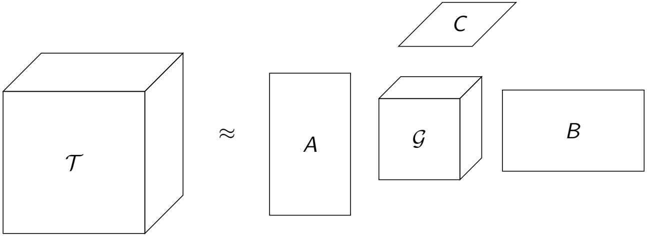
Parallel Higher-Order Orthogonal Iteration for Tucker Decomposition with Rank Adaptivity

Department of Computer Science

João Pinheiro, Grey Ballard, Aditya Devarakonda

Friday, March 7^{th} 2024

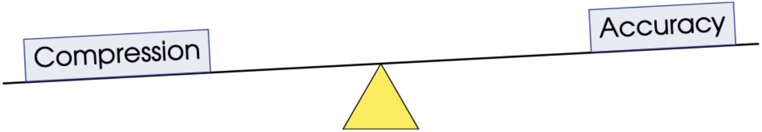
The Tucker Decomposition



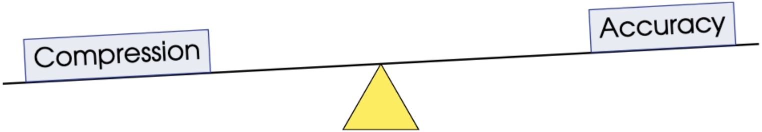
$$\mathcal{T} \approx \{\mathcal{G}; A, B, C\} = \mathcal{G} \times_1 A \times_2 B \times_3 C$$

$$t_{ijk} \approx \sum_{\alpha=1}^{q} \sum_{\beta=1}^{r} \sum_{\gamma=1}^{s} g_{\alpha\beta\gamma} \cdot a_{i\alpha} b_{j\beta} c_{i\gamma}, \; \forall (i, j, k) \in [m] \otimes [n] \otimes [p]$$

The Tucker Decomposition Trade-Off



The Tucker Decomposition Trade-Off

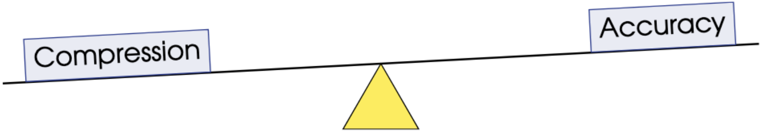


To know compression beforehand, we specify the size of the core tensor $\mathcal{G}$.

$$\text{compression ratio} = \frac{mnp}{qrs + qm + nr + sp} \approx \frac{mnp}{qrs}$$

This is the rank-specified formulation, where cannot say in advance what the error will be

The Tucker Decomposition Trade-Off



To know compression beforehand, we specify the size of the core tensor $\mathcal{G}$.

$$\text{compression ratio} = \frac{mnp}{qrs + qm + nr + sp} \approx \frac{mnp}{qrs}$$

This is the rank-specified formulation, where cannot say in advance what the error will be

To know accuracy beforehand, we specify the maximum relative error threshold

$$\frac{||\mathcal{X} - \{\mathcal{G}; U, V, W\}||}{||\mathcal{X}||} \leq \epsilon$$

This is the error-specified formulation, where cannot say in advance what the compression will be

The Two Protagonists

function $\mathrm{STHOSVD}(\mathcal{X}$, **r** or $\epsilon)$

$A \leftarrow \textcolor{blue}{LLSV}(X_{(1)}, r_1 \text{ or } \epsilon)$
$\mathcal{G} \leftarrow \mathcal{X} \times_1 A^\mathsf{T}$

$B \leftarrow \textcolor{blue}{LLSV}(G_{(2)}, r_2 \text{ or } \epsilon)$
$\mathcal{G} \leftarrow \mathcal{G} \times_1 B^\mathsf{T}$

