

CUDA と OpenGL を用いた三次元音響数値解析の GPGPU リアルタイム可視化——PMCC (Permeable Multi Cross-section Contours) の提案と評価——

河田 直樹[†] 大久保 寛^{†a)} 田川 憲男[†] 土屋 隆生^{††}
石塚 崇^{†††}

High-speed GPGPU Visualization for Three-dimensional Acoustic Simulation Using CUDA and OpenGL——Proposal and Evaluation of PMCC (Permeable Multi Cross-section Contours)——

Naoki KAWADA[†], Kan OKUBO^{†a)}, Norio TAGAWA[†], Takao TSUCHIYA^{††}, and Takashi ISHIZUKA^{†††}

あらまし 本研究では、CUDA と OpenGL を用いて GPU アーキテクチャによる三次元音響数値解析のリアルタイムの可視化（計算と同時に可視化を行う音響シミュレーション）について検討を行った。三次元音場の可視化法として、PMCC を提案し、その評価を行った。PMCC は従来の三次元空間中の複数断面を表示する (Multi Cross-section Contours) 方法に不透過度を組み合わせた非常にシンプルな可視化方法である。また、PMCC は波動現象特有の干渉や散乱の把握もボリュームレンダリングに比べて分かりやすく、また、描画のための計算負荷も非常に少ない、平行な複数断面表示を行う場合もオクルージョンがほとんど発生せず、断面表示でありながら、三次元空間を一目で把握できる。まさに、擬似的三次元表示といえる。描画速度の評価より、CPU を用いる場合に比べて、GPU によって 33 倍程度の加速化が可能であることが分かった。GPU 計算と PMCC 描画の組合せにより、三次元音響数値解析のリアルタイムの可視化の実現可能性を明らかにした。

キーワード GPGPU 可視化, FDTD, 音響数値解析, OpenGL, CUDA

1. ま え が き

近年、GPU (Graphics Processing Unit) を用いた科学計算の研究・開発が進んでいる。大きな理由はその高いコストパフォーマンスであり、今後、生体内音波伝搬解析や建築音響・騒音解析などの音響シミュレーションへの応用・実用化が期待されている。

従来、計算の高速化という点では、スーパーコンピュー

タやクラスタなどの大型の計算機が利用されているが、最近では GPU を用いて汎用的な数値計算を行おうとする GPGPU (General Purpose computation on GPUs) が様々な分野で注目され、盛んになってきている。GPU の演算性能は数年前より飛躍的に向上しており、最新の GPU は少し前のスーパーコンピュータ並の性能を秘めている [1], [2]。これまで、GPGPU の試みとして [3]~[6]、流体計算や津波シミュレーションを先駆けとして [7]、電磁界や音場という時間領域の波動数値シミュレーションの GPU への実装が行われて [8]~[11]。

これまで GPU では単精度計算を原則としていたが、最近の GPU では倍精度計算も可能となっており、今後、数値計算分野での利用が盛んになることが予想される。特に最近では、Fermi アーキテクチャ^(注1)が発表され、1GPU 当りのコア数が増え、キャッシュなど

[†] 首都大学東京システムデザイン学部, 日野市
Faculty of System Design, Tokyo Metropolitan University,
6-6 Asahigaoka, Hino-shi, 191-0065 Japan

^{††} 同志社大学理工学部, 京田辺市
Doshisha University, 1-3 Tatara Miyakodani, Kyotanabe-shi, 612 0321 Japan

^{†††} 清水建設株式会社技術研究所, 東京都
Shimizu Corporation, 3-4-17 Etchujima, Koto-ku, Tokyo,
135-8530 Japan

a) E-mail: kanne@sd.tmu.ac.jp

も大幅に改善されたため、更に GPU 計算の汎用性が広がると思われる。

さて、このように GPGPU は多くの分野で急速に広まりつつあるが、特に GPU 計算がパフォーマンスを発揮するアプリケーションは何だろうか。これは、今後 GPU による計算が普及するために非常に重要なことである。

筆者らは、この答えを GPU 高速計算中のリアルタイム可視化に見出そうと考えている。なぜなら、GPU はもともとグラフィックス用のハードウェアであるため、数値解析結果の高速可視化という面でも、大きなアドバンテージを有しているからである。すなわち、数値解析中に GPU 出力を超高速度で行うことで、高速リアルタイム可視化を可能とする。これは、個人使用においてタイムシェアリングが基本である大型スーパーコンピュータでは、実現はかなり難しいのではないかと考える。まさに、パーソナルパソコンを実現する GPU 計算環境に適している。

