

PEZY-SC2 上における倍々精度 Rgemm の実装と評価

菱沼 利彰¹, 中田 真秀²

¹ 株式会社 PEZY Computing, ² 理化学研究所

e-mail : hishinuma@pezy.co.jp

1 はじめに

計算科学において、数値演算に用いられる倍精度浮動小数点数は演算精度が原因で解が得られなかったり計算時間が増大することがある。これまで様々な高精度演算手法を用いて誤差の影響を低減し、応用が行われてきた [1]。今後、計算機の進化に伴う解くべき問題の大規模化、および小規模でも数値的に難しい問題を解くため、高精度演算の需要は高まると考えられる。

比較的低コストかつ高速に高精度演算が実現可能な、倍々精度演算を用いる手法がある。倍精度浮動小数点数を2つをベクトルのように用いてほぼ四倍精度を実現できる [2]。2つの倍精度浮動小数点数について加算および乗算を行うと一般に誤差が入る浮動小数点数同士の加算、乗算および、誤差は倍精度演算だけで厳密に評価でき、これを利用して倍精度演算 20 から 40 回程度で処理できるため古くから広く用いられている [3]。また、現在でもライブラリの開発 [2] や、アクセラレータ等による高速化 [4, 3] が行われている。

我々は、MIMD 型メニーコアプロセッサ PEZY-SC2 向けに高精度演算ライブラリ QD[2] の倍々精度演算機能を移植し、最適化を行い、倍々精度行列-行列積の性能評価を行い、倍々精度演算のピーク性能の 74 % 程度の結果を得た。

2 倍々精度ライブラリ pzqd

2.1 PEZY-SC2 のアーキテクチャ

PEZY-SC2 は MIMD アーキテクチャを採用しており、2048 個のコアが自身のプログラムカウンタをもつことで各コア 8 スレッドを別々に動作させることが可能である。GPU などと比べて条件分岐などによる性能劣化が起こりにくい。また、加算と乗算を 1 命令で行うことができる MAD 命令が使用可能である。

各コアにはスクラッチパッドメモリが搭載されており、スレッドごとのスタック領域が確保され、余剰領域をユーザがコア内共有のローカルメモリ空間として使用可能である。

2.2 倍々精度ライブラリ pzqd の実装

PEZY-SC2 上で動作するプログラムの開発には、OpenCL ライクなプログラミング環境である PZCL を使用する。PZCL ではデバイスで動作するカーネルプログラムと、データ転送やカーネル呼び出しなどの制御を行うホスト CPU 用のホストプログラムを分けて作成する。

我々は開発中の SDK の C++ の構文を利用し、QD のヘッダで定義された倍々精度の数学関数・算術演算の演算子などをカーネルプログラムで利用できる pzqd ライブラリを開発した。

これにより、ホストプログラムから転送された QD ライブラリの倍々精度型を、カーネルプログラムから扱うことができる。

3 数値実験

3.1 実験環境

我々は実験にあたり Intel Xeon D-1571 をホスト CPU とした 1984 コアモデルの PEZY-SC2 プロセッサを使用した。PEZY-SC2 の倍精度ピーク性能は $0.7 \text{ [GHz]} \times 1984 \text{ [cores]} \times 2 \text{ (MAD 演算)} \doteq 2777 \text{ GFLOPS}$ で、DDR4 2400 MHz, 64 GB メモリ (メモリ帯域 76 GB/s) と、40 MB の LLC メモリ、コアあたり 20 KB のスクラッチパッドメモリを搭載している。

OS は CentOS 7.2, ホストプログラムのコンパイラは gcc 4.8.5, カーネルプログラムのコンパイラは pzSDK4.0 + LLVM 3.6.2 を用いた。

比較対象として、Intel Xeon E5-2618L@2.3 GHz, 8core, 64GB を用いた。

3.2 PEZY-SC2 における Rgemm

倍々精度の Rgemm の核となる倍々精度積和演算は倍精度加算 35 回、乗算 9 回から構成される。PEZY-SC2 のピーク性能は MAD 命令で計算するが倍々精度演算は加算と乗算の回数が不均一なため、 $(35 + 9)/2 = 22$ サイクルでは処理できず、全ての乗算に対して MAD 命令を利用したとしても 35 サイクルかかる。従っ

