

HPC システムにおける Approximate Computing のための 非同期な演算精度変更

Asynchronous Precision Adjustment
for Approximate Computing on HPC Systems

和田 康孝
明治学院大学

1. 研究目的

電力の制約により駆動・利用可能な計算資源の量に強く制限がかかるようになるなど、様々な要因・制約から、特に HPC システムにおいては、従来の限界を超えた性能向上が求められる。この問題に対する一つの解決策として、演算結果にある程度の誤りが含まれることを許容し、その誤差と引き換えに実行性能や電力効率を向上させる Approximate Computing (以下 AC)、アプリケーションの実行状況や特性に応じ演算精度を動的に調整する Transprecision Computing (以下 TC)、といった技術が注目を集めている。一方、AC や TC を積極的に適用すると、最終的に得られる出力や結果に大きな誤差が発生し、演算が収束しなくなってしまうことが考えられる。そのような状況を考慮しつつ AC や TC の適用タイミングを最適化するようにアプリケーションを書き換えることには大きな負担・コストを必要とする。

本研究課題では、2024 年度からの継続課題として、アプリケーション内で動的に演算精度を切り替える際、そのためのデータ型変換やコピーを非同期に行うことで、そのオーバーヘッドを削減あるいは隠蔽することを目的として、そのための仕組みやライブラリ実装等について検討を行い、一部機能の自動化実現を目指す。

2. 研究成果の内容

上述の目的・目標に向けた研究開発を実施するにあたり、科研費研究課題「多要素協調型 Approximate Computing 実現に向けた HPC アプリケーション解析手法」における取り組みと連携し、動的に AC を適用した際の効果の評価・検証を主に実施した。

動的に AC を適用する際には、同じランク・スレッド内で扱われるデータの精度を変更することと、ランク間で通信を行う際に、送信元・送信先の状況に応じてデータの精度を変更すること、の 2 点が少なくとも必要となる。前者は、処理の上では単純なキャスト処理の繰り返しとして表現可能ではあるが、キャスト前後の値を同時に保持する必要があり、例えば FP64 (倍精度) と FP32 (単精度) の間での切り替えを考えると、FP64 のみの場合と比較して、概ね 1.5 倍のメモリ容量を必要とする。さらに後者においては、通信ライブラリ内に変換用のバッファを用意する必要があるため、アプリケーションの実装からは見えない部分でのオーバーヘッドも考慮しなければならない。

本報告執筆時点ではライブラリとしての実装には手が届いていないが、上記のような単純な処理をそのままライブラリとして実装するだけでは、メモリ量・演算両面のオーバーヘッドの削減・隠蔽が難しくなってしまう。演算精度変更のタイミングを緩やかにし、前後の処理とオーバーラップさせるなどの工夫が必要である。

3. 学際共同利用プログラムが果たした役割と意義

学際共同利用プログラムにおける取り組みを通じて、Pegasus, Miyabi を利用する機会を提供いただき、様々な特徴を持ったシステムを対象に評価・開発を行うことができた。特に、本研究課題における成果の一部を活用・発展させる形で検討・応募した、科研費・基盤研究 (C) 研究課題「Approximate Computing を用いたノイズあり量子回路シミュレーションの高速化」(2026年度~2028年度) が新たに採択となった。計算機環境の利活用に加え、研究者間の連携、さらには科研費研究課題の採択など、本学際共同利用プログラムの果たした役割は大きい。

4. 今後の展望

HPC システムにおける性能向上は、ハードウェアからアプリケーションに至るまで全てのレイヤが関係しており、様々な技術の連携が必要となる。本研究課題が対象とする Approximate Computing 技術においては特にそれが顕著であり、アプリケーションの実効性能のみならず、演算結果の正当性にも影響を及ぼす。現在は、単に動的に演算精度を変更するための仕組みをライブラリ化・API 化し、適用のためのコストを削減することを目指して取り組みを実施しているが、将来的には、実行結果の正しさをどのように保ち、補償していくかといった観点でも取り組みを拡張していく必要がある。

5. 成果発表

(1) 学術論文

- [1] Yasutaka Wada, Yoshiyuki Morie, Ryohei Kobayashi, Ryuichi Sakamoto, “Enabling Dynamic Approximate Computing for HPC Applications”, Journal of Information Processing, 2025, Volume 33, Pages 668-674, <https://doi.org/10.2197/ipsjjip.33.668>

(2) 学会発表

- [2] Yasutaka Wada, Yoshiyuki Morie, Ryohei Kobayashi, Ryuichi Sakamoto, “Enabling Rank- and Iteration-Level Approximate Computing on HPC Applications”, SCA/HPCAsia 2026 Poster Session, Jan., 2026.
- [3] 和田 康孝, 小林 諒平, 森江 義之, 坂本 龍一, “イタレーションレベル Approximate Computing 手法の提案と予備評価”, 情報処理学会第 199 回ハイパフォーマンスコンピューティング研究会, May, 2025.

(3) その他

- [1] Yasutaka Wada, Yoshiyuki Morie, Ryohei Kobayashi, Ryuichi Sakamoto,
 “Dynamic Approximate Computing Methods for HPC Applications” ,
 Exhibition, The International Conference for High Performance
 Computing, Networking, Storage, and Analysis (SC25), Nov., 2025.

使用計算機	使用計算機に○	配分リソース*		
		当初配分	移行*	一般利用による追加
Pegasus	○	3,200		0
Miyabi-G	○	10,800	0	0
Miyabi-C	○	1,260	0	0
※配分リソースについてはノード時間積をご記入ください。 *バジェット移行を行った場合、「+2000」「-1000」のように記入				