GPU クラスタにおける並列数論変換の自動チューニング

Automatic Tuning for Parallel Number-Theoretic Transforms on GPU Clusters

高橋 大介 (Daisuke Takahashi)[1]
[1] 筑波大学計算科学研究センター (Center for Computational Sciences, University of Tsukuba)
e-mail：daisuke@cs.tsukuba.ac.jp

1 はじめに

数論変換（number-theoretic transform，以下 NTT）は，離散 Fourier 変換を有限体に一般化したものであり，準同型暗号，多項式乗算，多倍長精度乗算などに広く用いられている．GPU クラスタにおいてチューニングを行う際に，最適な性能パラメータは GPU のアーキテクチャ，ノード間を結合するネットワーク，そして問題サイズなどに依存するため，これらのパラメータをその都度手動でチューニングすることは困難になりつつある．本論文では，GPU クラスタにおいて自動チューニングを並列 NTT に適用し性能評価を行った結果について述べる．

2 GPU クラスタにおける並列数論変換の自動チューニング

n 点 NTT は $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$（$p$ は素数）において以下のように表すことができる．

$$y(k) = \sum_{j=0}^{n-1} x(j)\omega_n^{jk} \bmod p, \quad 0 \le k \le n-1 \tag{1}$$

ここで，ω_n は 1 の原始 n 乗根である．式 (1) は高速 Fourier 変換（fast Fourier transform，以下 FFT）と同様のアルゴリズムを適用することで，演算回数を $O(n \log n)$ に削減することができる．

GPU クラスタにおける並列 NTT の実装が提案されている [1]．この実装は four-step NTT アルゴリズムに基づいており，MPI と OpenACC を用いて four-step NTT の並列化を行っている．並列 NTT を行う際には，全対全通信が 3 回行われることから，計算時間の大部分が全対全通信によって占められることになる．

GPU クラスタにおいて並列 NTT をチューニングする際には，全体に関わる性能パラメータとして主に以下の 3 つが存在する．

(1) 全対全通信方式
(2) 通信メッセージサイズの分割数
(3) 基底

(1) では 2 段階全対全通信アルゴリズム [2] において，MPI プロセスグリッドの最適な形状を探索する．(2) では演算と通信をオーバーラップする際に，通信メッセージサイズの最適な分割数 NDIV を探索する．NTT のデータサイズが $N = N_1 N_2$ と分解できる場合，P 個の MPI プロセスを持つ GPU クラスタにおける four-step NTT では，N_1 と N_2 の値は $N = N_1 N_2$ および $N_1, N_2 \ge P$ を満たしていればよい．そこで，(3) では最適な N_1 と N_2 の値について探索を行う．上記の性能パラメータのうち，(2) と (3) について自動チューニングを適用した．

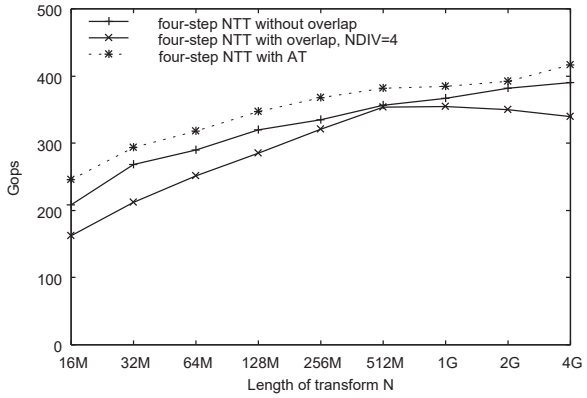

図 1. 並列 NTT の性能（Pegasus，16 ノード）

3　性能評価

性能評価にあたっては，four-step NTT の GPU 実装と，第 2 章で述べた自動チューニング手法を four-step NTT の GPU 実装に適用したものとの性能比較を行った．測定に際しては，順方向 NTT を連続 10 回実行し，その平均の経過時間を測定した．GPU クラスタとして，筑波大学計算科学研究センターに設置されている Pegasus（150 ノード）のうち 16 ノードを用いた．コンパイラは NVIDIA HPC Compilers 24.5 を用い，コンパイルオプションは`-fast -acc=gpu -gpu=cc90,pinned` を指定した．MPI ライブラリは OpenMPI 4.1.8 を用いた．各ノードあたりの MPI プロセス数は 1 に設定した．N 点 NTT の Giga-operations per second（Gops）値は $(3/2)N \log_2 N$ より算出している．

