Forward Error Bounds and Efficient Algorithms for Computing a Tensor Times Matrix Chain in Low Precision on GPUs

Julian Bellavita [1, 2] Piyush Sao [2] Ramakrishnan Kannan[2]

[1]Cornell University [2]Oak Ridge National Laboratory

1. Tensor Times Matrix Chain (TTMc) Background

Given an order-N tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ and a set of N matrices $U_1 \dots U_N$ with $U_n \in \mathbb{R}^{J_n \times I_n}$, a **Tensor Times Matrix Chain (TTMc)** contracts a tensor along each mode with one matrix:

$$\mathcal{Y} = \mathcal{X} \times_1 U_1 \times_2 U_2 \cdots \times_N U_N$$

TTMc can be implemented as a sequence of **Tensor Times Matrix (TTM)** operations, each of which can be written in terms of matrix multiplication:

$$\mathbf{Y}_{(n)} = U_n \mathbf{X}_{(n)}$$

where $\mathbf{X}_{(n)}, \mathbf{Y}_{(n)}$ denote the matricization of $\mathcal{X}, \mathcal{Y}$ along the n-th mode.

Thus, a TTMc is a sequence of General Matrix Multiply (GEMM) operations

Algorithm 1 Tensor Times Matrix Chain

Require: $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}, U_1 \in \mathbb{R}^{J_1 \times I_1}, \dots, U_N \in \mathbb{R}^{J_N \times I_N}$

1: $\mathcal{Y} = \mathcal{X}$
2: **for** $n = 1 : N$ **do**
3: $\quad \mathbf{Y}_{(n)} = U_n \mathbf{X}_{(n)} \leftarrow$ Computed using GEMM
4: **end for**

Problem Statement

Design a GPU-capable TTMc algorithm that uses low precision hardware to accelerate each GEMM while avoiding numerical overflow and minimizing rounding errors

Main Contributions

1. First forward error bound for TTMc in finite precision.
2. New Kronecker Product Scaling strategy → 2× GPU speedup in FP16.
3. Heuristic shows TTM order can reduce error growth.

2. Efficient Scaling Strategies to Avoid Overflow

Avoiding overflow requires scaling the tensor $\mathcal{X}$ and matrices $U_1 \dots U_N$ to cap the magnitude of each entry.

Strategy 1: One-Sided Scaling

Before each GEMM that computes $\mathbf{Y}_{(n)} = U_n \mathbf{X}_{(n)}$, scale the inputs:

$$\tilde{U}_n = R_n U_n, \tilde{\mathbf{Y}}_{(n)} = \mathbf{Y}_{(n)} S_n$$

where R_n, S_n are diagonal matrices with entries chosen based on the infinity-norm of each row/column of the operands.

This prevents overflow, but initializing and applying R_n, S_n requires significant data movement in high precision.

Strategy 2: Kronecker Product Scaling

Pick diagonal matrices $D_1 \dots D_N$ and scale $\mathcal{X}$ **once**:

$$\tilde{\mathbf{X}}_{(1)} = D_1 \mathbf{X}_{(1)} (D_N \otimes \cdots \otimes D_2)$$

then scale each U_n:

$$\tilde{U}_n = R_n U_n D_n^{-1}$$

the TTMc output can then be recovered in one shot:

$$\mathbf{Y}_{(n)} = R_n^{-1} \tilde{\mathbf{Y}}_{(n)} (R_N^{-1} \otimes \cdots \otimes R_{n+1}^{-1} \otimes R_{n-1}^{-1} \otimes \cdots \otimes R_1^{-1})$$

Only one pass through the input/output tensor is required – less data movement, and partial reconstruction is possible.

