

平成 25 年度 情報工学コース卒業研究報告要旨

枝廣 研究室	氏 名	橘川 雄樹
卒業研究題目	GPU を用いた大規模データ解析の高速化	
<p>コンピュータアーキテクチャの歴史において、プロセッサは目覚ましい性能向上を遂げてきた。その性能向上の主な要因は、半導体の微細化によるクロック周波数の飛躍的な増加によるものであるが、消費電力や発熱などの問題によりそのアプローチは近年限界を迎えている。クロック周波数の増加に代わる新たな性能向上の手段として、チップ内に複数の CPU コアを配置するマルチ・メニーコア技術が注目されている。複数のコアを用いて計算処理を並列化することによって性能を向上させようというものである。マルチコアにおける並列化手法として、POSIX 準拠である pthread を用いたマルチスレッドプログラミングや OpenMP によるコンパイラサポートが挙げられるが、それらの技術を用いて達成できる並列度は高々数十程度である。一方、メニーコアは、数百から数千以上の並列度を可能とする革新的な技術である。</p> <p>今日、メニーコア技術を具現化し、一般に普及している市場製品として GPU (Graphics Processing Unit) がある。GPU は当初、グラフィックス処理を目的として開発されたが、その大規模な並列計算処理能力が汎用計算にも用いられるようになった。その概念は GPGPU (General-Purpose Computing on GPUs) と呼ばれ、組み込みシステムからスーパーコンピュータまで幅広く活用されている。</p> <p>本研究では、GPGPU の応用例としてデータ解析アプリケーションに着目し、データインテンシブなワークロードに対する GPGPU 並列計算ライブラリの設計と実装を行う。特に GPU の性能を引き出す 3 種類の高速化アプローチを検討し、性能評価を行った。1 つ目は GPU での計算に必要なメモリ領域を集約的に確保するメモリ割当て手法、2 つ目は GPU が提供する読み取り専用設計されたテクスチャメモリの使用、3 つ目は CPU と GPU 間のメモリコピーを非同期化するデータ転送手法である。また、単一 GPU に対する性能評価だけでなく、複数の GPU を用いた場合のスケラビリティも評価した。</p> <p>各高速化アプローチを実装し、実行時間を比較したところ、集約的メモリ割当て手法により、メモリコピーに費やす時間を約 60% 短縮できた。また、テクスチャメモリを使用することにより、プログラム全体の実行時間を約 20% 短縮できた。一方で、非同期データ転送手法では、顕著な高速化を得ることはできなかった。この理由として、要素数の多い複数の倍精度浮動小数点型配列を同時に CPU から GPU に転送を行っても、CPU と GPU 間のデータ通信が PCI バスの帯域幅によって制限されてしまったためであると考えられる。</p> <p>GPU を複数枚に拡張した際には、GPU を 4 枚使った場合でプログラム全体の実行時間が 10% 短縮された。実行時間は短縮されたとはいえ、その改善率が 10% にとどまった理由として、データ解析のプログラムの特性上、複数の GPU で計算された結果の整合性を取るための同期処理オーバーヘッドが増加したことであると考えられる。</p> <p>本研究により、メモリ割当て手法、テクスチャメモリの使用、非同期データ転送、マルチ GPU 化は、総じてデータ解析アプリケーションにおける GPGPU 計算性能を向上させる有効手段であることが示されたが、その効力には大きな違いがみられた。今後の課題としては、GPU の枚数に応じた GPGPU 計算性能のスケールアップを実現するための複数 GPU 間データ同期手法の考案等が挙げられる。</p>		