Rank revealing factorizations, and low rank approximations

L. Grigori

Inria Paris, Sorbonne University

November 2021
Plan

Low rank matrix approximation

Low rank approximation based on max-vol

Rank revealing QR factorization

LU_CRTMP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTMP
Plan

Low rank matrix approximation

Low rank approximation based on max-vol

Rank revealing QR factorization

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP
Low rank matrix approximation

- Problem: given \( A \in \mathbb{R}^{m \times n} \), compute rank-\( k \) approximation \( ZW^T \), where
  \( Z \) is \( m \times k \) and \( W^T \) is \( k \times n \).

- Problem with diverse applications
  - from scientific computing: fast solvers for integral equations, H-matrices
  - to data analytics: principal component analysis, image processing, ...

\[
A x \rightarrow ZW^T x
\]

\[
\text{Flops} \quad 2mn \rightarrow 2(m + n)k
\]
For any given $A \in \mathbb{R}^{m \times n}$, $m \geq n$ its singular value decomposition is

$$A = U \Sigma V^T = (U_1 \ U_2 \ U_3) \cdot \begin{pmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{pmatrix} \cdot (V_1 \ V_2)^T$$

where

- $U \in \mathbb{R}^{m \times m}$ is orthogonal matrix, the left singular vectors of $A$, $U_1$ is $m \times k$, $U_2$ is $m \times n - k$, $U_3$ is $m \times m - n$
- $\Sigma \in \mathbb{R}^{m \times n}$, its diagonal is formed by $\sigma_1(A) \geq \ldots \geq \sigma_n(A) \geq 0$ $\Sigma_1$ is $k \times k$, $\Sigma_2$ is $n - k \times n - k$
- $V \in \mathbb{R}^{n \times n}$ is orthogonal matrix, the right singular vectors of $A$, $V_1$ is $n \times k$, $V_2$ is $n \times n - k$
Properties of SVD

Given \( A = U\Sigma V^T \), we have

- \( A^T A = V\Sigma^T \Sigma V^T \),
  the right singular vectors of \( A \) are a set of orthonormal eigenvectors of \( A^T A \).

- \( AA^T = U\Sigma^T \Sigma U^T \),
  the left singular vectors of \( A \) are a set of orthonormal eigenvectors of \( AA^T \).

- The non-negative singular values of \( A \) are the square roots of the non-negative eigenvalues of \( A^T A \) and \( AA^T \).

- If \( \sigma_k \neq 0 \) and \( \sigma_{k+1}, \ldots, \sigma_n = 0 \), then
  \[
  \text{Range}(A) = \text{span}(U_1), \quad \text{Null}(A) = \text{span}(V_2),
  \text{Range}(A^T) = \text{span}(V_1), \quad \text{Null}(A) = \text{span}(U_2 U_3).
  \]
Norms

\[ \|A\|_p = \max_{\|x\|_p = 1} \|Ax\|_p \]

\[ \|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2} = \sqrt{\sigma_1^2(A) + \ldots + \sigma_n^2(A)} \]

\[ \|A\|_2 = \sigma_{\text{max}}(A) = \sigma_1(A) \]

Some properties:

\[ \max_{i,j} |A(i, j)| \leq \|A\|_2 \leq \sqrt{mn} \max_{i,j} |A(i, j)| \]

\[ \|A\|_2 \leq \|A\|_F \leq \sqrt{\min(m, n)} \|A\|_2 \]

Orthogonal Invariance: If \( Q \in \mathbb{R}^{m \times m} \) and \( Z \in \mathbb{R}^{n \times n} \) are orthogonal, then

\[ \|QAZ\|_F = \|A\|_F \]

\[ \|QAZ\|_2 = \|A\|_2 \]
Low rank matrix approximation

Best rank-$k$ approximation $A_k = U_k \Sigma_k V_k$ is rank-$k$ truncated SVD of $A$ [Eckart and Young, 1936]

$$\min_{\text{rank}(\tilde{A}_k) \leq k} \|A - \tilde{A}_k\|_2 = \|A - A_k\|_2 = \sigma_{k+1}(A)$$

(1)

$$\min_{\text{rank}(\tilde{A}_k) \leq k} \|A - \tilde{A}_k\|_F = \|A - A_k\|_F = \sqrt{\sum_{j=k+1}^{n} \sigma_j^2(A)}$$

(2)

Image, size 1190 × 1920

Rank-10 approximation, SVD

Image source: https://pixabay.com/photos/billiards-ball-play-number-half-4345870/
Matrix $A$ might not exist entirely at a given time, rows or columns are added progressively.