この高速可視化をする上で、相性の良い手法としては FDTD (Finite-Difference Time-Domain) 法 [12] や GCIP (Generalized CIP) 法 [13] などの時間領域手法である。なぜならば、これらの手法は、離散化した音響空間を、時間領域で逐次更新していくため、更新結果を高速に表示することができれば、音響伝搬の様子を計算と同時にそのまま瞬時に把握することができるのである。また、GPU による高速化の効果は並列化効率に依存するため、並列化の効果が大きいこれらの手法は GPGPU リアルタイム可視化として実用に最も近い。

一般には、数値解析結果をディスプレイ上にリアルタイムで可視化しようとする場合、解析結果を色情報へ変換したのちにビデオカード上の描画用のビデオメモリ (VRAM) 領域に転送する。それゆえ、CPU を用いて音場計算をした場合、解析結果が格納されているメインメモリ (RAM) から VRAM への転送が必要であるため、表示させる結果によっては PCI インタフェース [14] の転送が問題となる。しかし、計算に CPU を使わず、GPU のみで処理している場合、数値解析結果はもともと VRAM 領域に格納されているため、描画に必要な色情報の伝送は非常に高速に処理される。

本研究では、Fermi アーキテクチャを搭載した GPU GeForce GTX シリーズを用いて、これまで筆者らが報告してきた CUDA を用いた GPU 音響計算に、

描画用の API として OpenGL を組み合わせることで、GPU による音響数値解析の高速可視化の検討を行う。また、三次元の音波伝搬シミュレーションの可視化方法について、まず従来の可視化方法をまとめ、これらの問題点を考察した上で PMCC (Permeable Multi Cross-section Contours) を提案・評価する。

2. GPU による音響数値解析

2.1 FDTD 法

本論文では、リアルタイムの高速可視化が目的であるため、比較的定式化が簡単で計算負荷の少ない FDTD 法を用いる。FDTD 法は Staggered grid を用いて支配方程式を直接差分化する手法である [12]。離散化された音響空間では音圧と粒子速度は $1/2$ グリッド離れた点に配置される。最も基本的なスキームは、時間・空間二次精度の中心差分近似によって解く手法 (Yee-FDTD 法) であり [12]、リーブフロッグアルゴリズムで各グリッドの音圧と粒子速度を交互に $1/2$ タイムステップごとに更新していく。

本手法は時間領域で解く手法であり、各グリッド上の値の更新は順序性がなく、場の値はそれぞれ独立して求めることができるため並列化に適した手法といえる。GPU の演算性能とメモリバンド幅を考慮すると、FDTD 計算のボトルネックはメモリ転送であり、メモリバンド幅に強く依存していると考えられる。

2.2 CUDA による GPU プログラミング

GPGPU の初期の段階では、コンピュータグラフィックス向けの専用言語を用いてプログラミングする必要があったため、変数の型に GPU 特有の型しか使えないなど汎用的なプログラムの記述はかなり困難であった。しかしながら、近年、GPGPU 向けのプログラミング言語として CUDA (Compute Unified Device Architecture) などの C 言語に近い言語が開発されたことにより、C 言語の知識だけで比較的手軽にプログラミング可能となった [15], [16]。本研究では GPU 計算コードの作成には CUDA を利用している。

本研究で用いた GeForce GTX 580 (GeForce GTX シリーズに属している) は 512 基のストリーミングプロセッサ (SP, CUDA Core と呼ばれる) で構成され、高速アクセス可能な共有メモリ (SM) をもってい

(注1) : Nvidia 社が近年発表した 2011 年 2 月時で最新の GPU アーキテクチャであり、そのコードネームをとり Fermi と呼ばれる。512 個のストリーミングプロセッサ (SP, または CUDA コア) を有するユニコアプロセッサである。

る。一方、ノートパソコン用に開発された 480M は 352 基の SP で構成されており、580 に比べると若干理論性能は低下する。

CUDA における最小実行単位はスレッド (Thread) と呼ばれる。更に 32 スレッドをワープと呼び、1 ワープの単位でマルチプロセッサ (MP) の中の多数の SP により並列実行される。また、CUDA ではスレッドのまとまりをブロック、ブロックのまとまりをグリッドと呼び管理している。グリッドは PC (ホスト側) から実行を指令する単位で、グリッド内の全スレッドは同じプログラム (カーネルと呼ぶ) を実行する。

GPU 上のグローバルメモリ (GM) から MP へのメモリ転送はかなり高速ではあるが、MP の並列計算に掛かる時間と比べるとデータ転送時間が総計算時間のかなりの部分を占めることになる。したがって、計算アルゴリズムによるが、頻繁に GM にアクセスする必要がある場合には、参照するデータをいったん SM へ転送してから演算を行う方が高速に処理できる。

