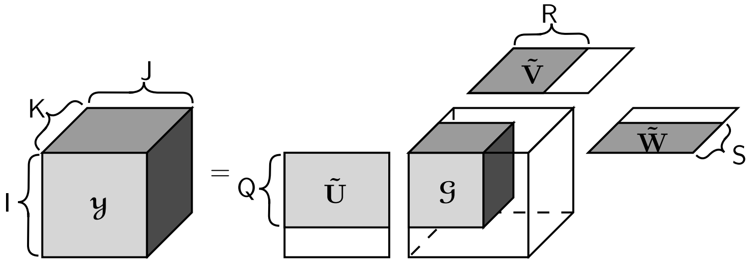
Parallel Tucker Decomposition with Numerically Accurate SVD

Zitong Li Qiming Fang Grey Ballard

50th International Conference on Parallel Processing
August 11, 2021

WAKE FOREST UNIVERSITY

- Tucker decomposition



Applications include

- Compression of multiway data (e.g. from physics simulations)
- Machine learning
- Signal processing

Gram vs QR Agorithm

Sequentially Truncated Higher-Order SVD

function $\mathrm{ST\text{-}HOSVD}(\mathcal{X}, t)$
  $\mathcal{Y} = \mathcal{X}$
  for $n = 0$ to $N-1$ **do**
    $[\mathbf{U}, \boldsymbol{\Sigma}, \sim] = \mathrm{SVD}(\mathbf{Y}_{(n)})$
    $R_n = \min\left\{R | \sum_{i=R+1}^{I_n} \sigma_i^2 \leq t^2\|\mathcal{X}\|^2/N\right\}$
    $\mathbf{U}_n = \mathbf{U}(:, 1 : R_n)$ ▷ nth factor matrix
    $\mathcal{Y} = \mathcal{Y} \times_n \mathbf{U}_n^T$ ▷ TTM truncation
  end for
  $\mathcal{G} = \mathcal{Y}$ ▷ core tensor
  return $(\mathcal{G}, \mathbf{U}_0, \ldots, \mathbf{U}_{N-1})$
end function

Gram-SVD

function $\mathrm{SVD}(\mathbf{Y})$
  $\mathbf{G} = \mathbf{Y}\mathbf{Y}^T$
  $[\boldsymbol{\Lambda}, \mathbf{U}] = \mathrm{EIG}(\mathbf{G})$
  return $[\mathbf{U}, \boldsymbol{\Lambda}^{1/2}]$
end function

QR-SVD

function $\mathrm{SVD}(\mathbf{Y})$
  $[\mathbf{Q}, \mathbf{R}] = \mathrm{QR}(\mathbf{Y}^T)$
  $[\mathbf{U}, \boldsymbol{\Sigma}, \sim] = \mathrm{SVD}(\mathbf{R}^T)$
  return $[\mathbf{U}, \boldsymbol{\Sigma}]$
end function

Numerical Accuracy Comparison

Theorem

Using a backward stable algorithm for SVD(such as the QR-SVD), the relative error of each computed singular value satisfies

$$\frac{|\tilde{\sigma}_i - \sigma_i|}{\sigma_i} = O\left(\varepsilon \frac{\|\mathbf{A}\|}{\sigma_i}\right). \tag{1}$$

Theorem

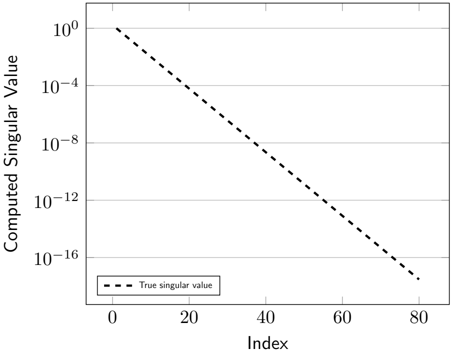
Using the Gram-SVD, the relative error of each computed singular value satisfies

$$\frac{|\tilde{\sigma}_i - \sigma_i|}{\sigma_i} = O\left(\varepsilon (\frac{\|\mathbf{A}\|}{\sigma_i})^2\right) \tag{2}$$