- **Streaming algorithm:** can solve an arbitrarily large problem with one pass over the data (a row or a column at a time).
- **Weakly streaming algorithm:** can solve a problem with $O(1)$ passes over the data.

Matrix $A$ might exist only implicitly, and it is never formed explicitly.
Communication optimal if computing a rank-k approximation on $P$ processors requires

$$\# \text{ messages} = \Omega(\log_2 P).$$
Communication optimal if computing a rank-k approximation on $P$ processors requires

$$\# \text{ messages} = \Omega \left( \log_2 P \right).$$
Idea underlying many algorithms

Compute $\tilde{A}_k = PA$, where $P = P^o$ or $P = P^{so}$ is obtained as:

1. Construct a low dimensional subspace $X = \text{range}(A\Omega_1)$, $\Omega_1 \in \mathbb{R}^{n \times l}$ that approximates well the range of $A$, e.g.

   $$\|A - P^o A\|_2 \leq \gamma \sigma_{k+1}(A), \text{ for some } \gamma \geq 1,$$

   where $Q_1$ is orth. basis of $(A\Omega_1)$

   $$P^o = A\Omega_1(A\Omega_1)^+ = Q_1Q_1^T,$$ or equiv $P^o a_j := \arg\min_{x \in X} \|x - a_j\|_2$

2. Select a semi-inner product $\langle \Theta_1 \cdot, \Theta_1 \cdot \rangle_2$, $\Theta_1 \in \mathbb{R}^{l' \times m}$ $l' \geq l$, define

   $$P^{so} = A\Omega_1(\Theta_1 A\Omega_1)^+ \Theta_1,$$ or equiv $P^{so} a_j := \arg\min_{x \in X} \|\Theta_1(x - a_j)\|_2$
Idea underlying many algorithms

Compute \( \tilde{A}_k = PA \), where \( P = P^o \) or \( P = P^{so} \) is obtained as:

1. Construct a low dimensional subspace \( X = \text{range}(A\Omega_1) \), \( \Omega_1 \in \mathbb{R}^{n \times l} \) that approximates well the range of \( A \), e.g.

\[
\|A - P^o A\|_2 \leq \gamma \sigma_{k+1}(A), \text{ for some } \gamma \geq 1,
\]

where \( Q_1 \) is orth. basis of \((A\Omega_1)\)

\[
P^o = A\Omega_1(A\Omega_1)^+ = Q_1 Q_1^T, \text{ or equiv } P^o a_j := \text{arg min}_{x \in X} \|x - a_j\|_2
\]

2. Select a semi-inner product \( \langle \Theta_1 \cdot, \Theta_1 \cdot \rangle_2 \), \( \Theta_1 \in \mathbb{R}^{l' \times m} \) \( l' \geq l \), define

\[
P^{so} = A\Omega_1(\Theta_1 A\Omega_1)^+ \Theta_1, \text{ or equiv } P^{so} a_j := \text{arg min}_{x \in X} \|\Theta_1(x - a_j)\|_2
\]
Properties of the approximations

Definitions and some of the results taken from [?].

Definition
[low-rank approximation] A matrix $A_k$ satisfying $\|A - A_k\|_2 \leq \gamma \sigma_{k+1}(A)$ for some $\gamma \geq 1$ will be said to be a $(k, \gamma)$ low-rank approximation of $A$.

Definition
[spectrum preserving] If $A_k$ satisfies

$$\sigma_j(A) \geq \sigma_j(A_k) \geq \gamma^{-1} \sigma_j(A)$$

for $j \leq k$ and some $\gamma \geq 1$, it is a $(k, \gamma)$ spectrum preserving.

Definition
[kernel approximation] If $A_k$ satisfies

$$\sigma_{k+j}(A) \leq \sigma_j(A - A_k) \leq \gamma \sigma_{k+j}(A)$$

for $1 \leq j \leq n - k$ and some $\gamma \geq 1$, it is a $(k, \gamma)$ kernel approximation of $A$. 
Plan

Low rank matrix approximation

Low rank approximation based on max-vol

Rank revealing QR factorization

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP
Theorem ([Goreinov and Tyrtyshnikov, 2001, Thm. 2.1]) Given the matrix

\[ A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \] (3)

where \( A_{11} \in \mathbb{R}^{k \times k} \) has maximal volume (i.e., maximum determinant in absolute value) among all \( k \times k \) submatrices of \( A \), then we have

\[ \| S(A_{11}) \|_{\text{max}} \leq (k + 1)\sigma_{k+1}, \] (4)

where \( S(A_{11}) = A_{22} - A_{21}A_{11}^{-1}A_{12} \).