CUDA で GPU プログラミングを行う場合、高速化の重要なポイントはブロック内のスレッドモデルをどのように構成するかと、レジスタや SM を活用することである。すなわち、複数回 GM にアクセスする必要がある場合には、参照するデータをいったん SM やレジスタへ転送し、メモリアクセスを極力低減させる必要がある。

2.3 OpenGL による GPGPU 可視化

続いて計算結果の可視化について述べる。従来の CPU 計算における可視化の方法としては、一般には、数値解析結果をディスプレイ上にリアルタイムで可視化しようとする場合、解析結果を色情報へ変換したのちにビデオカード上の描画用の VRAM 領域に転送する (図 1 参照)。このプロセスは、CPU を用いて音響数値計算をした場合、解析結果が格納されている RAM から VRAM への転送が必要であるため、表示させる結果によっては PCI インタフェースの転送が問題となる。

他方、CUDA で GPU 計算を実装する場合、OpenGL の関数を CUDA から直接呼び出すことができ、また、計算領域が VRAM 内に確保されているため、PCI インタフェースの転送をバイパスしてビデオカード上の描画用の VRAM 領域へ表示情報を書き込むことができる (図 2 参照)。これはリアルタイム可視化を目指す上で非常に大きな利点である。CUDA による GPU 計算によって、計算速度自体が高速化さ

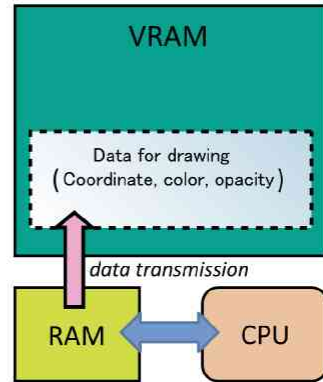


図 1 CPU 使用時の可視化
Fig.1 CPU visualization.

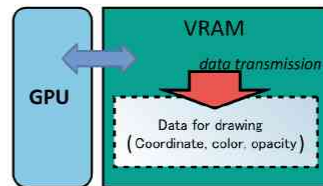


図 2 GPU 使用時の可視化
Fig.2 GPU visualization.

せられるが、CUDA と OpenGL の連携によって更に描画におけるアドバンテージも受けることができるのである。

3. 三次元の音場解析と可視化

3.1 GPGPU と高速可視化

従来、可視化を含めた音場解析を行う場合、計算環境への負荷を考え、対称性を仮定した二次元モデルや一次元モデルを用いた解析が行われてきていた。特に、パーソナルコンピュータ (パソコン) で解析する場合には、三次元空間をモデル化するためのメモリ容量の問題や、メモリが確保できたとしてもその計算を行うための計算時間を考慮すると、現実問題として三次元解析は困難な場合が多かった。

更に、シミュレーションをしながら、同時に解析結果を可視化する、いわゆるリアルタイム可視化を行う場合は、十分な描画スピードを維持するためには、三次元解析はパソコンレベルではほぼ不可能であった。

しかし、近年の計算機技術の発展 (特に、GPU による高速並列計算) は、描画スピードの問題を一気に解決し、三次元の音響シミュレーションとそのリアルタイム可視化を現実のものとしようとしている。すなわ

ち、GPU を用いたパーソナルスーパーコンピュータが同時可視化シミュレーションを大きく後押しすることになる。今後、この GPU パーソナルパソコン [1] は、三次元音響シミュレーションのリアルタイム可視化において、中心的な役割を担っていくことと考えている。

3.2 従来の三次元音場の可視化方法

従来、三次元音場の計算結果を可視化する場合、初歩的な手法として三次元空間の一つの断面を表示する方法 (図 3(a) 参照) やこの断面の見る角度を変え、 z 軸方向に音圧などの強度を割り当てる (サーフェスプロット) 方法 (図 3(b) 参照) などが挙げられる [17]。この表示方法は比較的描画のための計算負荷や転送負荷が小さく、ある特定の面のみが見たい場合には有効であるかもしれない。しかし、三次元空間を計算したにもかかわらず、二次元の情報しか表示できていないため、三次元での音場の広がりを理解するには効果的な方法とは言えない。