- if there are small singular values, Gram SVD will have trouble determining the appropriate rank, R_n:

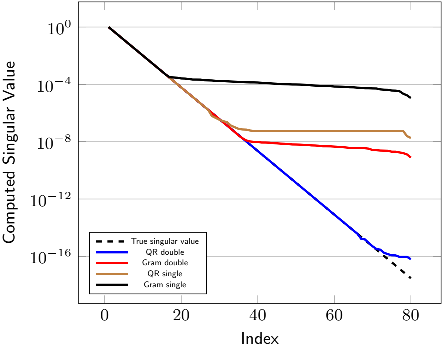
$$R_n = \min \left\{ R | \sum_{i=R+1}^{I_n} \sigma_i^2 \leq t^2 \|\mathfrak{X}\|^2 / N \right\}$$

Numerical Experiment



- 80×512000 synthetic matrix with geometrically decreasing singular values from 1 to 1e-18

Numerical Experiment



- 80×512000 synthetic matrix with geometrically decreasing singular values from 1 to 1e-18
- Gram single $<$ QR single $<$ Gram double $<$ QR double, in terms of accuracy.

Our Contributions

This project is built on top of TuckerMPI [BKK20], a parallel C++ library designed to compute Tucker decomposition efficiently.

Our goals:

1. Improve attainable accuracy for ST-HOSVD with reasonable increase in time
2. Achieve speedup over TuckerMPI while maintaining accuracy

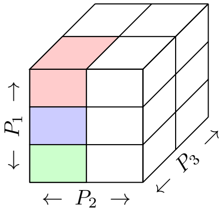
Our contributions:

1. We implemented our QR-SVD-based algorithm using TuckerMPI
2. We templated TuckerMPI to use single precision

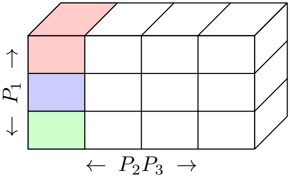
The following table shows the options that are available to the users

	Original TuckerMPI	Our implementation
Gram double	Yes	Yes
Gram single	-	Yes
QR double	-	Yes
QR single	-	Yes

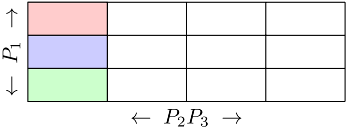
Data Distribution and Tensor Unfolding



Tensor distribution

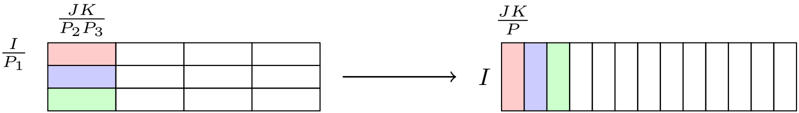


Unfold processor grid



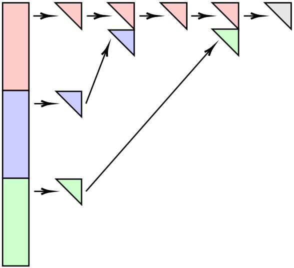
Unfold local tensor

Parallel Implementation



Redistribution

- QR is more expensive than Gram in terms of computation (almost 2x) and communication (increase by $O(\log P)$).



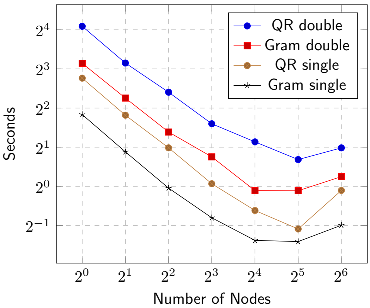
Tall Skinny QR (TSQR) [DGHL12]

Experimental Platform

- Andes is a 704-node linux cluster at the Oak Ridge Leadership Computing Facility
- Each node has 2 16-core (AMD EPYC 7302 CPU) at 3 GHz

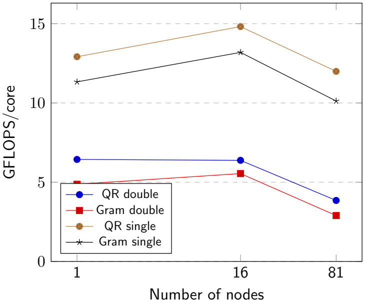
Scaling Performance on Synthetic Data

Strong Scaling



- Input: $256 \times 256 \times 256 \times 256$ tensor
- Output: $32 \times 32 \times 32 \times 32$ tensor