$C \leftarrow \textcolor{blue}{LLSV}(G_{(3)}, r_3 \text{ or } \epsilon)$
$\mathcal{G} \leftarrow \mathcal{G} \times_3 C^\mathsf{T}$
return $\{\mathcal{G}; A, B, C\}$

end function

The Two Protagonists

function $\mathrm{STHOSVD}(\mathcal{X}$, **r** or $\epsilon)$

$A \leftarrow LLSV(X_{(1)}, r_1$ or $\epsilon)$

$\mathcal{G} \leftarrow \mathcal{X} \times_1 A^\intercal$

$B \leftarrow LLSV(G_{(2)}, r_2$ or $\epsilon)$

$\mathcal{G} \leftarrow \mathcal{G} \times_1 B^\intercal$

$C \leftarrow LLSV(G_{(3)}, r_3$ or $\epsilon)$

$\mathcal{G} \leftarrow \mathcal{G} \times_3 C^\intercal$

return $\{\mathcal{G}; A, B, C\}$

end function

function $U = \mathrm{LLSV}(Y, r$ or $\epsilon)$

$S = Y \cdot Y^\intercal$

$[U, \Lambda] = \mathrm{eig}(S)$

return $U(:, 1:r)$

end function

LLSV $\rightarrow$ Left Leading Singular vectors

The Two Protagonists

function $\mathrm{STHOSVD}(\mathcal{X}, \mathbf{r} \text{ or } \epsilon)$
  $A \leftarrow LLSV(X_{(1)}, r_1 \text{ or } \epsilon)$
  $\mathcal{G} \leftarrow \mathcal{X} \times_1 A^\intercal$

  $B \leftarrow LLSV(G_{(2)}, r_2 \text{ or } \epsilon)$
  $\mathcal{G} \leftarrow \mathcal{G} \times_1 B^\intercal$

  $C \leftarrow LLSV(G_{(3)}, r_3 \text{ or } \epsilon)$
  $\mathcal{G} \leftarrow \mathcal{G} \times_3 C^\intercal$
  return $\{\mathcal{G}; A, B, C\}$
end function

function $U = \mathrm{LLSV}(Y, r \text{ or } \epsilon)$
  $S = Y \cdot Y^\intercal$
  $[U, \Lambda] = \mathrm{eig}(S)$
  return $U(:, 1 : r)$
end function

LLSV $\rightarrow$ Left Leading Singular vectors

function $\mathrm{HOOI}(\mathcal{X}, \mathbf{r})$
  Initialize A, B, C randomly
  for Max Iterations **do**
    $\mathcal{Y} = \mathcal{X} \times_2 B^\intercal \times_3 C^\intercal$
    $A \leftarrow LLSV(Y_{(1)}, r_1)$

    $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_3 C^\intercal$
    $B \leftarrow LLSV(Y_{(2)}, r_2)$

    $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_2 B^\intercal$
    $C \leftarrow LLSV(Y_{(3)}, r_3)$
  end for
  $\mathcal{G} \leftarrow \mathcal{Y} \times_3 C^\intercal$
  return $\{\mathcal{G}; A, B, C\}$
end function

Optimization 1: Dimension Tree Memoization

function $\mathrm{HOOI}(\mathcal{X}, \mathbf{r})$
  Initialize A, B, C randomly
  for Max Iterations **do**
    $\mathcal{Y} = \mathcal{X} \times_2 B^\intercal \times_3 C^\intercal$
    $A \leftarrow \mathit{LLSV}(Y_{(1)}, r_1)$

    $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_3 C^\intercal$
    $B \leftarrow \mathit{LLSV}(Y_{(2)}, r_2)$

    $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_2 B^\intercal$
    $C \leftarrow \mathit{LLSV}(Y_{(3)}, r_3)$
  end for
  $\mathcal{G} \leftarrow \mathcal{Y} \times_3 C^\intercal$
  return $\{\mathcal{G}; A, B, C\}$
end function

Optimization 1: Dimension Tree Memoization

function $\text{HOOI}(\mathcal{X}, \mathbf{r})$
    Initialize A, B, C randomly
    for Max Iterations **do**
        $\mathcal{Y} = \mathcal{X} \times_2 B^\intercal \times_3 C^\intercal$
        $A \leftarrow LLSV(Y_{(1)}, r_1)$

        $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_3 C^\intercal$
        $B \leftarrow LLSV(Y_{(2)}, r_2)$

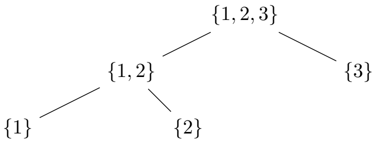
        $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_2 B^\intercal$
        $C \leftarrow LLSV(Y_{(3)}, r_3)$
    end for
    $\mathcal{G} \leftarrow \mathcal{Y} \times_3 C^\intercal$
    return $\{\mathcal{G}; A, B, C\}$
end function