次に示す方法は、図 3(a) の発展として、三次元空間の複数の断面を同時に表示する方法 (図 4 参照) である [17]。ここでは、例として (a) では原点を通る xy 面、 yz 面及び zx 面を、(b) では z 軸に垂直な面を 2 面と yz 面を表示している。この方法は三次元の情報のうち、見たい範囲を中心に複数の断面を同時に見ることができ点で図 3 よりも三次元での音場の広がりが分かりやすい。しかし、同図より分かるように、ある一つの面の裏側の情報は見ることができない (オクルージョンが発生する)。したがって、平行な複数の面を表示する場合には、このオクルージョンの影響で効果が半減してしまう。

そこで、近年では、不透過度 (不透明度) を利用するボリュームレンダリング (VR) なる可視化法が提案されている [18], [19]。三次元音場の可視化にこのボリュームレンダリングを利用する場合、三次元空間の全領域における各ボクセルに対して、音圧値などの強度に合わせた不透過度を設定し、それをある視点からのパス上で積分することで三次元の空間を表す方法 (図 5 参照) である。

この方法は、近年、医療画像の表示や流体力学分野などで利用されることがある。特徴として、三次元音場の解析空間中の全ての情報を利用しており、三次元空間が一気に表示できる点でとても優れている。しかし、解析空間中に壁などの障害物があったり、閉空間を解析したりと複雑な干渉・散乱を繰り返す波動伝搬を可視化すると、視点によっては描画された結果が

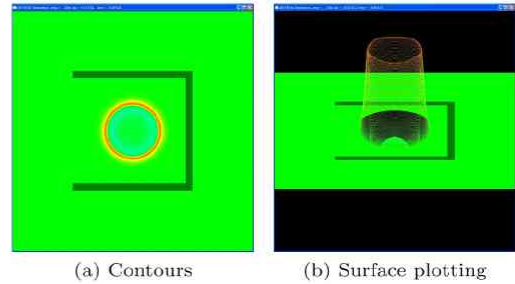


図 3 Cross-section Contour による表示
Fig.3 Visualization using Cross-section Contour.

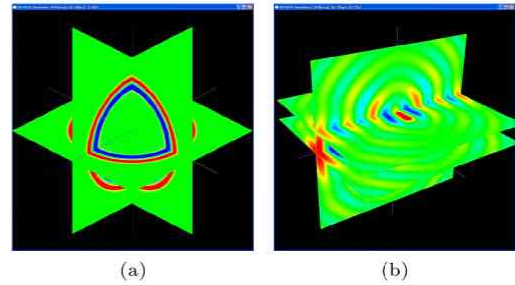


図 4 Multi Cross-section Contours による表示
Fig.4 Visualization using Multi Cross-section Contours.

複雑すぎて、かえって理解ができない場合がある。また、波動現象そのものは、もともと煙などのようにまとまった動きをすることは少なく、干渉・散乱を繰り返すことを考えると、ボリュームレンダリングを音響シミュレーションの可視化に利用し、しかも可視化の効果を明確にするには、かなりのノウハウが必要と考える。

加えて、この方法は三次元音場の各ボクセルの全ての情報を処理するため、描画のために多くの計算負荷がかかるという欠点もあり、後で示すとおり、結果として計算領域が同様な場合に他の手法に比べて描画スピードが低下する。

3.3 提案する可視化方法 (PMCC)

そこで、本研究ではこれらの問題を解決するための方法をとって PMCC (Permeable Multi Cross-section Contours) を提案する (図 6 及び図 7 参照)。図 6 は、解析空間の中心に単一の点音源を、右側奥に剛体壁を配置している。図中の (a) から (f) の順に時間経過となっている。図 7 は、解析空間の中心に点音源 4×4 個を配置している。図中の (a) から (c) は視点や表示面を三つのパターンで変更したものを示している。

この PMCC は一言でいえば、図 4 の方法において

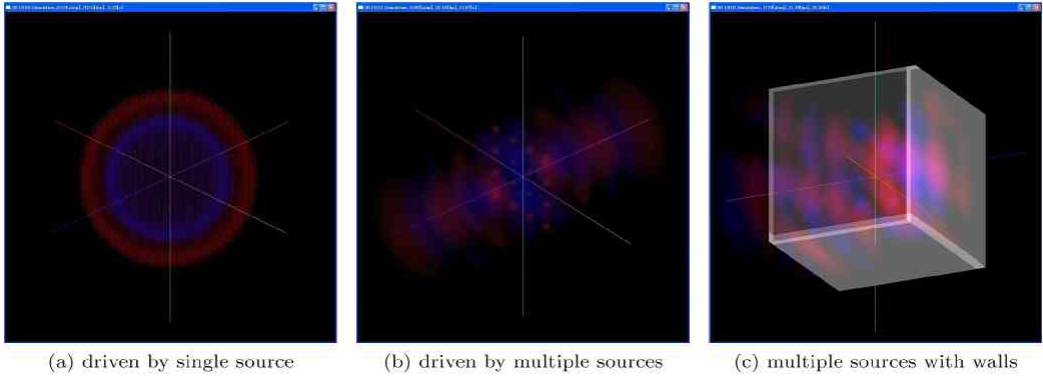


図 5 VR による 3D acoustic simulation
Fig.5 3D Visualization by volume rendering.

