

eigen_sg: マルチコア+GPGPU 環境における固有値ソルバ開発

今村 俊幸^{†,††} 山田 進^{†,††} 町田 昌彦^{†,††}

Householder 三重対角化を利用する伝統的な計算手法により、(CPU 単体に対して) 広帯域なメモリバンド幅を有する GPU の利点を生かす形で計算割り当てを実施する eigen_sg の開発を行っている。本ポスター発表では、特定部分を CUBLAS レベルで GPU 実装する方法で CPU 単体に対して 2 から 3 倍 程度の速度向上を達成したので、その実装方法や性能分析を行いたい。

キーワード: GPGPU, 固有値ソルバ, マルチコアプロセッサ.

1. GPU 版固有値ソルバ

「天河」や「TSUBAME2」に代表されるマルチコア-GPU 複合型クラスタシステムがスパコンのトレンドとして台頭してきている。これらシステム上で機能する高性能な固有値ソルバの開発は避けて通れない (LAPACK の GPU 移植や MAGMA プロジェクトなどが存在)。GPGPU に限ることなく深いメモリ階層かつ複合計算リソースを混載するシステムに対する固有値ソルバ開発の方向性は今後大きな意味がある。

2. BLAS/CUDA, DGEMV の性能

図 1, 2 は NVIDIA GTX285 並びに Tesla C2050 での DGEMV の性能評価である (デバイス上でのカーネル関数の動作時間のみ測定, 赤: CUBLAS3, 緑: MAGMABLAS 0.2, 青: 今回開発版)。ホスト計算機の性能は 2~3GFLOPS が上限であることから、GPU の高いポテンシャルは容易に想像できる。また、行列を転置させたものは性能が悪く、DGEMM の高性能さと比較すると総じて性能面と安定性に改良の余地がある。

3. eigen_sg 総合評価

表 1 は DGEMM と DSYMV (DGEMV と DSYMV を組み合わせて実装) を GPU 化した固有値ソルバ eigen_sg の開発版の測定結果である。全固有対の計算時間を示している。各種 BLAS の組み合わせで性能が変化するため、適切な BLAS の静的または動的な切り替えが必要だが、3 倍近い性能向上が確認できる。(自動チューニング技術の重要性!!)。

表 1 eigen_sg 計算時間 (秒), CPU: Intel Core i7 860 (2.8GHz), GPU: NVIDIA Tesla C2050

次元	CPU4cores	CUBLAS	Tuned
1000	.243	.286	.298
2000	1.813	1.397	1.178
3000	5.857	3.944	2.997
4000	13.67	8.153	5.841
5000	25.04	14.78	10.66
6000	42.32	24.27	17.02
8000	101.2	54.23	35.49

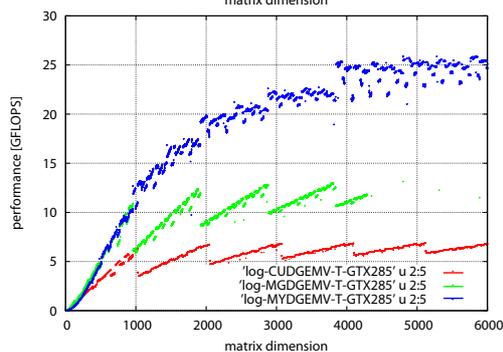
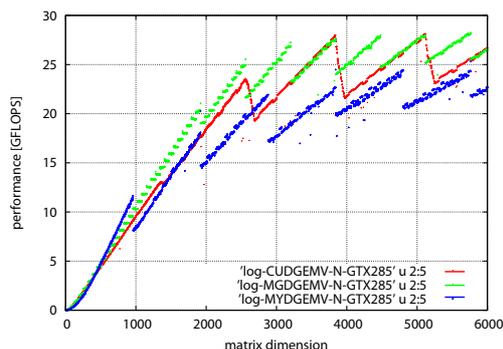


図 1 DGEMV の性能評価 (GFLOPS), GTX285 (行列非転置, 行列転置)

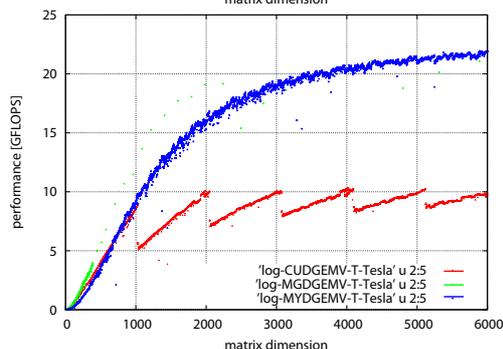
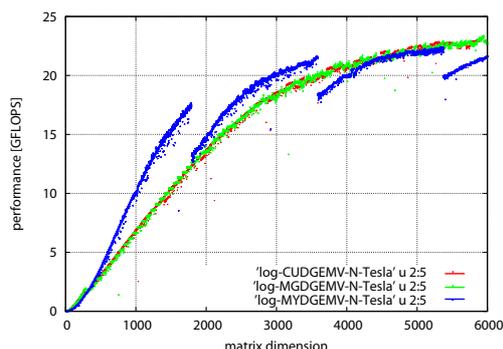


図 2 DGEMV の性能評価続き (GFLOPS), TeslaC2050 (行列非転置, 行列転置)

[†] 電気通信大学情報理工学研究所
^{††} 日本原子力研究開発機構システム計算科学センター
^{†††} 戦略的創造研究推進事業 CREST (科学技術振興機構 JST)