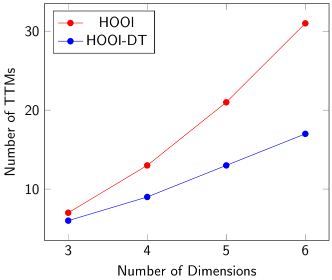
function $\text{HOOI-DT}(\mathcal{X}, \mathbf{r})$
    Initialize A, B, C randomly
    for Max Iterations **do**
        $\mathcal{Y}_{\text{temp}} = \mathcal{X} \times_3 C^\intercal$
        $\mathcal{Y} = \mathcal{Y}_{\text{temp}} \times_2 B^\intercal$
        $A \leftarrow LLSV(Y_{(1)}, r_1)$

        $\mathcal{Y} = \mathcal{Y}_{\text{temp}} \times_1 A^\intercal$
        $B \leftarrow LLSV(Y_{(2)}, r_2)$

        $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_2 B^\intercal$
        $C \leftarrow LLSV(Y_{(3)}, r_3)$
    end for
    $\mathcal{G} \leftarrow \mathcal{Y} \times_3 C^\intercal$
    return $\{\mathcal{G}; A, B, C\}$
end function

Optimization 1: Dimension Tree Memoization





function $\mathrm{HOOI\text{-}DT}(\mathcal{X}, \mathbf{r})$

Initialize A, B, C randomly

for Max Iterations **do**

$\mathcal{Y}_{\text{temp}} = \mathcal{X} \times_3 C^\intercal$

$\mathcal{Y} = \mathcal{Y}_{\text{temp}} \times_2 B^\intercal$

$A \leftarrow LLSV(Y_{(1)}, r_1)$

$\mathcal{Y} = \mathcal{Y}_{\text{temp}} \times_1 A^\intercal$

$B \leftarrow LLSV(Y_{(2)}, r_2)$

$\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_2 B^\intercal$

$C \leftarrow LLSV(Y_{(3)}, r_3)$

end for

$\mathcal{G} \leftarrow \mathcal{Y} \times_3 C^\intercal$

return $\{\mathcal{G}; A, B, C\}$

end function

Optimization 2: Subspace Iterations

function $\mathrm{HOSI}(\mathcal{X}, \mathbf{r})$
    Initialize A, B, C randomly
    for Max Iterations **do**
        $\mathcal{Y} = \mathcal{X} \times_2 B^\intercal \times_3 C^\intercal$
        $A \leftarrow \mathit{LLSV}(\mathcal{Y}, A, 1)$

        $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_3 C^\intercal$
        $B \leftarrow \mathit{LLSV}(\mathcal{Y}, B, 2)$

        $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_2 B^\intercal$
        $C \leftarrow \mathit{LLSV}(\mathcal{Y}, C, 3)$
    end for
    $\mathcal{G} \leftarrow \mathcal{Y} \times_3 C^\intercal$
    return $\{\mathcal{G}; A, B, C\}$
end function

Optimization 2: Subspace Iterations

function $\mathrm{HOSI}(\mathcal{X}, \mathbf{r})$
  Initialize A, B, C randomly
  for Max Iterations **do**
    $\mathcal{Y} = \mathcal{X} \times_2 B^\intercal \times_3 C^\intercal$
    $A \leftarrow \color{red}{LLSV}(\mathcal{Y}, A, 1)$

    $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_3 C^\intercal$
    $B \leftarrow \color{red}{LLSV}(\mathcal{Y}, B, 2)$

    $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_2 B^\intercal$
    $C \leftarrow \color{red}{LLSV}(\mathcal{Y}, C, 3)$
  end for
  $\mathcal{G} \leftarrow \mathcal{Y} \times_3 C^\intercal$
  return $\{\mathcal{G}; A, B, C\}$
end function

function $U = \color{red}{\mathrm{LLSV}}(\mathcal{Y}, U, n)$
  $\mathcal{G} = \mathcal{Y} \times_n U^\intercal$
  $U = \mathrm{Contract}(\mathcal{Y}, \mathcal{G}, n)$ $\triangleright$ TTT
  $[U, \sim] = \mathrm{qr}(U)$
end function

function $U = \color{blue}{\mathrm{LLSV}}(Y, r \text{ or } \epsilon)$
  $S = Y \cdot Y^\intercal$
  $[U, \Lambda] = \mathrm{eig}(S)$
  return $U(:, 1:r)$
end function