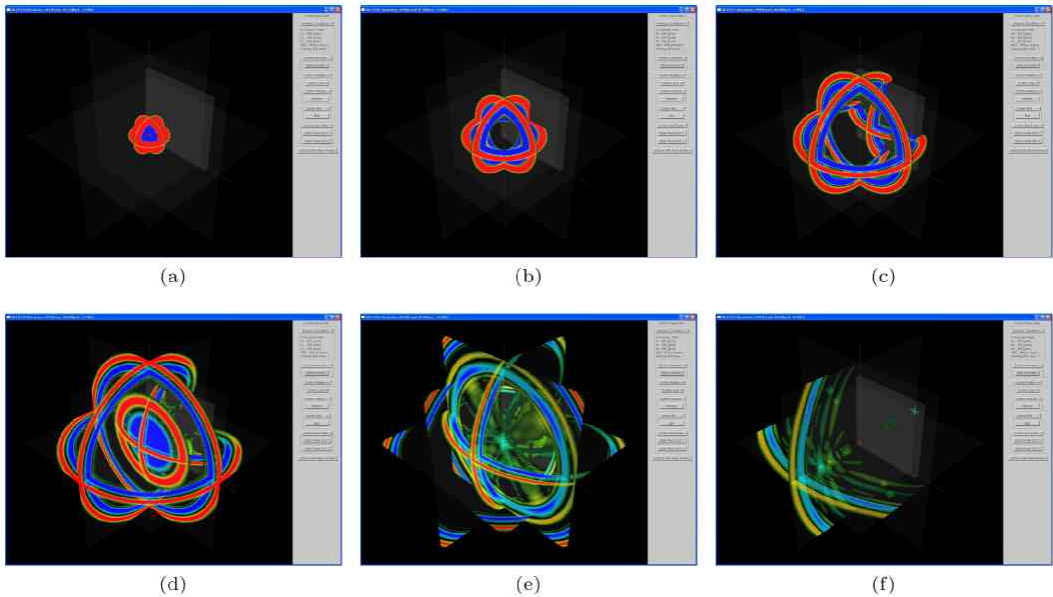


図 6 PMCC (Permeable Multi Cross-section Contours) による可視化
Fig.6 3D visualization by PMCC (Permeable Multi Cross-section Contours).

表示する複数の断面に対して、表示する強度に合わせて不透過度を設定する、方法である。

描画の手順をまとめると、以下ようになる。

(1) 三次元空間のある断面に対して、音圧値などをカラー表示する。

(2) 更に、音圧値の強度に合わせて不透過度 (α 値) を設定する (音圧値が大きい場合に不透過度は 1 に近づく)。

(3) 不透過度 (α 値) の与え方は計算対象によって変動させることができるので、表示させる強度の下限や強度と α 値の関係式などを自由に設定できる。

(4) 同様の手順で表示させたい断面を複数描画する。

特徴としては、

- 複数の断面を表示しても、音圧値が小さい点は不透過度が小さくなるため、オクルージョンが発生しにくく、ある断面の裏に隠れてしまっていた情報も見ることができる。

- したがって、平行な複数の断面も同時に表示可能である (図 6 (a) 及び (b))。

- 断面の回転及び移動も可能である。しかも、GPU 実装では高速に動作し、待ち時間なく回転・移

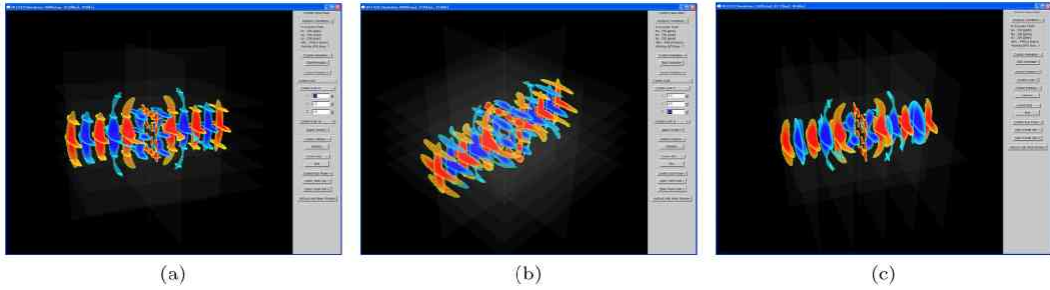


図 7 複数平行面を含む PMC による可視化 (Permeable Multi Cross-section Contours)

Fig. 7 3D visualization by PMCC (Permeable Multi Cross-section Contours) including parallel contours.