Weak Scaling

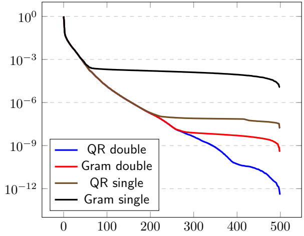


- Input: $250k \times 250k \times 250k \times 250k$ tensor where $k \in [1, 2, 3]$
- Output: $25k \times 25k \times 25k \times 25k$ tensor

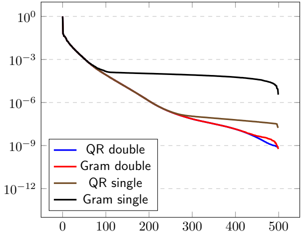
- 2x speed up from double to single precision
- QR single out-performs Gram double consistently
- Time increase from Gram double to QR double stays within 2x

Singular Values of Application Dataset

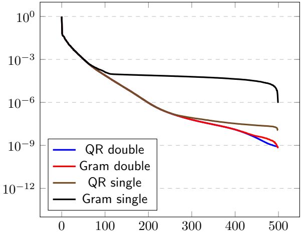
This $500 \times 500 \times 500 \times 11 \times 100$ 1.1TB dataset [KZCS16] is generated from a simulation of air-methonal combustion.



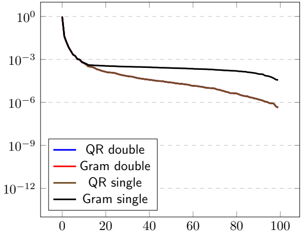
(a) Spatial Dim 1



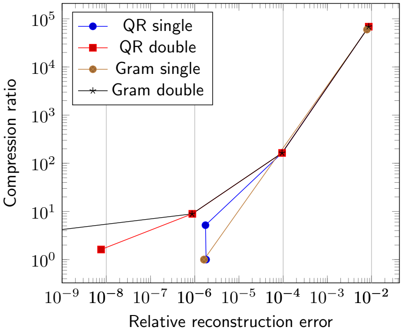
(b) Spatial Dim 2

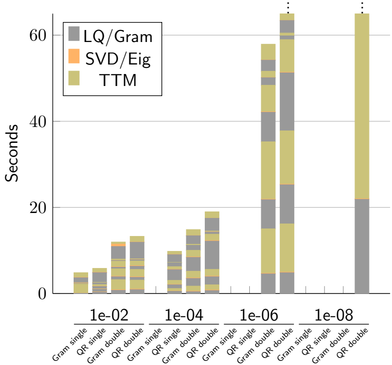


(c) Spatial Dim 3



(d) Time Steps





Tolerance on relative error	Fastest method that satisfies the tolerance
tol > 1e-2	Gram Single
1e-6 < tol < 1e-2	QR Single
1e-7 < tol < 1e-6	Gram Double
tol < 1e-7	QR Double

Summary

- Implemented QR-SVD in parallel that improved the accuracy of the Tucker decomposition
- Using single precision resulted in 60% speed up for TuckerMPI library in real data
- The newly implemented algorithms should be used in most cases instead of Gram double

Feel free to email me at liz20@wfu.edu for comments and questions

References I

G. Ballard, A. Klinvex, and T. G. Kolda.
TuckerMPI: A parallel C++/MPI software package for large-scale data compression via the tucker tensor decomposition.
ACM Transactions on Mathematical Software, 46(2), June 2020.

J. Demmel, L. Grigori, M. Hoemmen, and J. Langou.
Communication-optimal parallel and sequential QR and LU factorizations.
SIAM Journal on Scientific Computing, 34(1):A206–A239, 2012.

H. Kolla, X.-Y. Zhao, J. H. Chen, and N. Swaminathan.
Velocity and reactive scalar dissipation spectra in turbulent premixed flames.
Combustion Science and Technology, 188(9):1424–1439, 2016.