NVIDIA Tensor コアを用いた テンソルネットワーク型量子回路シミュレーション

大友広幸（東京工業大学 情報理工学院）

第 18 回 HPC-Phys 勉強会 （2023/2/8）

自己紹介

- 名前：大友広幸
- 所属：東京工業大学 横田理央研究室 D3
- 趣味：ピザ窯作り、GPU 熱燗・ローストビーフ

興味

おもしろプロセッサ

- Preferred Networks、R-CCS、NVIDIA、PEZY、Fixstars などでそれっぽいインターン。

研究

混合精度演算。特に行列積。乱択数値線形代数と量子回路シミュレーションも。

今日の話

テーマ

深層学習用ハードウェアはそれ以外の用途に使えるか？

- 深層学習により計算機が発展
  - 低精度・混合精度
  - 行列積専用回路
- HPC アプリに応用可能？
  - 単純には難しい。

Myth 11: HPC Will Pivot to Low or Mixed Precision!

A high-performance language is nothing without proper data types, but high-precision operations such as fp64 come at a significant cost in terms of silicon area, energy and speed, according to Myth 6. Lowering this precision can save costs

出典: S. Matsuoka, et al. (2022)
Myths and Legends in High-Performance Computing

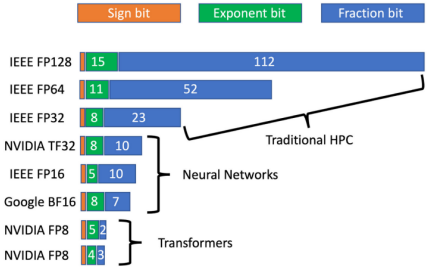
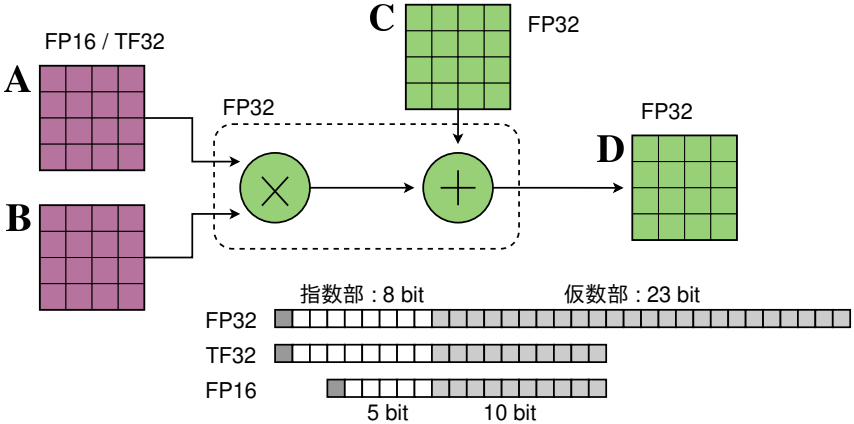


FIGURE 1. Overview of FP representations.

出典: H. Ltaief, et al., (2022)
Responsibly Reckless Matrix Algorithms for HPC Scientific Applications

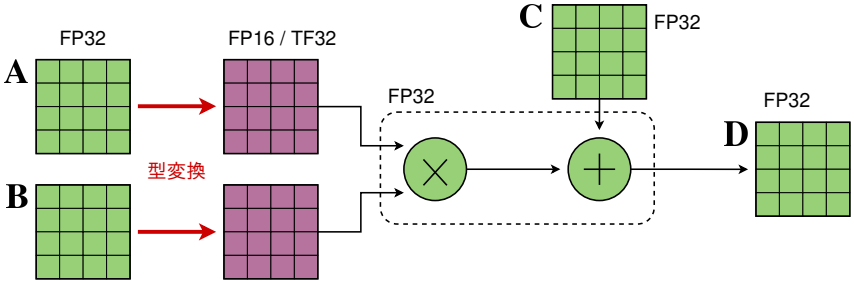
NVIDIA Tensor コア

- 混合精度行列積和回路（$\mathbf{A} \cdot \mathbf{B} + \mathbf{C}$）
- 最近の NVIDIA GPU にはどれでも搭載されている。
- A100 GPU で 312 TFlop/s、H100 GPU で 989.5 TFlop/s。
  - 単精度性能はそれぞれ 19.5 TFlop/s、67 TFlop/s。**15 倍ほどの性能差。**