Input: Tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$

Ranks $\mathbf{r} = r_1, \ldots, r_d$

Output: TTensor $\mathcal{T}$ of ranks $\mathbf{r}$ with $\mathcal{T} \approx \mathcal{X}$

function $\text{HOSI-DT}(\mathcal{X}, \mathbf{U}, \mathbf{m}, \mathbf{r})$

  if length($\mathbf{m}$) == 1 **then**

    $\mathcal{G} = \mathcal{X} \times_m U_m^\intercal$ ▷ Update Core

    $U_m = Y_{(m)} \cdot G_{(m)}^\intercal$ ▷ Contract Two Tensors on Mode k

    $[U_m, \sim] = \text{qr}(U_m)$ ▷ Orthogonalize Factor Matrix

  else

    Equally partition m $= [\mu, \eta]$

    $\mathcal{X}_{\text{left}} = \mathcal{X} \times_i \mathbf{U}_i, \forall i \in \eta$

    $[\mathcal{G}, \mathbf{U}] = \text{HOSI-DT}(\mathcal{X}_{\text{left}}, \mathbf{U}, \mu, \mathbf{r})$ ▷ Left Recursion

    $\mathcal{X}_{\text{right}} = \mathcal{X} \times_i \mathbf{U}_i, \forall i \in \mu$

    $[\mathcal{G}, \mathbf{U}] = \text{HOSI-DT}(\mathcal{X}_{\text{right}}, \mathbf{U}, \eta, \mathbf{r})$ ▷ Right Recursion

  end if

end function

Transforming HOOI into an error-specified algorithm

$$\min ||\mathcal{X} - \mathcal{G} \times_1 A \times_2 B \times_3 C_3||$$
$$\text{subject to } \mathcal{G} \in \mathbb{R}^{q\times r\times s}, A \in \mathbb{R}^{m\times q}, B \in \mathbb{R}^{n\times r}, C \in \mathbb{R}^{p\times s}$$

Now suppose we have a relative error tolerance of how accurate we want our approximation to be. Then the approximation must satisfy:

$$\frac{||\mathcal{X} - \mathcal{T}||}{||\mathcal{X}||} \leq \epsilon$$
$$||\mathcal{X} - \mathcal{T}||^2 \leq \epsilon^2 \cdot ||\mathcal{X}||^2$$
$$||\mathcal{X}||^2 - ||\mathcal{G}||^2 \leq \epsilon^2 \cdot ||\mathcal{X}||^2$$
$$(1 - \epsilon^2) \cdot ||\mathcal{X}||^2 \leq ||\mathcal{G}||^2$$

Transforming HOOI into an error-specified algorithm

$$\min||\mathcal{X} - \mathcal{G} \times_1 A \times_2 B \times_3 C_3|| \tag{1}$$
$$\text{subject to } \mathcal{G} \in \mathbb{R}^{q\times r\times s}, A \in \mathbb{R}^{m\times q}, B \in \mathbb{R}^{n\times r}, C \in \mathbb{R}^{p\times s}$$

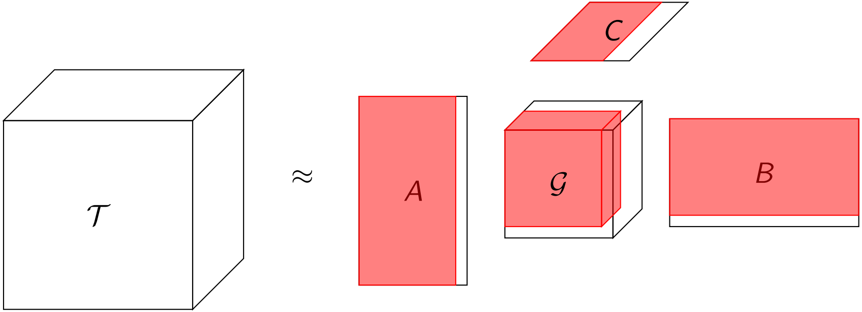
Now suppose we have a relative error tolerance of how accurate we want our approximation to be. Then the approximation must satisfy:

$$(1-\epsilon^2)\cdot||\mathcal{X}||^2 \leq ||\mathcal{G}||^2 \tag{2}$$

Then we know if our tucker tensor satisfies the error tolerance solely on $||\mathcal{G}||^2$. If the error tolerance is satisfied, we can find a smaller tucker representation that still satisfies the error by analyzing the core

