

GPU の実用数値アプリケーションの適用性

境界要素法の実装 *

和木 康祐[†] 矢吹 太朗 佐久田 博司

青山学院大学大学院理工学研究科[‡]

1 はじめに

数値シミュレーションの適用範囲の拡大に従って、数値計算リソースの需要が増大している。そこで、リソースの一つとして、画像処理用のハードウェアである GPU (Graphics Processing Unit) が注目されている。GPU は、コンピュータゲームなどにおける高速な 3D グラフィックスの描画への需要の高まりなどを背景に急速な発展を遂げ、演算性能において CPU を大きく凌駕している。また、大手 GPU ベンダーである NVIDIA 社より、GPU を汎用数値計算に適用するための「CUDA」技術 (3 章で解説) が提供された [1]。このようなことから、GPU のグラフィックス処理以外の数値計算への活用が期待されている。

2 研究目的

本研究では、GPU を利用した高速な数値計算の実用例として、境界要素法 (4 章で解説) による数値解析アプリケーションの作成とその評価を行う。アプリケーションは、CUDA を用いて開発を行い、その概要は、非定常熱伝導問題を境界要素法を用いて解を求め、逐次的にグラフィックスの出力を行うというものである。そして、開発したアプリケーションを用いて、CPU のみで実行した場合と、CPU に加えて GPU を用いて実行した場合との比較と評価を行う。

3 GPU を汎用数値計算へ適用する背景

3.1 GPU の性能向上

近年、GPU は複数のコアにより、浮動小数点演算能力において CPU を大きく上回っている。過去 4 年間の CPU と GPU の性能向上の比較を図 1 に示す。そして、Direct3D などのグラフィックス API を通して、ユーザーが処理内容をプログラミング可能になった。さらに、より

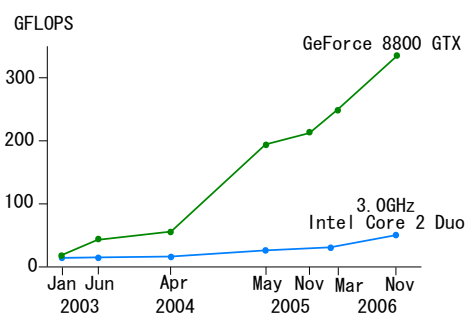


図 1 過去 4 年間の CPU と GPU の性能向上 [2]

汎用的な活用のための「CUDA」環境が登場した。

3.2 GPU のプログラミング環境: CUDA

3.2.1 CUDA とは

CUDA (Compute Unified Device Architecture) とは、NVIDIA 社が提供する、GPU を並列データ計算デバイスとして利用するためのハードウェア・ソフトウェアアーキテクチャである。GeForce 8800 シリーズまたはそれ以降の GPU で利用可能である。

3.2.2 特徴

CUDA では C 言語の拡張という形で、グラフィックスに関係した概念を意識せずに開発を行うことができる。

CUDA を通じてプログラミングを行うとき、GPU は大量のスレッドを実行できる演算デバイスとしてみなすことができ、CPU に対するコプロセッサとして作動する。プログラムは、カーネルと呼ばれる独立して GPU で実行する宣言をした関数を作成する形となる。カーネル実行時、指定した数のスレッドがカーネルの中の命令を並列的に実行する。

4 境界要素法

境界要素法とは、解析対象を有限個の領域に分割し、各領域の境界上の節点における支配方程式を連立 1 次方程式に近似して解く数値解析法である。同じ目的に使われ、広く普及した手法である有限要素法では、計算領域内部に要素分割 (離散化) を行う必要があるが、境界要素法では、計算領域の境界を要素分割するだけでよい。また、コードの並列化が比較的容易である特徴を持つため、GPU での

* Applicability of Practical Numeric Application on GPU Implementation of Boundary Element Method

[†] Kosuke WAKI (kous-waki@idea.aoyama.ac.jp)

