ランダムに生成した100万個の座標を元にkd木を構築し、探索フォトン数を500として、kd木に対してランダムに生成したクエリを与えた。この時、クエリの取り出し、比較、ソート時間を計測し、合計1秒を超えた段階で探索を打ち切った。この

Table 1 計測環境

CPU	Core i7 2600K (3.4GHz)
Memory	8GB
OS	Linux 2.6.38 x86_64
CPU code Compiler	gcc 4.4.5

条件の基で10回程測定を行ったところ、350万～500万個程度のフォトン进行处理できていることを確認した。また、そのうち、ソートを含むフォトンの処理数は200万～250万個程度であった。検討する対象のFPGAとして、200Mhz駆動のXilinx Virtex6(XC6VSX475T)を想定する。FPGAへの実装方法としては、kd木ではなく、FPGA内のブロックRAMブロックに各フォトンを設置し、二分探索を行う方法を想定する。1フォトンのデータ量は、以下のような、計32bitであると想定する。

- 各座標 (x,y,z)10bit
- 軸座標 2bit

Virtex6は36bit幅×1024の36KbのブロックRAMを最大1064個搭載することができる。これは、38,304Kbに相当し、100万個のフォトンのデータ量である26Mbを上回るため、全てのフォトンFPGA上に載せることができる。ブロックRAMはデュアルポートをサポートしており、1Clockあたり、2つのデータを取り出すことが可能である。データの取り出し処理と、比較処理はパイプライン処理可能なため、データの取り出し処理は間断なく行うことができる。Virtex6は最大200MHzで動作するため、1秒間に4億個のデータを取り出すことができる。一方で、フォトンのソート処理については、取り出し、比較処理に比べて大きな時間がかかる。まず、500個のフォトン単純ソートし、新しいフォトンについては、挿入位置を二分探索し、挿入位置以降のフォトン1つシフトする方法を取る。1個のフォトンとの比較に1Clockを要するとすると、挿入位置の決定に平均9Clockかかる。シフトは1Clock、挿入に1Clock程度で行うことができるため、合計11Clockが1フォトンあたりに必要となる。これは、1秒間に1800万フォトンの取り出しが行えることになる。すなわち、200万～250万個のCPUに対して、9倍程度の速度でフォトンの探索が行えることになる。よって、FPGAに対するフォトンマップ探索処理の実装においては、単純な二分探索でも、CPUに対して大きな高速化が期待できる。

### 参考文献

- 1) 大西信寛, 鎌田俊昭, 西川由理, 設楽明宏, 吉見真聡, 藤代一成, 天野英晴. Cell broadband engineを用いた photon mappingの実装と評価(2010年並列/分散/協調処理に関する『金沢』サマー・ワークショップswopp2010). 電子情報通信学会技術研究報告. CPSY, コンピュータシステム, Vol. 110, No. 167, pp. 19–24, 2010-07-28.
- 2) Henrik Wann Jensen. Global illumination using photon maps. In *Proceedings of the eurographics workshop on Rendering techniques '96*, pp. 21–30, London, UK, 1996. Springer-Verlag.