て倍々精度の積和演算は $2777/35 \times 22 \approx 1745$ GFLOPS をピーク性能とした。

また、比較対象の CPU は AVX2 を用いない場合、理論ピーク性能は 73.6 GFLOPS で、上述の倍々精度演算の加算、乗算の不均一さを考慮すると、46 GFLOPS が倍々精度演算のピーク性能となる。

行列-行列演算の高速化のために 2×2 のブロック化を行い、ブロック化した小行列と倍々精度演算に必要な中間変数をすべてローカルメモリ空間に確保するようにした。

このとき、行列サイズが大きくなると必要なデータサイズが増大し、ローカルメモリ空間のサイズを超えてしまうケースがあったため、SDK にスレッド数を減らす機能を実装しスタック領域を減らすことで、1 スレッドが使えるローカルメモリ空間のサイズを増やせるようにした。

3.3 実験結果

図 1 に倍々精度 Rgemm の性能を示す。ホスト CPU と PEZY-SC2 間のデータ転送時間は含めていない。ローカルメモリを使用しない場合の結果を “no localmem” に示した。また、実験に用いた行列は乱数で生成された正方密行列である。縦軸は性能で演算量は $44 \times N^3$ として求めた。横軸は行列サイズである。

このとき、高速化のために倍々精度の加算、乗算をすべてインライン展開しているが、スタックサイズが増大するため 8 スレッドではローカルメモリ空間にデータが置けず、行列サイズ 1900 までしか計算できない。そのため、pzdq ライブラリでは行列サイズに合わせてスレッド数を変えるようにした。

また、CPU ではブロック化、ループアンローリングを施したコードを OpenMP で 16 スレッド並列したものをを用いた。

実験の結果、スレッド数、ローカルメモリ空間の使用の有無にかかわらず CPU より高速で、ローカルメモリ空間を使わない場合は性能は 150GFLOPS 程度だが、ローカルメモリを用いることで、8 スレッドで最大で 1291 GFLOPS で、ピーク性能の 74%、倍々精度換算で約 53 GFLOPS となった。行列サイズ 600 でも 894 GFLOPS 出ており、小さいサイズの問題でも性能が高い。4 スレッドにすることで行列サイズに関わらず計算できるようになるが、8 スレッドと比べて 10% から 20% 遅い。

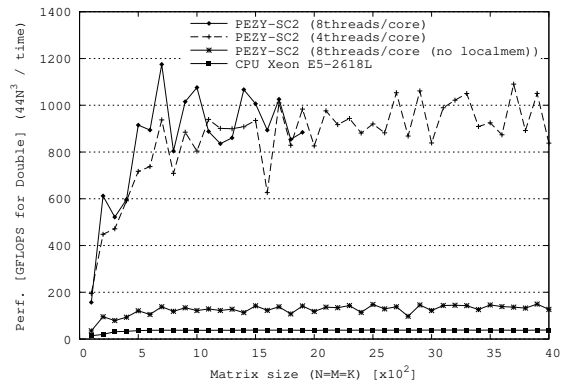


図 1. 倍々精度 Rgemm の性能 (カーネルのみ)

4 結論

我々は MIMD 型メニーコアプロセッサ “PEZY-SC2” 上に高精度演算ライブラリ QD の倍々精度演算機能を移植し、それを用いて倍々精度 Rgemm を実装し、性能評価を行った。

スレッド数を減らしてローカルメモリ空間に倍々精度演算に必要な中間変数や行列のブロック化をした小行列などを入れることで、どの行列サイズでも計算できるようになり、性能は倍々精度換算で最大 58GFlops、倍精度換算で 1291 GFlops、ピーク性能の 74% 程度の結果が得られた。

今後の課題として、スタックサイズの削減や、8 スレッドでも計算できるサイズを大きくしていくことが考えられる。

参考文献

- [1] D. H. Bailey, High-Precision Floating-Point Arithmetic in Scientific Computation, computing in Science and Engineering, pp.54-61, 2005.
- [2] Y. Hida, QD library, <http://crd-legacy.lbl.gov/~dhbailey/mpdist/> and reference therein.
- [3] M. Nakata, *et al.*, “A fast implementation of matrix-matrix product in double-double precision on NVIDIA C2050 and its application to semidenite programming”, ICNC2012, pp.68-75, 2012.
- [4] D. Mukunoki, *et al.*, “Implementation and Evaluation of Quadruple Precision BLAS Functions on GPUs”, PARA2010, pp.249-259, 2010.