動することを確認している。

- 断面のみの不透過度を利用しているため、ボリュームレンダリングに比べて、描画のための計算負荷が極端に少ない。
 - 断面表示をもとにしているため、複雑な波動伝搬現象も把握しやすい。
- ということが挙げられる。アニメーションにより視点を変更することで、より三次元空間を把握することが容易になる [20]。

4. 数値解析結果 (各手法の描画スピードの比較)

本章では、各手法の描画スピードの比較を行う。使用した計算環境は OS : Windows XP Pro x64 edition, CPU : Intel Core i7 920 2.67 GHz, メモリ : DDR3 6 GByte, コンパイラ : マイクロソフト C/C++ コンパイラ Ver.15.00 for x64, OpenMP : OpenMP 2.0 である。以下では特に断らない限りは単精度型を用いて計算を行っている。本計算では GPU として GeForce GTX 580 及び 480 M を使用している。表 1 に GPU の主要スペックを示す。

解析領域 (グリッド数) に対する描画速度の結果を図 8 に示す。図中 (a) はボリュームレンダリングを用いた場合の三次元音響シミュレーションの描画スピードを示している。(b) には PMCC を用いた場合の三次元音響シミュレーションの描画スピードを示している。(b) には、あわせて CPU で同様の描画を行った結果を示している。解析モデルは図 5 (a) 及び 6 (a) で示したように中心に音源を配置したものである。同図では横軸がグリッド (セル) 数、縦軸が描画速度 (fps: Frames Per Second) としている。fps は 1 秒間に何

表 1 GeForce GTX 580 及び 480 M のスペック一覧
Table 1 GPU Specifications (GTX 580 and GTX 480 M).

主要スペック	GTX 580	GTX 480 M
CUDA Core 数	512	352
プロセッサクロック	1544 MHz	850 MHz
メモリクロック	2004 MHz	1200 MHz
メモリアインタフェース	384 bit	256 bit
メモリサイズ	1536 MByte	2 GByte
メモリタイプ	GDDR5	GDDR5

枚のフレームを表示できるかを示した指標であり、リアルタイムシミュレーションとしては 30 fps 以上を考えている (ワンセグ放送の描画速度は 15 fps であるので、その 2 倍にあたる)。

また吸収境界として、4 層 PML (Perfectly Matched Layer [21]) と 8 層 PML を適用した場合と壁面を剛体とした場合の三つの結果を表示している。4 層 PML と 8 層 PML では、解析領域が小さいほど、その影響が顕著であるが、ある程度解析領域が大きくなると一つの結果の差は小さくなっている。

更に、4 層 PML のモデルに対して、VR と PMCC を比較表示すると、PMCC の方が 2 倍弱描画スピードが速いことが分かる (図 9 参照)。この差が生じる理由としては、VR は三次元領域全ての不透過度を計算しているのに対して、PMCC は擬似的三次元表示であり、ある面のみデータを用いているため、データの転送負荷と計算負荷が小さいためである。

図 8 (b) を見ると、 256^3 グリッド (4 層 PML) の場合、GTX 580 において PMCC を用いると約 74 fps である。一方、本研究で用いた CPU : Intel Core i7 920 2.67 GHz にて OpenMP を使用し 8 スレッド並列化をした場合、同様の計算条件で PMCC を用いると約 1/33 倍となっており、GPU と CPU の可視化の性