並列 NTT の性能を図 1 に示す．図 1 から，並列 NTT に自動チューニングを適用した場合（four-step NTT with AT）が最も性能が高いことが分かる．自動チューニングの結果，演算と通信をオーバーラップしない場合の方が演算と通信をオーバーラップする場合よりも性能が高くなっているため，自動チューニングによる性能向上は最適な基底を選択したことにより得られている．また，演算と通信をオーバーラップする場合（four-step NTT with overlap, NDIV=4）では通信メッセージサイズが常に 4 分割されており全対全通信性能が低くなっていることから，演算と通信をオーバーラップしない場合（four-step NTT without overlap）よりも性能が低くなっていることが分かる．

4　まとめ

本論文では，GPU クラスタにおいて並列 NTT の自動チューニングを実現し性能評価を行った結果について述べた．MPI と OpenACC を用いて four-step NTT を並列化した．並列 NTT において自動チューニングを行うことで，性能をさらに向上させることができることを示した．

謝辞　本研究成果は筑波大学計算科学研究センターの学際共同利用プログラム（Pegasus）を利用して得られたものである．本研究は，JSPS 科研費 25K15134 の支援によって行われた．

参考文献

[1] D. Takahashi, “Parallel implementation of number-theoretic transform on GPU clusters,” in *Proc. 24th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP 2024), Part III*, ser. LNCS, vol. 15253. Springer, pp. 204–218, 2025.

[2] D. Takahashi, “Automatic Tuning for Parallel FFTs,” in *Software Automatic Tuning: From Concepts to State-of-the-Art Results*, Springer, pp. 49–67, 2010.

混合精度版 ILU(0) 前処理付き BiCGSTAB 法における低精度方式実現方法による影響の分析

Analysis of the effect of low precision realization methods on the Mixed Precision ILU(0) Preconditioned BiCGSTAB Method

久木田 仁 (Kukita Jin)[1], 藤井 昭宏 (Fujii Akihiro)[1], 田中 輝雄 (Tanaka Teruo)[1]
[1] 工学院大学
e-mail : em24017@ns.kogakuin.ac.jp

1 概要

コンピュータシミュレーションにおいて，大規模で疎な係数行列をもつ連立一次方程式の求解は重要である．一方で，科学技術計算に単精度以下の低精度演算を活用する混合精度演算の試みが盛んに行われている．本研究では，メモリアクセス性能の向上等の高速化を目標として，ILU(0) 前処理付き BiCGSTAB 法に対して低精度化を施した影響を分析する．我々はこれまでテスト行列の条件数を変化させた上で収束性の分析を行ってきた [2]．その中で，行列の非対称性の変化について議論の余地があった．本発表では，行列の非対称性に対する収束性の挙動や，非対称性の異なる行列に対する低精度化の影響を分析する．

2 混合精度版 ILU(0) 前処理付き BiCGSTAB 法とデータの低精度化

ILU(0) 前処理付き BiCGSTAB 法には，BiCGSTAB 法に低精度演算を取り入れる際に解の精度が低下する問題を解決する MP-IR(Mixed Precision Iterative Refinement) 方式がある [1]．MP-IR 方式は，反復改良法の誤差算出に低精度の ILU(0) 前処理付き BiCGSTAB 法を用いており，反復法の中に反復法が入る二重構造をとる．

本研究では，ビット演算により対象となる倍精度データの仮数部の下位 0〜52 ビットを 0 にしてデータの精度を変化させる実装を行なっている．なお，仮数部ビット数が 52bit の場合は通常の倍精度と同一の挙動となり，0bit の場合は 2 の冪乗数のみを表現する．

3 行列の条件数による影響の分析

本研究では，二次元定常移流拡散方程式を離散化して得られる疎な五重対角行列をテスト行列として用いる．これは図 1 に示される 5 点差分法により得られ，初期パラメータは以下となっている．行列サイズ $n = 3600$，離散化幅$\Delta x = \Delta y = \frac{1}{60}$，移流項 $v = 1$，拡散項 k は二次元平面内の座標 (i, j) に依存して 1 または 10 の値をとる．これまで，我々は拡散項 k を設定して任意の条件数の行列に対して実験を行った．その主な結果は 2 つある．第一に，ILU(0) 前処理に対しては悪条件で仮数部 52bit から 11bit への低精度化を行なっても収束性の悪化は軽微だった．第二に，BiCGSTAB 法に対しては低精度化が進むほど収束性が悪化し，17bit までの低精度化において求解できた．

4 行列の非対称性による影響の分析

以上の結果を踏まえ，移流項 v により行列の非対称性を変化させた影響を分析する．まず，通常の倍精度において移流項 v を変化させた影響を分析する．反復回数の制限は行列サイズの 3 倍の 10800 とし，これを超えた場合は打ち切られ，解は求まらなかったとして扱う．まず，通常の倍精度で移流項 v を 1 から 200 の範囲で変化させた結果のグラフを図 2 に示す．縦軸が反復回数，横軸が移流項