3. Forward Error Bound and TTM Ordering

The **forward error** of a TTMc computed in finite precision with unit roundoff μ is

$$\|\hat{\mathcal{Y}} - \mathcal{Y}\|_F \leq \|\mathcal{X}\|_F \prod_{n=1}^{N} \|U_n\|_F \Big(\sum_{i=1}^{N} \prod_{j=1}^{i} \gamma_{I_j} \Big).$$

where

$$\gamma_n \approx 1.01 n \mu$$

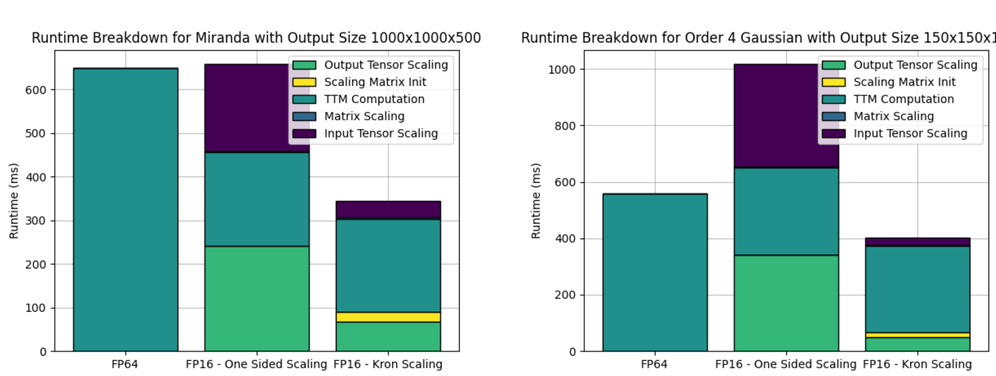
TTMc can be computed in any order – **the chosen ordering impacts the forward error bound**.

We want to select a TTMc order that minimizes

$$\min_{\text{perms of } \gamma_{I_1} \cdots \gamma_{I_N}} \Big(\sum_{i=1}^{N} \prod_{j=1}^{i} \gamma_{I_j} \Big).$$

Smallest Shared First Heuristic: Sort the TTM operations by the size of their shared dimension, smallest shared dimensions go first.

4. Runtime Breakdown Experiments

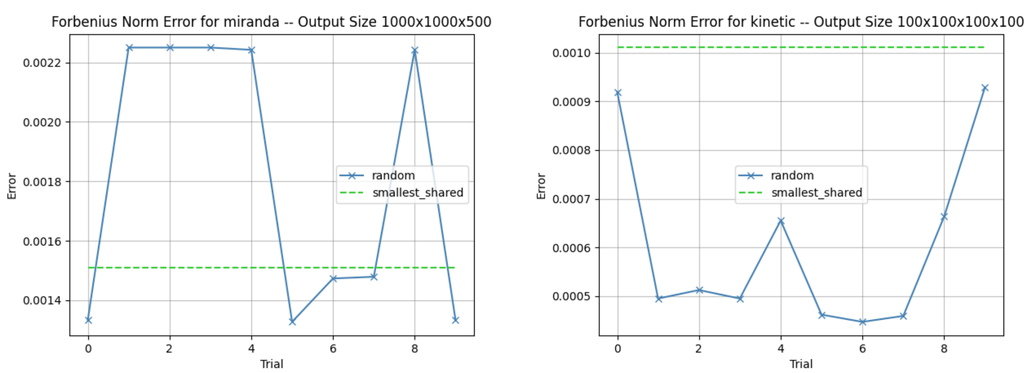


Runtime breakdown of TTMc on the Miranda Tensor and an order 4 synthetic Gaussian tensor computed in various precisions with different normalization strategies.

TTMc in FP16 is not faster than FP64 if one-sided scaling is used, but it is up to **2x** faster than FP64 if Kronecker Product scaling is used.

Speedups are more modest for higher order tensor due to smaller matrix sizes.

5. TTM Ordering Experiments



Forward error of different TTM orderings on the Miranda and Kinetic tensors. Smallest shared first is effective for Miranda, but ineffective for Kinetic.