NVIDIA Tensor コアで単精度行列積

- Tensor コアで**単精度**行列積和を計算したい
- 入力行列 $\mathbf{A}, \mathbf{B}$ の型変換が必要

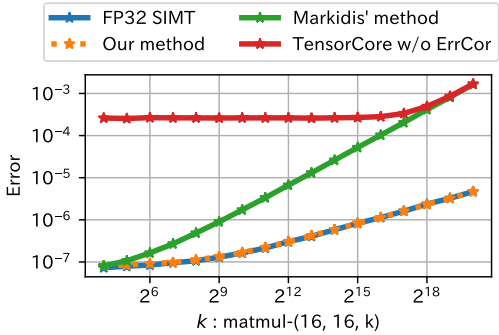


- 型変換の影響で計算精度が**半精度程度まで劣化**

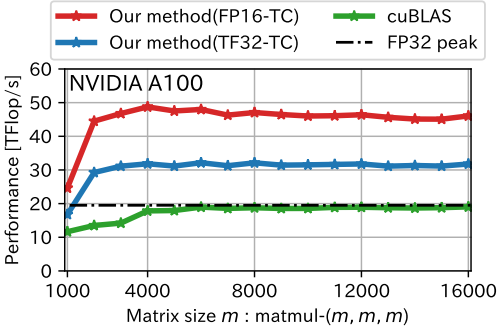
単精度行列積エミュレーション

単精度行列積を **FP32 の理論ピーク性能よりも速く**計算。
（Ootomo and Yokota, 2022）。

計算精度



計算性能



論文：Recovering single precision accuracy from Tensor Cores while surpassing the FP32 theoretical peak performance, IJHPCA, 2022