But finding a submatrix with maximum volume is NP-hard [Civril and Magdon-Ismail, 2013].
Plan

Low rank matrix approximation

Low rank approximation based on max-vol

Rank revealing QR factorization

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP
Rank revealing QR factorization

Given $A$ of size $m \times n$, consider the decomposition

$$AP_c = QR = Q \begin{bmatrix} R_{11} & R_{12} \\ R_{22} & \end{bmatrix},$$

(5)

where $R_{11}$ is $k \times k$, $P_c$ and $k$ are chosen such that $\|R_{22}\|_2$ is small and $R_{11}$ is well-conditioned.

- By the interlacing property of singular values [Golub, Van Loan, 4th edition, page 487],

  $$\sigma_i(R_{11}) \leq \sigma_i(A) \quad \text{and} \quad \sigma_j(R_{22}) \geq \sigma_{k+j}(A)$$

  for $1 \leq i \leq k$ and $1 \leq j \leq n - k$.

- $\sigma_{k+1}(A) \leq \sigma_{\text{max}}(R_{22}) = \|R_{22}\|_2$
Given $A$ of size $m \times n$, consider the decomposition

$$AP_c = QR = Q \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}.$$

If $\|R_{22}\|_2$ is small,

- $Q(\cdot, 1 : k)$ forms an approximate orthogonal basis for the range of $A$,
  
  $$A(\cdot, j) = \sum_{i=1}^{\min(j, k)} R(i, j)Q(\cdot, i) \in \text{span}\{Q(\cdot, 1), \ldots Q(\cdot, k)\}$$

  $$\text{Range}(A) \in \text{span}\{Q(\cdot, 1), \ldots Q(\cdot, k)\}$$

- $P_c \begin{bmatrix} -R_{11}^{-1}R_{12} \\ I \end{bmatrix}$ is an approximate right null space of $A$. 
The factorization from equation (7) is rank revealing if

$$1 \leq \frac{\sigma_i(A)}{\sigma_i(R_{11})}, \frac{\sigma_j(R_{22})}{\sigma_{k+j}(A)} \leq \gamma_1(n, k),$$

for $1 \leq i \leq k$ and $1 \leq j \leq \min(m, n) - k$, where

$$\sigma_{\max}(A) = \sigma_1(A) \geq \ldots \geq \sigma_{\min}(A) = \sigma_n(A)$$

It is **strong** rank revealing [Gu and Eisenstat, 1996] if in addition

$$\| R_{11}^{-1} R_{12} \|_{\max} \leq \gamma_2(n, k)$$
Low rank approximation with strong RRQR

Given $A \in \mathbb{R}^{m \times n}$ and $R_{11} \in \mathbb{R}^{k \times k}$,

$$AP_c = QR = (Q_1 \quad Q_2) \begin{pmatrix} R_{11} & R_{12} \\ R_{22} \end{pmatrix},$$

$$\tilde{A}_{qr} = Q_1 (R_{11} \quad R_{12}) P_c^T = Q_1 Q_1^T A = P^o A$$

It can be shown that

$$\sigma_j(R_{22}) = \sigma_j(A - \tilde{A}_{qr})$$

[Gu and Eisenstat, 1996] show that given $k$ and $f$, there exists permutation $V \in \mathbb{R}^{n \times n}$ such that the factorization satisfies,

$$1 \leq \frac{\sigma_i(A)}{\sigma_i(R_{11})}, \frac{\sigma_j(R_{22})}{\sigma_{k+j}(A)} \leq \gamma(n, k), \quad \gamma(n, k) = \sqrt{1 + f^2 k(n - k)}$$

$$\|R_{11}^{-1} R_{12}\|_{max} \leq f$$

for $1 \leq i \leq k$ and $1 \leq j \leq \min(m, n) - k$.

Cost: $4mnk$ (QRCP) plus $O(mnk)$ flops and $O(k \log_2 P)$ messages.

$\tilde{A}_{qr}$ with strong RRQR is $(k, \gamma(n, k))$ spectrum preserving and kernel approximation of $A$. 

19 of 64
Given $A \in \mathbb{R}^{m \times n}$ and $R_{11} \in \mathbb{R}^{k \times k}$,

\[
AP_c = QR = (Q_1 Q_2) \begin{pmatrix} R_{11} & R_{12} \\ R_{22} & \end{pmatrix},
\]

\[
\tilde{A}_{qr} = Q_1 (R_{11} R_{12}) P_c^T = Q_1 Q_1^T A = P^o A
\]

- We show that

\[
\sigma_j(R_{22}) = \sigma_j(A - \tilde{A}_{qr})
\]

\[
\sigma_j(A - \tilde{A}_{qr}) = \sigma_j(A - Q_1 Q_1^T A) = \sigma_j(Q_2 Q_2^T A) = \sigma_j(Q_2 (0 R_{22}) P_c^{-1}) = \sigma_j(R_{22})
\]
QR with column pivoting [Businger and Golub, 1965]

Idea:
- At first iteration, trailing columns decomposed into parallel part to first column (or $e_1$) and orthogonal part (in rows 2 : $m$).
- The column of maximum norm is the column with largest component orthogonal to the first column.