Visualizing Adaptive HOOI

$$\min_{\mathbf{r}} ||\mathcal{G}(1:\mathbf{r})||^2 \qquad (3)$$
$$\text{subject to } ||\mathcal{G}(1:\mathbf{r})||^2 \geq (1-\epsilon^2)||\mathcal{X}||^2$$



Adaptive HOOI

function ADAPTIVEHOOI($\mathcal{X}$, $\mathbf{r}$, ϵ)
    Initialize A, B, C randomly
    for Maximum Number of Iterations **do**
        $\mathcal{Y} = \mathcal{X} \times_2 B^\intercal \times_3 C^\intercal$
        $A \leftarrow \text{LLSV}(Y_{(1)}, r_1)$

        $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_3 C^\intercal$
        $B \leftarrow \text{LLSV}(Y_{(2)}, r_2)$

        $\mathcal{Y} = \mathcal{X} \times_1 A^\intercal \times_2 B^\intercal$
        $C \leftarrow \text{LLSV}(Y_{(3)}, r_3)$
        $\mathcal{G} \leftarrow \mathcal{Y} \times_3 C^\intercal$
        $\mathbf{r} = \text{performCoreAnalysis}(\mathcal{G}, \epsilon, \mathbf{r})$
    end for
    return $[\mathcal{G}, U_{1:d}]$
end function

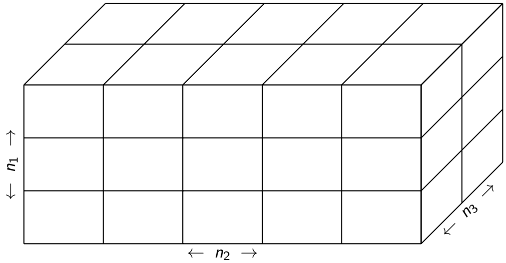
function PERFORMCOREANALYSIS($\mathcal{G}, \epsilon, \mathbf{r}$)
    if $||\mathcal{G}||^2 \geq (1-\epsilon^2)||\mathcal{X}||^2$ **then**
        Find $\mathbf{r} = \arg\min ||\mathcal{G}(1:\mathbf{r})||^2$
            subject to $||\mathcal{G}(1:\mathbf{r})||^2 \geq (1-\epsilon^2)||\mathcal{X}||^2$

        Truncate $\mathcal{G}, A, B, C$ according to $\mathbf{r}$
    else
        $\mathbf{r} = \alpha\ \mathbf{r}$
        Increase columns of A, B, C according to $\mathbf{r}$
    end if
    return r
end function

TuckerMPI and Parallel Tensor Distribution

- Existing C++/MPI library
- Implements deterministic STHOSVD
- Has efficient sequential and parallel kernels for SVD and TTM

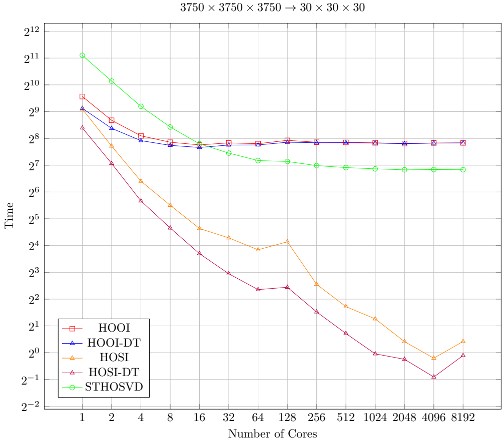
For d-way tensor, we use d-way processor grid with Cartesian block distribution



