

NA-12 青井真 (防災科学技術研究所)

GPGPUの地震ハザード予測シミュレーションへの適応性評価



1. はじめに

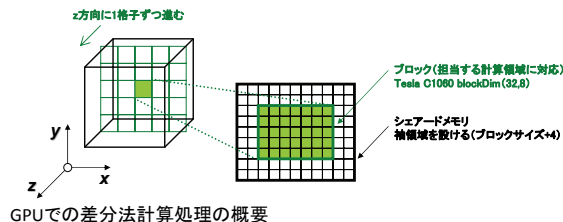
今や地震ハザード評価の標準的なツールとなった3次元差分法は、近年の計算機環境の劇的な進歩の恩恵を受けたとはいえ、大きな計算機リソース(CPUパワー及びメモリ)を必要とし、実務等で一般的に使用可能な計算機では数日以上の計算となる事もしばしばである。このような状況を打開する手段として、多くの数値計算分野で広く使われるようになりつつあるGPGPU (General Purpose Computation on Graphics Processing Unit) の利用 (例えば、青木尊之・額田彰, 2009) が考えられる。

青井・他(2009)は、食い違い格子を用いた空間・時間共に二次精度の差分演算子による速度・応力型の定式化 (e.g. Virieux, 1984) を用いた単純なモデルによる予備的な実験を行い、単一CPUの場合に比べGPU (NVIDIA Tesla D870) による計算時間は1/17.8に短縮されることを示した。

本研究では、不連続な食い違い格子 (Aoi and Fujiwara, 1999) を用いた、空間四次・時間二次精度の差分演算子による実用コードであるGMS (Ground Motion Simulator, 青井・他, 2004) をベースにGPU化を図った。

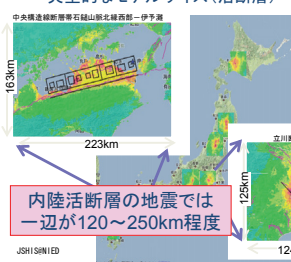
【GPU開発環境】

ハードウェア		SGIU Asterism AQ232		HP Z800	
CPU		Dual-Core AMD Opteron 8222 3.0GHz x4 (8 core)		Quad Core Intel Xeon E5520 2.26GHz	
メインメモリ		DDR2 667MHz 4GB x 4 (16GB)			
GPU	浮動小数点演算性能 (1GPU当たり)	500GFLOPS/GPU			933GFLOPS/GPU
	メモリ容量 (1GPU当たり)	1.5GB/GPU		Tesla C1060 (1GPU)	4.0GB/GPU
	メモリバンド幅 (1GPU当たり)	76.8GB/s/GPU			102.0GB/s/GPU
ソフトウェア					
Compiler		NVIDIA CUDA 2.2		NVIDIA CUDA 2.2	



GPUでの差分法計算処理の概要

典型的なモデルサイズ (活断層)



シングルCPUでは、例えば
計算時間15日 & 必要なメモリは33GB

大きなモデルだと計算
時間が数ヶ月以上

◎アルゴリズムの最適化:

・格子の最適化 (不連続格子・不規則格子: 数倍)

◎並列計算:

・WS (マルチコア、マルチCPU: 数倍)

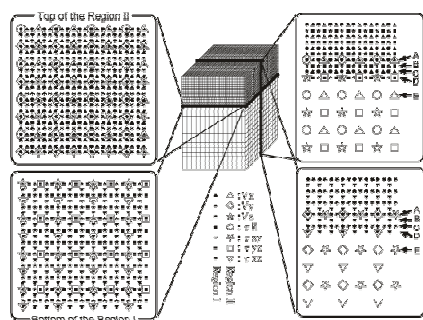
・クラスター (数倍~数十倍) ~ スーパーコンピュータ (数百倍)

2. 不連続格子を用いた差分法による波動場の計算

差分法により地震波伝播シミュレーションを行う場合、均質な大きさの格子を用いる限り、格子サイズは計算すべき最低波長により決定される。低速度の媒質がモデルのごく一部にのみ存在する構造を扱う場合でも計算領域全体を小さな格子に分割せざるを得ず、大規模なモデル計算を行う際には大きな障害となる。実地震の再現計算や地震ハザード評価を行う際には、堆積平野構造を有するモデルなど地表に近いほど極端に地震波速度が遅い構造を取り扱うことが多いため、上記のような問題を回避するためには地震波速度構造に合わせた不均質な格子を用いることが有効である。

Aoi and Fujiwara (1998) や青井・他 (2004) は、大きな異なる格子を組み合わせることにより効率的かつ高精度に計算を行うことの出来る不連続格子による差分法の定式化を提案した。