Implementation:
- Find at each step of the QR factorization the column of maximum norm.
- Permute it into leading position.
- If rank($A$) = $k$, at step $k + 1$ the maximum norm is 0.
- No need to compute the column norms at each step, but just update them since

$$Q^T v = w = \begin{bmatrix} w_1 \\ w(2 : n) \end{bmatrix}, \quad ||w(2 : n)||_2^2 = ||v||_2^2 - w_1^2$$
QR with column pivoting [Businger and Golub, 1965]

Sketch of the algorithm

column norm vector: $\text{colnrm}(j) = ||A(:, j)||_2, j = 1 : n.$

for $j = 1 : n$ do
  Find column $p$ of largest norm
  if $\text{colnrm}[p] > \epsilon$ then
    1. Pivot: swap columns $j$ and $p$ in $A$ and modify $\text{colnrm}$.  
    2. Compute Householder matrix $H_j$ s.t. $H_jA(j : m, j) = \pm ||A(j : m, j)||_2 e_1$.
    3. Update $A(j : m, j + 1 : n) = H_jA(j : m, j + 1 : n)$.  
    4. Norm downdate $\text{colnrm}(j + 1 : n)^2 - = A(j, j + 1 : n)^2$.
  else Break
  end if
end for

If algorithm stops after $k$ steps

$$\sigma_{\text{max}}(R_{22}) \leq \sqrt{n - k} \max_{1 \leq j \leq n - k} ||R_{22}(:, j)||_2 \leq \sqrt{n - k}\epsilon$$
Since

\[
det(R_{11}) = \prod_{i=1}^{k} \sigma_i(R_{11}) = \sqrt{det(A^TA)/\prod_{i=1}^{n-k} \sigma_i(R_{22})}
\]

a strong RRQR is related to a large \(det(R_{11})\). The following algorithm interchanges columns that increase \(det(R_{11})\), given \(f\) and \(k\).

Compute a strong RRQR factorization, given \(k\):

Compute \(A\Pi = QR\) by using QRCP

\[\text{while}\ \text{there exist } i \text{ and } j \text{ such that } \frac{det(\tilde{R}_{11})}{det(R_{11})} > f, \text{ where} \]
\[R_{11} = R(1:k,1:k), \ Pi_{i,j+k} \text{ permutes columns } i \text{ and } j+k, \]
\[R\Pi_{i,j+k} = \tilde{Q}\tilde{R}, \tilde{R}_{11} = \tilde{R}(1:k,1:k) \ do\]
\[\text{Find } i \text{ and } j\]
\[\text{Compute } R\Pi_{i,j+k} = \tilde{Q}\tilde{R} \text{ and } \Pi = \Pi\Pi_{i,j+k}\]

end while
It can be shown that
\[
\frac{\det(\tilde{R}_{11})}{\det(R_{11})} = \sqrt{(R_{11}^{-1} R_{12})_{i,j}^2 + \rho_i^2 (R_{11}) \chi_j^2 (R_{22})}
\] (7)

for any \(1 \leq i \leq k\) and \(1 \leq j \leq n - k\) (the 2-norm of the \(j\)-th column of \(A\) is \(\chi_j(A)\), and the 2-norm of the \(j\)-th row of \(A^{-1}\) is \(\rho_j(A)\)).

Compute a strong RRQR factorization, given \(k\):

Compute \(A\Pi = QR\) by using QRCP

while \(\max_{1 \leq i \leq k, 1 \leq j \leq n - k} \sqrt{(R_{11}^{-1} R_{12})_{i,j}^2 + \rho_i^2 (R_{11}) \chi_j^2 (R_{22})} > f\) do

Find \(i\) and \(j\) such that \(\sqrt{(R_{11}^{-1} R_{12})_{i,j}^2 + \rho_i^2 (R_{11}) \chi_j^2 (R_{22})} > f\)

Compute \(R\Pi_{i,j+k} = \tilde{Q}\tilde{R}\) and \(\Pi = \Pi\Pi_{i,j+k}\)

end while
Strong RRQR (contd)

- $\det(R_{11})$ strictly increases with every permutation, no permutation repeats, hence there is a finite number of permutations to be performed.
Theorem