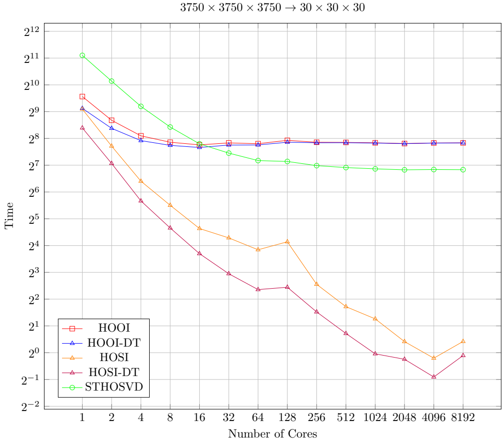
Example: $p_1 \times p_2 \times p_3 = 3 \times 5 \times 2$

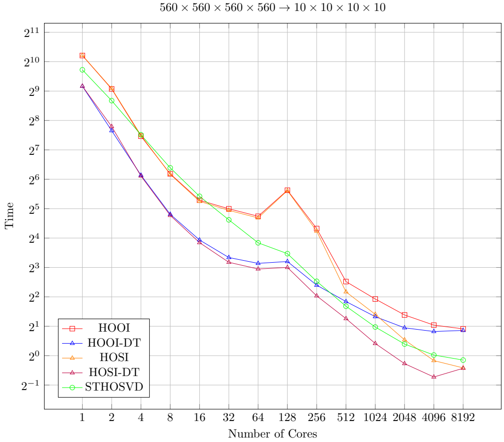
Local tensor size: $\frac{n_1}{p_1} \times \frac{n_2}{p_2} \times \frac{n_3}{p_3}$