下図に示すように、浅い部分 (領域I) の格子点間隔は細かく、深い部分 (領域II) の格子点間隔は領域Iの3倍の粗い格子点間隔を有する格子モデルを用いている。2つの領域はオーバーラップしており、波動場の連続性が保たれるよう内挿される。典型的な盆地構造モデルの計算において、均質な格子による場合と比較し数倍から十数倍程度効率が良いことが分かっている。



(中央) 計算に用いる不連続格子。(右) 不連続格子の垂直断面。領域と領域IIの接続部分で、内挿のために格子が重なっている。(左) 領域IIの最上面 (A面) と領域IIの最下面 (D面) における不連続格子の水平断面。

3. 実装の概要

GPU版プログラムでは、ブロックを2次元で構成し、計算の対象となるx-y平面にブロックを敷き詰めるように配置している。ブロック内の各スレッドはz方向の始点から終点まで1格子点ずつ計算処理を進め、3次元領域全体の計算処理を行う (右上图)。

差分法の計算処理では計算の対象となる格子のデータの他に隣り合う格子のデータが必要となるが、各スレッドが隣り合う格子のデータをグローバルメモリから読み込んだ場合、スレッド間では重複して同じデータが読み込まれることになる。グローバルメモリへのアクセスはレジスタやシェードメモリと比べ低速であるため、このような重複したデータの読み込みは性能低下の要因となる。

GPU版プログラムでは、各スレッドが担当する格子のデータをシェードメモリに読み込み、隣接する格子点のデータを参照する際にはシェードメモリに読み込んだデータを利用することによりグローバルメモリへのアクセス回数の軽減を図っている。CPUで言うところのキャッシュのような機能をシェードメモリが行っている。

4. 計算速度性能測定

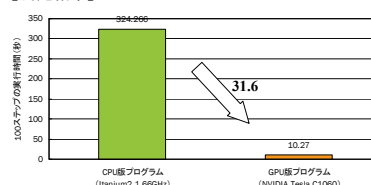
【測定条件】

データ	計測条件 (共通)
ex2a	
格子数	第1領域 (1020x720x38) 第2領域 (340x240x105) =36,475,200格子
ステップ数	100
浮動小数点演算精度	単精度 (32bit)
測定範囲	タイムステップのループの差分計算部分のみを対象とする (ファイル入力/出力、初期化処理等の計算は含めない)

計測条件 (CPU版プログラム)	
実行環境CPU	Itanium2 (1.66GHz)、ifort 10.1 -O3 ※1コアを使用

計測条件 (GPU版プログラム)	
実行環境GPU	Tesla C1060、nvcc2.2
実行環境CPU	Xeon E5520(2.27GHz)、 gfortran 4.1-O3、g++ 4.3-O3

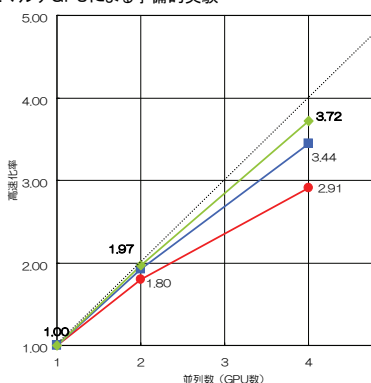
【測定結果】



対象	浮動小数点演算回数	GFLOPS	理論性能比 (%)
all	7.663E+09	69.5	7.4

性能		
CPU/GPU	100step	1stepあたり
CPU	324.266s	3.243s (3243ms)
GPU	10.270s (31.6x)	0.103s (103ms)

5. マルチGPUによる予備の実験



5. まとめ

地震波動シミュレーションの標準ツールの一つであるGMS (Ground Motion Simulator, 青井・他, 2004) を、GPUに移植し性能試験を行ったところ、CPU版プログラムと比較し30倍を超える計算速度が得られた。実用的なコードにおいても高い計算効率を得られたことから、スーパーコンピュータや大規模クラスターが使用しにくい実務や研究等においても規模の大きなシミュレーションが可能となることが期待される。

青井・他 (2010) は、均質格子差分法を用いた単純なモデルにおいてマルチGPUによる予備的な実験を行い、4つのGPUを用いて1つ場合の3.44倍の性能を得ることに成功している。GMSにおいても、実用的な規模の計算を行うため、今後、マルチGPUでの実行に対応する予定である。

実用的な使用目的においては比較的大量の計算結果をファイル出力する必要がある場合が多く、ビデオメモリとメインメモリ間のデータ転送が必要となるため、高速な計算のボトルネックになることが懸念される。計算と出力を同時にしない、出力に要する時間を見かけ上隠蔽するなどの工夫が必要となるであろう。