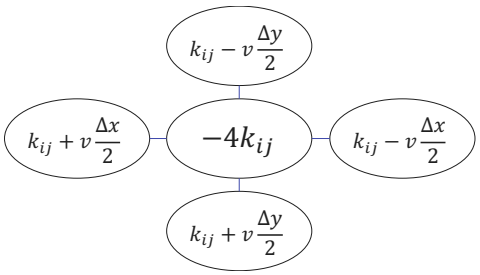

図 1. 対象問題の 5 点ステンシル

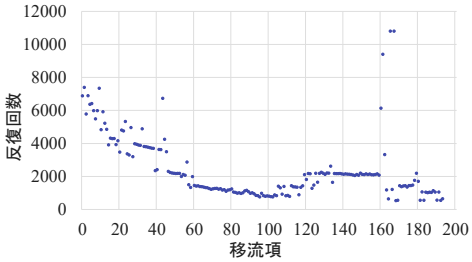

図 2. 移流項に対する反復回数

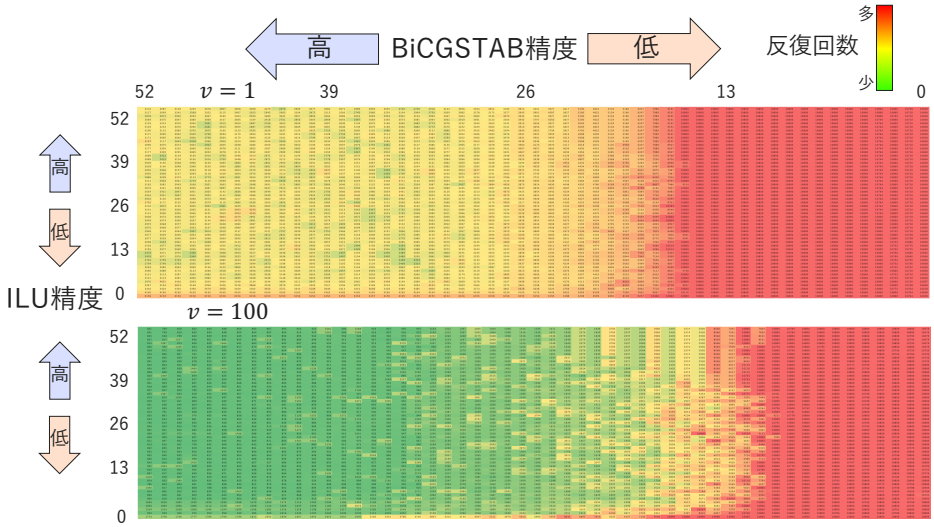

図 3. 移流項による低精度化の影響の違い

の値である．この結果から，移流項 v の増加に伴い，反復回数が減少していく挙動が見られる．なお，$v > 200$ とした場合は求解不可となった．

次に，異なる移流項における低精度化の影響の違いを分析する．求解可能な範囲で収束までの反復回数に差異が見られた $v = 1$, 100 において，ILU(0) 前処理と BiCGSTAB 法の双方を低精度化した結果のグラフが図 3 である．上部が $v = 1$，下部が $v = 100$ である．横軸は BiCGSTAB 法の精度，縦軸は ILU(0) 前処理の精度を表す．左上端が通常の倍精度であり，右下ほど精度が低くなっている．色は赤いほど収束にかかった反復回数が多かったことを表しており，最も濃い赤は反復上限に達して求解不可であったことを意味する．なお，上下ともに共通のスケールを用いている．ILU(0) 前処理に対する低精度化ではともに仮数部ビット数が 0 でも求解可能であり，拡散項 k を変動させた実験に続き ILU(0) 前処理に対する低精度化の有効性が示唆された．BiCGSTAB 法への低精度化では，$v = 100$ において 13bit までの低精度化を行なっても求解できた．

謝辞　本研究の一部は JSPS 科研費 JP23K11126 の助成により実施しました．

参考文献

[1] Yingqi Zhao, Takeshi Fukaya, Takeshi Iwashita, Numerical Behavior of Mixed Precision Iterative Refinement Using the BiCGSTAB method, Journal of Information Processing, Vol.31 860-874, 2023.

[2] 久木田 仁, 藤井昭宏, 田中輝雄, ILU(0) 前処理付き BiCGSTAB 法における低精度演算導入の分析, 計算工学講演会論文集 Vol.30 (2025 年 6 月).