単精度行列積エミュレーション

既存研究（Markidis, et al. 2018）

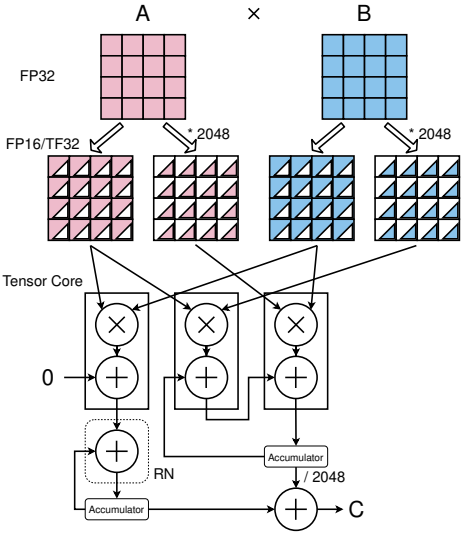
- 型変換により失われる仮数部を計算
- これを用いて行列積の精度を補正

⇒ が、エミュレートできていない。

我々の研究

- Tensor コア内部の「丸め」が悪い
- それをソフトウェア的に回避

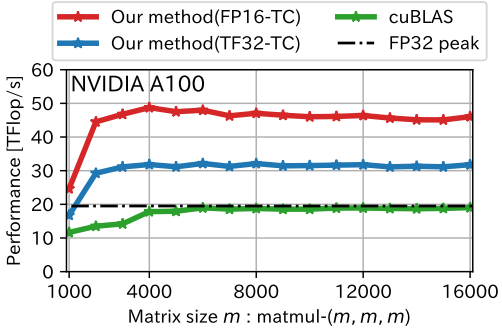
⇒ 単精度をエミュレート



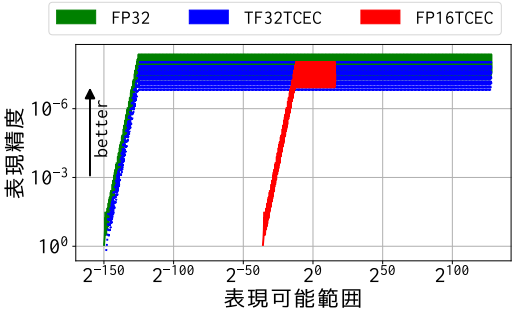
単精度行列積エミュレーション

- TF32TCEC: **完全** SGEMM エミュレーション
  - TF32 入力の Tensor コアを使用。最大 33 TFlop/s 実効。
- FP16TCEC: **限定指数部範囲** SGEMM エミュレーション
  - FP16 入力の Tensor コアを使用。最大 51 TFlop/s 実効。

計算精度



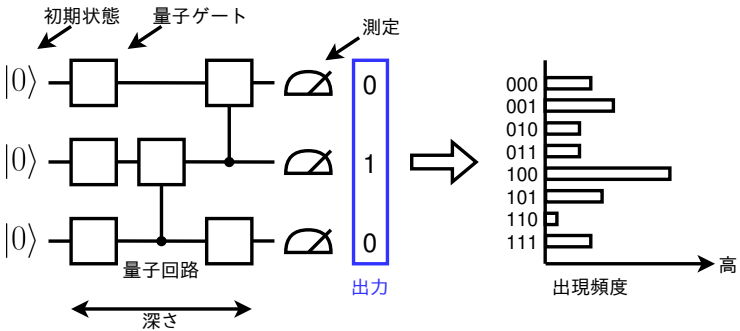
入力行列の表現



- FP16TCEC は何かに使えるか？ ⇒ 量子回路シミュレーションはどう？

量子計算機と量子回路シミュレーション

量子計算機



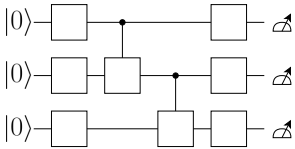
量子回路シミュレーション

ビット列が与えられた時、その出現確率 $p = |a|^2$（正確には振幅 a）を計算

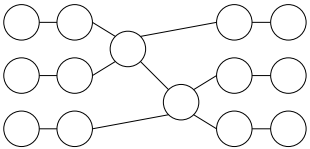
テンソルネットワーク縮約によるシミュレーション

量子回路を表現するテンソルネットワークを縮約することにより振幅を計算。

(a) 量子回路



(b) テンソルネットワーク



(c) 縮約パスの最適化

$$\mathbf{T}_{1,2} \leftarrow \mathrm{Contraction}(\mathbf{T}_1, \mathbf{T}_2, \mathrm{axis} = (0,0))$$
$$\mathbf{T}_{1,2,3} \leftarrow \mathrm{Contraction}(\mathbf{T}_{1,2}, \mathbf{T}_3, \mathrm{axis} = (1,0))$$
$$\vdots$$

$$\mathbf{M}_{1,2} \leftarrow \mathrm{Matmul}(\mathbf{M}_1, \mathbf{M}_2)$$
$$\mathbf{M}_{1,2,3} \leftarrow \mathrm{Matmul}(\mathbf{M}_{1,2}, \mathbf{M}_3)$$
$$\vdots$$

(d) TTGTにより行列積へ変換

単精度行列積エミュレーションを応用

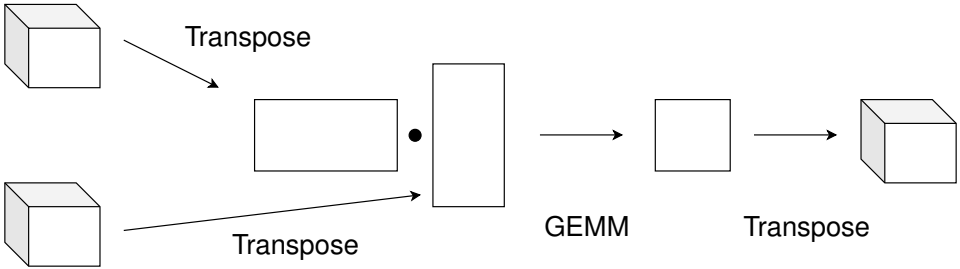
主たる計算は単精度複素行列積 ⇒ **SGEMM (CGEMM) を速くしたい。**