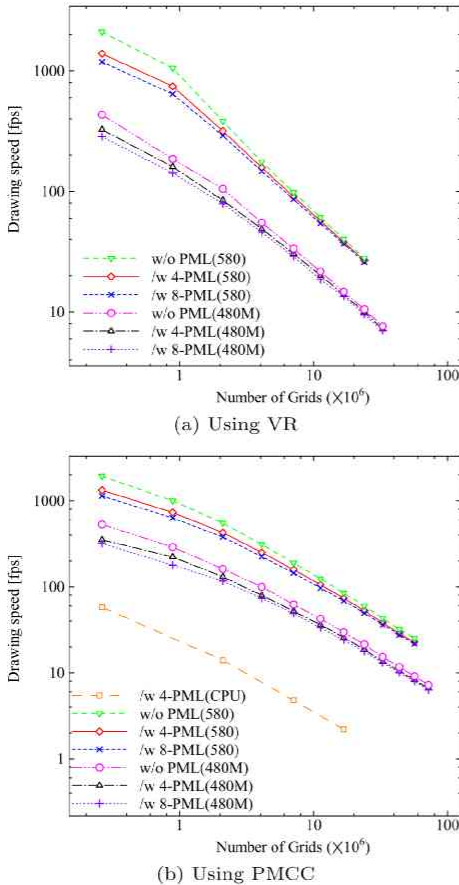


図 8 グリッド数に対する描画速度 (FDTD, 三次元音響解析)
Fig. 8 Drawing speed v.s. grid number.

能差は歴然である。本研究で用いた CPU 及び GPU でリアルタイム可視化の指標である 30 fps を満たすグリッド数を考察してみると、CPU では約 80^3 グリッド、GPU では 352^3 グリッドとなっている。以上より、CUDA と OpenGL を用いた GPU 計算と PMCC を用いた高速可視化によって、従来では考えられないような性能が得られることが明らかとなった。

5. むすび

本研究では、GPU を用いて三次元音響数値解析のリアルタイムの可視化 (計算と同時に可視化する音響シミュレーション) について検討を行った。三次元音場の可視化法として、PMCC を提案し、その評価を行った。PMCC は従来の三次元空間中の複数断面を表示する (Multi Cross-section Contours) 方法に不

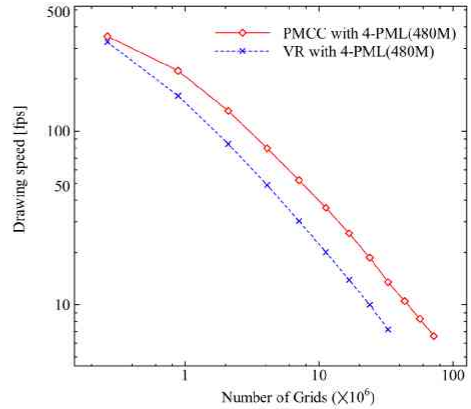


図 9 グリッド数に対する描画速度の比較 (FDTD, 三次元音響解析)
Fig. 9 Drawing speed v.s. grid number.

透過度を組み合わせた非常にシンプルな可視化方法である。

波動現象特有の干渉や散乱の把握もボリュームレンダリングに比べて分かりやすく、また、描画のための計算負荷も非常に少ない。平行な複数断面表示を行う場合もオクルージョンがほとんど発生せず、断面表示でありながら、三次元空間を一目で把握できる。まさに、擬似的三次元表示といえる。

本研究の結果より、三次元音響数値解析の高速可視化においては、GPU 計算と時間領域手法、そして PMCC を組み合わせて利用することは、非常に有用な方法であることが分かった。今後は、この可視化を生体情報のイメージング技術や室内音響解析などに応用したいと考えている。

文 献

- [1] 河田直樹, 大久保寛, 田川憲男, 土屋隆生, “音響数値計算のための GPU によるパーソナルスーパーコンピュータの実現へ向けて,” 音響春季講義集 2-2-6, March 2010.
- [2] 河田直樹, 大久保寛, 土屋隆生, “GPU 計算を用いた時間領域電磁界数値解析の高速化性能に関する比較,” 信学論 (B), vol. J94-B, no. 3, pp. 480-483, March 2011.
- [3] D. De Donno, A. Esposito, L. Tarricone, and L. Catarinucci, “Introduction to GPU computing and CUDA programming: A case study on FDTD,” IEEE Antennas Propag. Mag., vol. 52, no. 3, pp. 116-122, Jan. 2007.
- [4] M.D. Mccool, “Signal processing and general-purpose computing and GPUs,” IEEE Signal Process. Mag., vol. 24, no. 3, pp. 110-115, 2007.
- [5] S. Adams, J. Payne, and R. Boppana, “Finite difference time domain (FDTD) simulations using graphics processors,” IEEE Computer Society HPCMP Users