スパコン Miyabi と GPU アプリケーション開発

Miyabi Supercomputer and Development of GPU applications

下川辺 隆史 (Takashi Shimokawabe)[1]
[1] 東京大学 (The University of Tokyo)
e-mail : shimokawabe@cc.u-tokyo.ac.jp

1 概要

近年、電力と設置面積の制約のもと、最大限に計算性能を高くするために、多くのスパコンで GPU などの演算加速装置が導入されている。東京大学情報基盤センターと筑波大学計算科学研究センターと共同で設置する最先端共同 HPC 基盤施設（JCAHPC）では 2025 年 1 月から GPU を搭載したスパコン Miyabi の運用を開始した。本講演では、流体計算などの GPU アプリケーションの開発事例と、Miyabi の概要や GPU 移行に向けた取り組みについて紹介する。

2 Miyabi スーパーコンピュータ

東京大学情報基盤センターと筑波大学計算科学研究センターと共同で設置する最先端共同 HPC 基盤施設（JCAHPC）が運用するスーパーコンピュータ Miyabi は、学術目的で「富岳」に次ぐ国内第 2 位の性能を持つ。本システムは、「AI for Science」という新たな研究パラダイムを強力に推進することを目的として設計されており、特に科学研究への貢献を目指す。

システムの計算能力の中核を成し、総理論演算性能 80.1 PFLOPS のうち 78.8 PFLOPS を占めるのが、1,120 ノードから構成される GPU 演算加速ノード群「Miyabi-G」である 。その GPU には、NVIDIA 社が開発した GH200 Grace-Hopper Superchip が搭載されており、スーパーコンピュータに採用されるのは、これが国内初の事例である。GH200 では、CPU と GPU を 900 GB/s という極めて広帯域なチップ間インターコネクト NVLink-C2C で直接結合している。従来の PCIe バス接続に内在していたデータ転送のボトルネックを根本的に解消し、ハードウェアレベルでキャッシュコヒーレンシを保った単一の共有アドレス空間を構築する。これにより、プログラマは明示的なデータ転送命令を意識することなく、GPU 計算において、開発効率と実行性能を飛躍的に向上させることが可能となる [1]。

一方で Miyabi は、190 ノードから成る汎用 CPU ノード群「Miyabi-C」も併設している。Intel Xeon Max プロセッサを搭載したこのノード群では、引き続き CPU が必要なユーザーのニーズに応えている。

3 GPU 移行に向けた取り組み

スーパーコンピュータ Miyabi の導入は、3,000 人を超える広範な利用者コミュニティを、長年慣れ親しんだ CPU 中心のプログラミングから、GPU を活用する新たなパラダイムへと導く大きな挑戦である 。JCAHPC は、この大規模な移行には 18 ヶ月から 30 ヶ月を要すると考え、ハードウェア導入と並行して「GPU 移行プロジェクト」を立ち上げ、能動的かつ多層的なユーザー支援エコシステムを構築した。

この支援エコシステムの一つが、課題持ち込み型の「GPU ミニキャンプ」である。これは、参加者が自身の研究で実際に使用しているコードを持ち寄り、数日間にわたって GPU への移植や性能最適

化に集中的に取り組む、実践的なハンズオン形式の講習会である。ミニキャンプでは、JCAHPC の教員だけでなく、国内大学の基盤センターの教員や NVIDIA 社などの研究者・開発者が、メンターとして参加し、参加者一人ひとりの固有の課題解決を支援する。さらに、より手軽に利用できる支援策として、個別の技術的課題に専門家が応えるオンライン「GPU 相談会」も定期的に開催している。性能が向上しない原因の特定や適切なライブラリの選定など、マニュアルを読むだけでは解決が難しい具体的な問題に対応し、移行支援を行っている。

既存コードの GPU 化手法としては、指示文を用いる OpenACC や OpenMP、標準言語の並列機能の活用などがあり、これらを流体計算などで評価 [2] するとととともに、高性能計算ソフトウェア財団が推進する C++ パフォーマンスポータビリティライブラリ Kokkos の利用などの検討を進めている。

本講演では、流体計算などの GPU アプリケーションの開発事例を紹介しながら、Miyabi の概要や GPU 移行に向けた取り組みについて紹介する。

参考文献

[1] T. Hanawa, K. Nakajima, Y. Miki, T. Shimokawabe, K. Yamazaki, S. Sumimoto, O. Tatebe, T. Boku, D. Takahashi, A. Nukada, N. Fujita, R. Kobayashi, H. Tadano and A. Naruse, ”Preliminary Performance Evaluation of Grace-Hopper GH200”, 2024 IEEE International Conference on Cluster Computing Workshops (CLUSTER Workshops), pp. 184 - 185, 2024.

[2] Z. Yuan and T. Shimokawabe, ”Accelerating LBM with C++ STL Asynchronous Parallel Model”, 25th International Conference on Computational Science (ICCS 2025), Singapore, 7-9 July 2025