テンソルネットワーク縮約と TTGT

- テンソル縮約：ベクトルの内積や行列積の一般形
- テンソルネットワーク縮約：連鎖行列積の一般形
  - 縮約順序によって計算量が変わる

計算順序（縮約パス）を決定した後、、

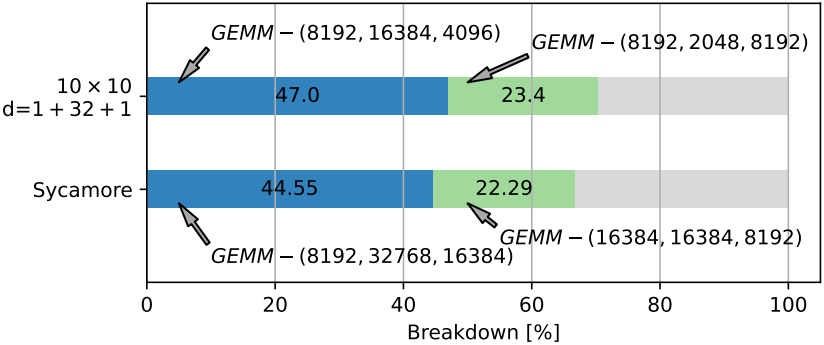
計算機上でのテンソル縮約：Transpose-Transpose-GEMM-Transpose (TTGT)



どれくらいの大きさの行列積が行われているのか？

量子超越性検証で用いられるランダム量子回路で計算時間割合を調査。

- ランダム量子回路 10×10 量子ビット、深さ $1 + 32 + 1$
- Google Sycamore 量子回路プロセッサ（53 量子ビット、深さ $1 + 20 + 1$）