- Group Conference, pp.334–338, 2007.
- [6] J.D. Owens, D. Luebke, N. Govindaraju, M. Harris, J. Kruger, A.E. Lefohn, and T.J. Purcell, “A Survey of General-Purpose Computation on Graphics Hardware,” *Computer Graphics Forum*, vol.26, pp.80–113, 2007.
- [7] 青木尊之, “フル GPU による CFD アプリケーション,” *情報処理学会誌*, vol.50, no.2, pp.107–115, 2009.
- [8] T. Tsuchiya and H. Sekoguchi, “An evaluation method of calculation performance for acoustic field simulation using a graphic processing unit (GPU),” *J. Japan Society for Simulation Technology*, vol.27, no.4, pp.245–254, 2008.
- [9] 土屋隆生, 世古口久史, “GPU による時間領域音場シミュレーションの高速化～DHM と FDTD の比較,” *信学技報*, US2007-89, Jan. 2008.
- [10] 河田直樹, 大久保寛, 田川憲男, 土屋隆生, “GCIP(l, m)法を用いた音波伝搬シミュレーションと GPU 計算による高速化の検討,” *日本音響学会 AI 研究会, ウェブライブラリ AI-2009-21*, 2009.
- [11] 河田直樹, 大久保寛, 土屋隆生, “GPU を用いた電磁界数値解析の高速化に関する検討,” *信学技報*, A-P2009-141, Nov. 2009.
- [12] K.S. Yee, “Numerical solution of initial boundary value problems involving Maxwell’s equations in isotropic media,” *IEEE Trans. Antennas Propag.*, vol.AP-14, no.4, pp.302–307, May 1966.
- [13] K. Okubo and N. Takeuchi, “Analysis of an electromagnetic field created by line current using the CIP method,” *IEEE Trans. Antennas Propag.*, vol.55, no.1, pp.111–119, Jan. 2007.
- [14] OFFICE TAKASAKU, 徹底図解パソコンのしくみ—最新ハードウェアのテクノロジーから近未来パソコンまで, 新星出版社, 2006.
- [15] http://www.nvidia.co.jp/object/cuda_home_jp.html
- [16] 青木尊之, 額田 彰, はじめての CUDA プログラミング, 工学社, 2009.
- [17] 可視化情報学入門編集委員会, 可視化情報学入門, 東京電機大学出版局, 1994.
- [18] M. Levoy, “Display of surfaces from volume data,” *IEEE Comput. Graph. Appl.*, vol.8, no.3, pp.29–37, 1988.
- [19] R.A. Drebin, L. Carpenter, and P. Hanrahan, “Volume rendering,” *Comput. Graph.*, vol.22, no.4, pp.65–74, 1988.
- [20] <http://www.sd.tmu.ac.jp/klolab/gallery/3DV/PMCC/index.html>
- [21] J.P. Berenger, “A perfectly matched layer for the absorption of electromagnetic waves,” *J. Comput. Phys.*, vol.114, pp.185–200, Jan. 1994.

(平成 23 年 2 月 7 日受付, 5 月 31 日再受付)



河田 直樹 (学生員)

首都大・システムデザイン・情報通信システム卒。2010 年 4 月より首都大学東京大学院システムデザイン研究科博士前期課程に在学中。現在, GPU を用いた音響シミュレーションと可視化に関する研究に従事。



大久保 寛 (正員)

1999 東北大・工・電気卒。2001 同大大学院工学研究科博士課程前期課程了。2004 同後期課程了。2009 年 4 月より首都大学東京大学院システムデザイン研究科准教授。博士 (工学)。専門は波動情報工学。現在, 電磁界及び音響信号の解析・シミュレーションに関する研究に従事。2008 年会 (環境電磁工学研究会) 若手優秀賞。2007 IEEE AP-S Japan Chapter Young Engineer Award 受賞。



田川 憲男 (正員)

1986 都立大・工・電気卒。1989 東工大大学院・総理工・電子システム専攻修士課程了。同年, (株) 富士通研究所入社。1992 都立大・工・助手。都立大・工・助教授を経て, 2007 年より首都大学東京・システムデザイン学部・教授。博士 (工学)。コンピュータビジョン, 統計的信号処理, 医用超音波工学の研究に従事。



土屋 隆生 (正員)

1984 同志社大・工・電子卒。1986 同大大学院工学研究科博士課程前期課程了。1989 同後期課程了。工博。富山大, 岡山大, 秋田県立大を経て, 2004 同志社大工学部 (現理工学部) 教授。専門は数値音響工学。現在, 音波, 弾性波などの数値解析に関する研究に従事。1990 日本音響学会粟屋潔学術奨励賞, 1991 日本シミュレーション学会研究賞受賞。



石塚 崇

1999 九州芸術工科大・芸術工・音響設計卒。2001 同大大学院芸術工学研究科博士課程前期課程了。2004 同後期課程了。2005 年 11 月より清水建設技術研究所勤務。博士 (芸術工学)。専門は建築音響及び騒音制御。建築空間周辺の音環境改善及び音響数値解析に関する研究に従事。2002 日本騒音制御工学会研究奨励賞受賞。