Parallel Scaling of Synthetic Data In Single Precision

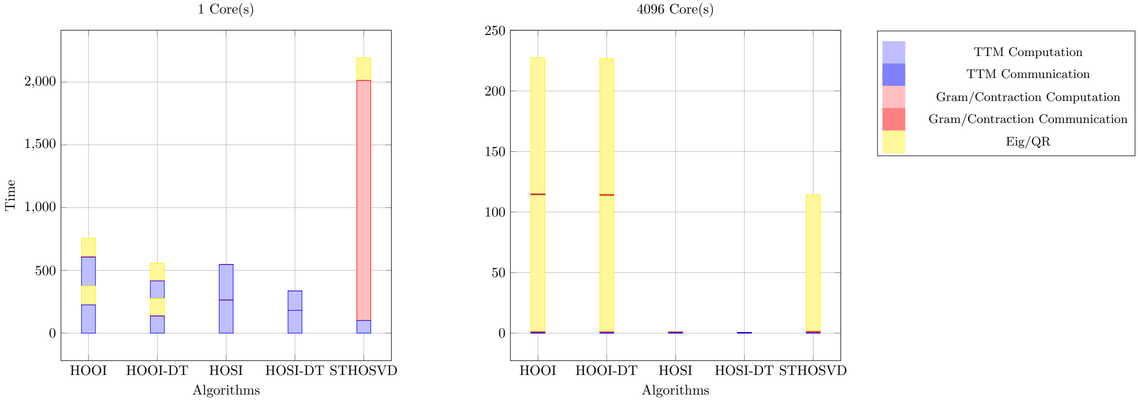


Parallel Scaling of Synthetic Data In Single Precision

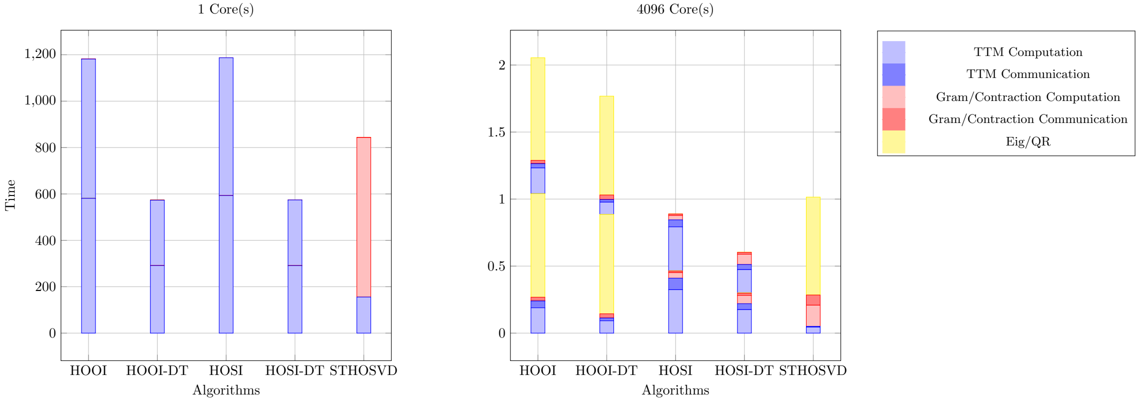




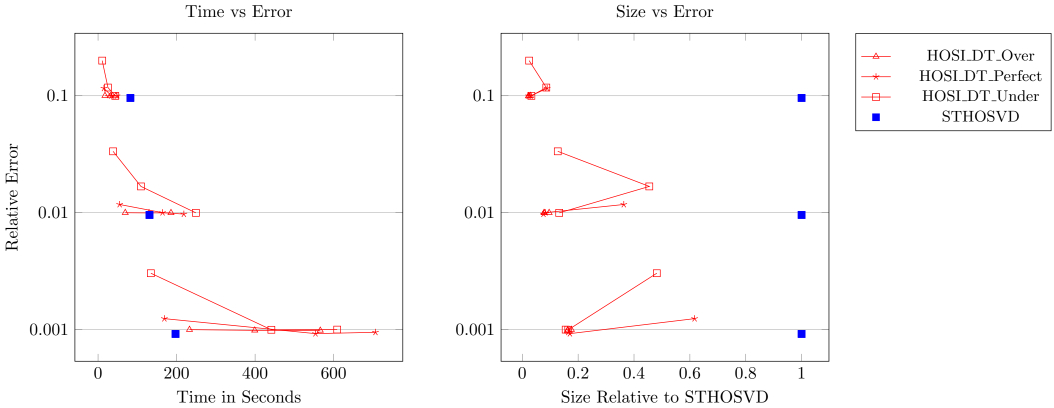
Breakdown of 3way Parallel Scaling of Synthetic Data in Single Precision



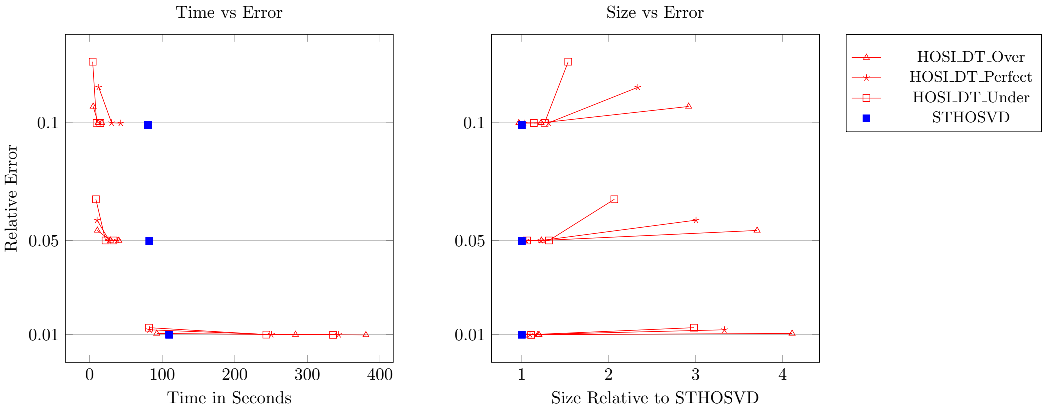
Breakdown of 4way Parallel Scaling of Synthetic Data in Single Precision



Homogeneous Charge Compression Ignition (HCCI) Data Set ($672 \times 672 \times 32 \times 626$)



Miranda Data Set (3072 × 3072 × 3072)



Conclusions, Closing Remarks, and Plug

- Dimension Tree optimizes TTM operations on HOOI
- Subspace Iteration optimizes LLSV operations on HOOI
- AdaptiveHOOI removes rank-specified constraint on HOOI
- `https://gitlab.com/tensors/TuckerMPI`
- This work will be soon be published to `arXiv` with more detail

Conclusions, Closing Remarks, and Plug

- Dimension Tree optimizes TTM operations on HOOI
- Subspace Iteration optimizes LLSV operations on HOOI
- AdaptiveHOOI removes rank-specified constraint on HOOI
- `https://gitlab.com/tensors/TuckerMPI`
- This work will be soon be published to `arXiv` with more detail
- Available August 2025 from Cambridge University Press:
- `https://users.wfu.edu/ballard/pdfs/tensor_textbook.pdf`