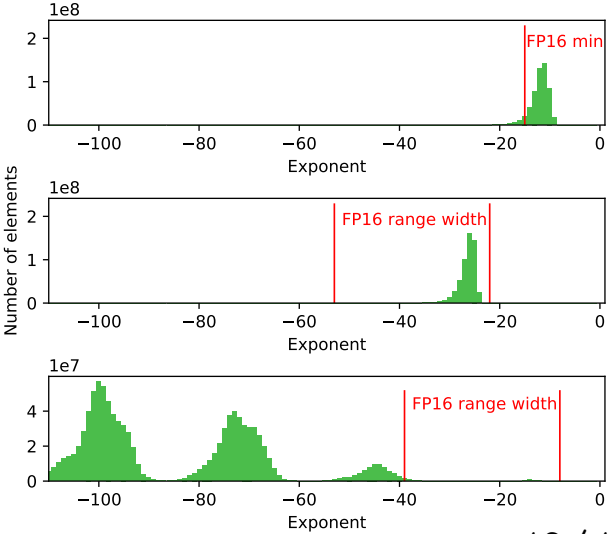


- **大きな行列積が支配的**
- 単純に TF32TCEC を適用するだけでも高速化されるが、、

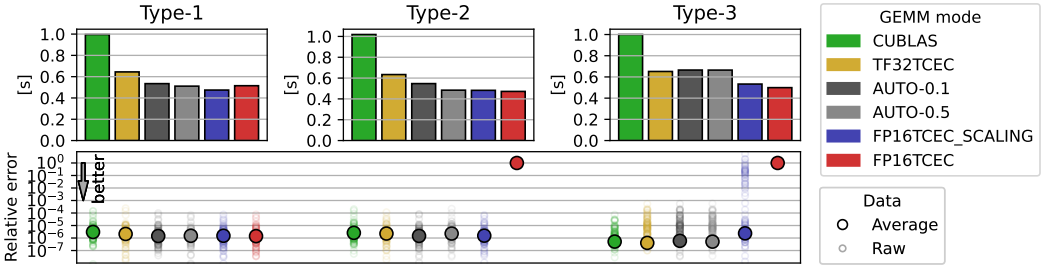
自動精度選択

入力行列の指数部分布を見て、

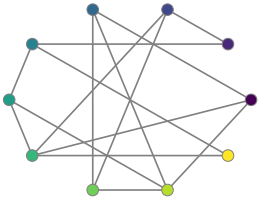
- 何もせず FP16TCEC が使える場合は「FP16TCEC」
- Scaling すれば FP16TCEC が使える場合は「Scaling+FP16TCEC」
- FP16TCEC が使えない場合は「TF32TCEC」



自動精度選択のテスト

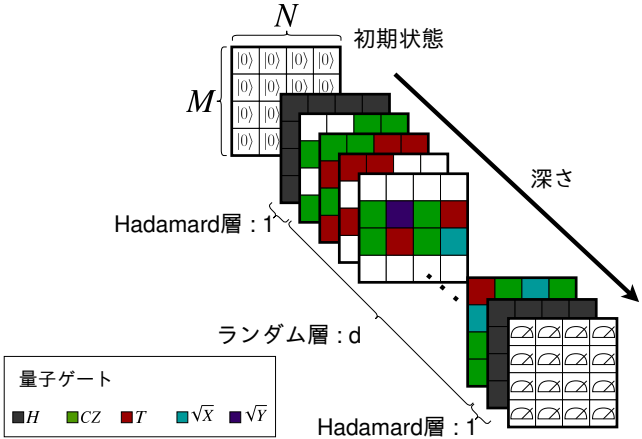


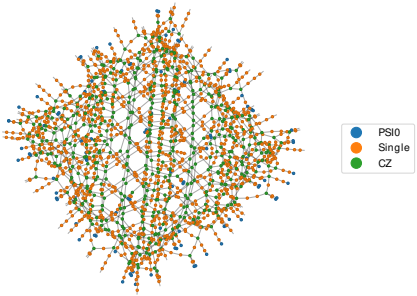
- ランダムに生成したテンソルネットワーク
- $\mathcal{T}_i \in \mathbb{C}^{128\times\cdots\times128}$、ネットワーク $\Rightarrow$
- Type-1: 正規分布 $\mathcal{N}(0, 10^{-4})$ から生成
- Type-2: Type-1 の初期化の後、全ての要素を 10^{-6} 倍
- Type-3: Type-2 の初期化の後、$10 \sim 20$ 要素を 1 へ



