

Fermi GPU を用いたディレクティブによる FDTD 高速計算

Accelerataion of FDTD Method on Fermi GPU Using Directive Compiler

園田 潤¹⁾, 佐藤 源之²⁾

Jun Sonoda and Motoyuki Sato

1) 仙台高専 知能エレクトロニクス工学科 (〒 989-3128 宮城県仙台市青葉区愛子中央 4-16-1)

2) 東北大学 東北アジア研究センター (〒 980-8576 宮城仙台市青葉区川内 41)

Key Words: Directive compiler, CUDA, FDTD method, Fermi GPU

1 はじめに

近年, GPU を用いた FDTD 法の高速化に関する研究が盛んに行われており, CPU に比べ数十倍高速化できることが報告されている [1, 2, 3]. これまでの FDTD 法の GPU 実装に関する研究では, C/C++ の GPU 開発環境である CUDA を用いたものがほとんどであった. しかしながら, CUDA ではソースコードを書き換える必要があるため, GPU への移行が容易でないことなどの問題があった. そこで本研究では, コードに指示文を挿入するだけで容易に GPU が使用できるディレクティブによる FDTD 法の GPU 実装の検討を行う.

2 ディレクティブによる FDTD 法の GPU 実装

ディレクティブは, ソースコードに数行の指示文を挿入するだけで GPU 用の実行プログラムが得られたため, CUDA のようにソースコードを大きく書き換える必要がなく, GPU への移行が容易であるという特長がある. しかしながら, コンパイラが CUDA コードを自動生成するので, 直接 CUDA コードを書く場合に比べ性能は劣ることが知られている.

GPU における FDTD 法の計算手順は, (1) 電磁界値や誘電率などの電気定数と変数を GPU にアップロード, もしくは GPU 内のローカル変数に設定し, (2) CPU は GPU 上にカーネルを起動することで電磁界値を GPU 上で計算し, (3) 計算結果を GPU から CPU へ転送する.

3 Fermi GPU における FDTD 法の計算時間

ディレクティブの一つである PGI コンパイラ pgcc 10.9 と CUDA nvcc 3.1 による, 問題サイズ 256^3 の 3 次元 FDTD 法の 1000 ステップに要する計算時間を評価する. 解析モデルは, 中央に設置された点状電界ハードソースによる電波伝搬問題であり, 周囲を完全導体で囲んだ場合と, Mur 1 次吸収境界条件を適用した場合について検討する. 使用する GPU は Tesla C2070 であり, 比較に用いた CPU は Intel Core i7 980X

3.33 GHz の 1 コア, コンパイラは gcc 4.4.3 で -O3 をオプションとする. OS は Ubuntu 10.04 server 64 bit である.

表 1 に計算時間を示す. 表 1 には, CPU の計算時間 t_C , GPU の計算時間 t_G とそのブロック構成, GPU と CPU の計算時間比 t_C/t_G を示している. ここで, CUDA コードでは共有メモリは使用せず, コンパイラ検出による Fermi GPU のキャッシュメモリを使用している. 表 1 の結果より, 完全導体で囲んだ問題では, CPU の 1 コアに比べ, 単精度ではディレクティブで 22 倍, CUDA で 62 倍, 倍精度ではディレクティブで 13 倍, CUDA では 19 倍の高速化が得られた. また, Mur 1 次吸収境界条件を適用した単精度では, ディレクティブでは 19 倍, CUDA では 49 倍が得られた. また, ディレクティブと CUDA を比較すると, 単精度では 40 % 程度, 倍精度では 70 % 程度の性能であった.

4 まとめ

本稿では, ディレクティブによる FDTD 法の GPU 実装を検討した. CPU Core i7 980X 3.33 GHz 1 コアとの計算時間を比較した結果, 吸収境界条件を付加した問題においても, Tesla C2070 ではディレクティブでも 20 倍程度の性能が得られた. また, CUDA と比較した結果, 単精度では 40 % 程度, 倍精度では 70 % 程度の性能が得られた.

今後の展開として, 地中レーダ探査など実際の問題への応用 [3], 様々な種類の GPU で評価すること等が挙げられる.

謝辞

本研究の一部は, 科研費 (22760284) および電気事業連合会・パワアカデミー萌芽研究の助成を受けている.

参考文献

- [1] P. Sypek, et al., IEEE Magnetics, pp.1324–1327, March 2009.
- [2] 高田, 下馬場, 増田, 伊藤, FIT 2009, pp.457–462, Sept. 2009.
- [3] 小関, 園田, 佐藤, 2011 信学総大, March 2011.

表 1: ディレクティブと CUDA による FDTD 法の計算時間 (時間 1000 ステップ, 問題サイズ 256^3)

precision	boundary	CPU t_C (s)	Directive (pgcc 10.9)			CUDA (nvcc 3.1)		
			GPU t_G^P (s)	block size	t_C/t_G^P	GPU t_G^C (s)	block size	t_C/t_G^C
single	PEC	1289.25	57.76	128×1	22.32	20.81	256×2	61.95
double	PEC	962.85	75.49	128×1	12.75	51.76	128×4	18.60
single	Mur 1st	1317.98	67.84	$128 \times 1(16 \times 16)$	19.42	27.09	$256 \times 2(16 \times 16)$	48.65