[‡] Graduate School of Science and Engineering, Aoyama Gakuin University

実装に向いていると考えられる。

5 開発したアプリケーション

境界要素法を用いた、非定常熱伝導解析を行うアプリケーション [3] を CUDA により開発した。計算量の多い反復処理部分を GPU 上で実行し、そのほかの部分は CPU 上で実行される。計算結果をグラフィックスとして出力することも可能で、時間ごとの温度変化のアニメーションを描画する (図 2)。

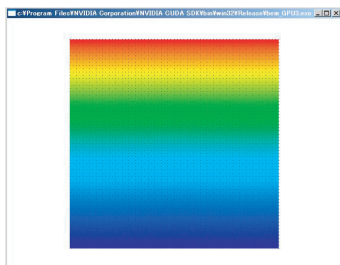


図 2 計算結果の可視化

6 実験及び結果

今回の実験に使用した GPU と CPU およびその他の環境を表 1 に示す。

表 1 今回使用した環境

CPU	Intel Core 2 Duo E6600
GPU	NVIDIA GeForce 8800 GTX
OS	Windows XP Professional
Memory	2GB

開発したアプリケーション (以下 CUDA 版) を用い、図 3 のような条件の非定常熱伝導問題の解析を行った。一定の時間刻み幅で 20 回ほど各節点の温度を計算し、それにかかる時間を測定した。現在の GPU 上の制約により、単精度での計算となる。また、比較対象として、CPU 向けに実装したもの (以下 CPU 版) で同様の問題の解析を行った (シングルスレッドでの実行)。

CUDA 版および CPU 版のアプリケーションの実行時間を 5 回測定し平均した結果と、CPU 版により倍精度で実行した解との最大誤差は表 2 のようになった。シングルスレッドによる CPU 版での実行時間に対して、CUDA 版では約 4 倍の速度向上となった。今回の実験では、理論的な演算性能ほどの速度向上は実現できなかった。これは、GPU とビデオメモリとの転送速度が原因の一つと考えられる。データの一部を CUDA で利用できる、より転送速度の速い共有メモリに置くことで、ある程度の速度向

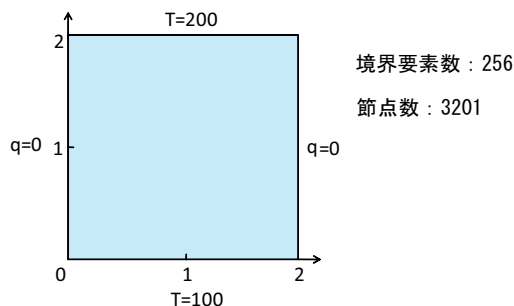


図 3 対象とした問題

上を行ったが、この共有メモリに入れられるデータのサイズはかなり小さいため、大部分のデータは通常のビデオメモリ上で利用することとなった。データの通信が少ない計算部分のみの比較では、CUDA 版による 10 倍程度の速度向上を確認できた。精度については、CPU での単精度と倍精度で実行した場合の誤差と同等であった。

表 2 平均実行時間など

	CUDA	CPU(double)
平均実行時間 (sec)	1.2×10^2	30
解の最大誤差	9.2×10^{-4}	

7 まとめ

CUDA 環境を用い、GPU を利用した境界要素法による非定常熱伝導解析アプリケーションを開発し、性能評価を行った。GPU の利用による速度向上を確認できたが、メモリの転送速度が原因で理論的な演算性能の差ほどの高速化は達成されなかった。しかし、GPU の性能は現在、CPU を大きく上回る速度で向上しており、ビデオメモリの転送速度や共有メモリの容量が改良されることで、大幅な速度向上を望むことができる。境界要素法での解析のさらなる高速化が進むことで、リアルタイムでのデータの取得から解析といった、新しい使い方が実現できるだろう。

参考文献

- [1] NVIDIA CUDA : <http://developer.nvidia.com/object/cuda.html>.
- [2] NVIDIA Corporation : CUDA Programming Guide (2007).
- [3] 黒木健実, 荒巻軍治, 添田朋子 : パソコンによる境界要素法入門, 森北出版 (1984).