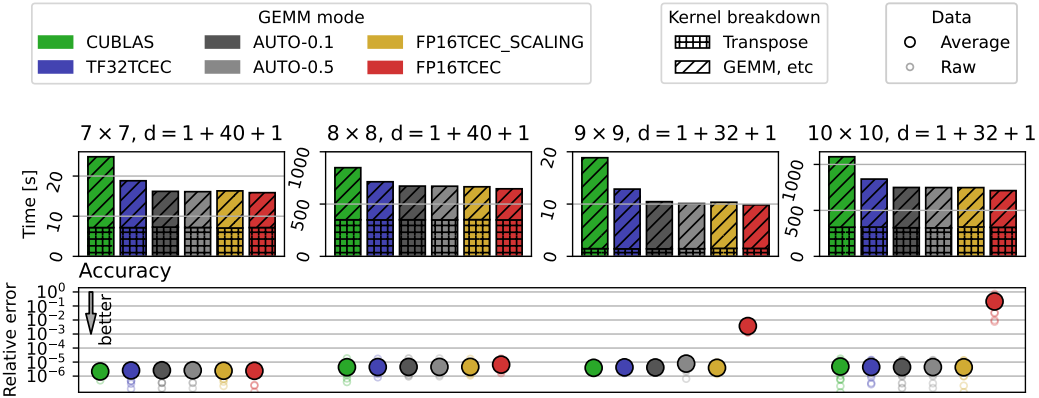
ランダム量子回路

量子超越性検証ベンチマーク



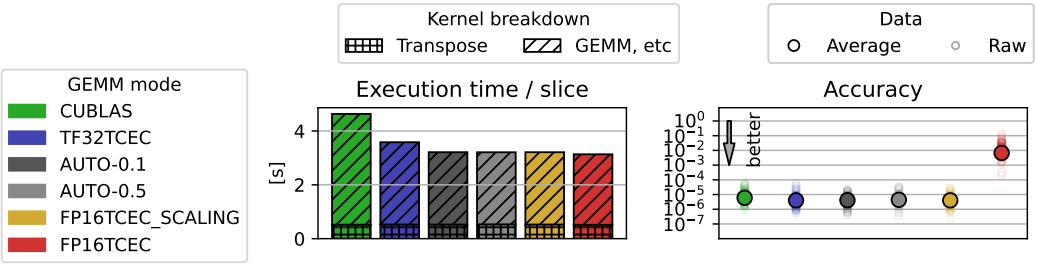


ランダム量子回路



- FP16TCEC 又は Scaling+FP16TCEC が自動選択
- 計算精度そのままで最大 1.86 倍高速化

Google Sycamore シミュレーション



- Google Sycamore 回路の縮約の 1 スライス
- FP16TCEC が自動選択
- 計算精度そのままで 1.44 倍高速化

まとめ

テーマ

深層学習用ハードウェアはそれ以外の用途に使えるか？

- ⇒ 使えた

（単精度）量子回路シミュレーションでは、

- TF32TCEC は SGEMM に置き換え可能
- FP16TCEC を使えるときに使うことでさらなる高速化

混合精度演算器の利点

- 精度補正手法により細かく・部分的に計算精度調整が可能
- 仮数部も指数部も単精度が必要ですか？

[おまけ] 倍精度計算はできる？

残念ながら、今回の手法を単純には倍精度拡張できない。

- ハードウェアサポートが必要
  - FP32 入力、FP64 出力みたいな Tensor コア

ソフトウェア的にどうにかできないの？

尾崎スキーム

- 同様に仮数部を分割する方法
- 分割の仕方を工夫することで、一部の計算を丸め誤差なく計算

"DGEMM on Tensor Cores" by Mukunoki et al., 2020

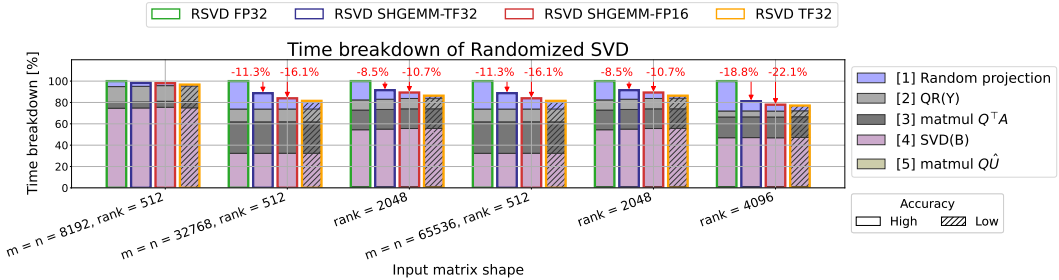
- TESLA V100 では DGEMM より遅いが、TITAN RTX 上では速い。

[おまけ] 精度エミュレーション以外に何ができるか？

例えば、、、

Randomized SVD by SHGEMM

- Randomized SVD：行列の低ランク近似 Algorithm
- SHGEMM：“FP32 行列・FP16 行列” を Tensor コアで高速に
- ランダム行列は低精度（FP16）で良くない？



- **必要な部分に必要な精度だけ**