[Gu and Eisenstat, 1996] If the QR factorization with column pivoting as in equation (7) satisfies inequality

\[
\sqrt{(R_{11}^{-1} R_{12})_{i,j}^2 + \rho_i^2 (R_{11}) \chi_j^2 (R_{22})} < f
\]

for any \(1 \leq i \leq k\) and \(1 \leq j \leq n - k\), then

\[
1 \leq \frac{\sigma_i(A)}{\sigma_i(R_{11})}, \quad \frac{\sigma_j(R_{22})}{\sigma_{k+j}(A)} \leq \sqrt{1 + f^2 k(n - k)},
\]

for any \(1 \leq i \leq k\) and \(1 \leq j \leq \min(m, n) - k\).
Sketch of the proof ([Gu and Eisenstat, 1996])

Assume $A$ is full column rank. Let $\alpha = \sigma_{\text{max}}(R_{22})/\sigma_{\text{min}}(R_{11})$, and let

$$R = \begin{bmatrix} R_{11} & R_{22} \\ \end{bmatrix} \begin{bmatrix} I_k & R_{11}^{-1} R_{12} \\ \alpha l_{n-k} \end{bmatrix} = \tilde{R}_1 W_1.$$ 

We have

$$\sigma_i(R) \leq \sigma_i(\tilde{R}_1) \|W_1\|_2, 1 \leq i \leq n.$$ 

Since $\sigma_{\text{min}}(R_{11}) = \sigma_{\text{max}}(R_{22}/\alpha)$, then $\sigma_i(\tilde{R}_1) = \sigma_i(R_{11})$, for $1 \leq i \leq k$.

$$\|W_1\|_2^2 \leq 1 + \|R_{11}^{-1} R_{12}\|_2^2 + \alpha^2 = 1 + \|R_{11}^{-1} R_{12}\|_2^2 + \|R_{22}\|_2^2 \|R_{11}^{-1}\|_2^2$$

$$\leq 1 + \|R_{11}^{-1} R_{12}\|_F^2 + \|R_{22}\|_F^2 \|R_{11}^{-1}\|_F^2$$

$$= 1 + \sum_{i=1}^{k} \sum_{j=1}^{n-k} ((R_{11}^{-1} R_{12})_{i,j}^2 + \rho_i^2 (R_{11}) \chi_j^2 (R_{22})) \leq 1 + f^2 k(n - k)$$

We obtain,

$$\frac{\sigma_i(A)}{\sigma_i(R_{11})} \leq \sqrt{1 + f^2 k(n - k)}$$
Deterministic column selection: tournament pivoting

1D tournament pivoting (1Dc-TP)

- 1D column block partition of $A$, select $k$ cols from each block with strong RRQR

\[
\begin{pmatrix}
A_{11} \\ \\
\| \\
\downarrow \\
I_{00}
\end{pmatrix}
\begin{pmatrix}
A_{12} \\ \\
\| \\
\downarrow \\
I_{10}
\end{pmatrix}
\begin{pmatrix}
A_{13} \\ \\
\| \\
\downarrow \\
I_{20}
\end{pmatrix}
\begin{pmatrix}
A_{14} \\ \\
\| \\
\downarrow \\
I_{30}
\end{pmatrix}
\begin{pmatrix}
Q_{00}R_{00}P_{c00}^T \\ \\
Q_{10}R_{10}P_{c10}^T \\ \\
Q_{20}R_{20}P_{c20}^T \\ \\
Q_{30}R_{30}P_{c30}^T
\end{pmatrix}
\]

- Reduction tree to select $k$ cols from sets of $2k$ cols,

\[
\begin{pmatrix}
A(:, I_{00} \cup I_{10}) \\ \\
\| \\
\downarrow \\
I_{01}
\end{pmatrix}
\begin{pmatrix}
A(:, I_{20} \cup I_{30}) \\ \\
\| \\
\downarrow \\
I_{11}
\end{pmatrix}
\begin{pmatrix}
Q_{01}R_{01}P_{c01}^T \\ \\
Q_{11}R_{11}P_{c11}^T
\end{pmatrix}
\]

\[A(:, I_{01} \cup I_{11}) = Q_{02}R_{02}P_{c02}^T \rightarrow I_{02}\]

- Return selected columns $A(:, I_{02})$
Deterministic column selection: tournament pivoting

1D tournament pivoting (1Dc-TP)

- 1D column block partition of $A$, select $k$ cols from each block with strong RRQR

\[
\begin{pmatrix}
A_{11} \\
\| \\
Q_{00}R_{00}P_{c_{00}}^T \\
\downarrow \\
l_{00}
\end{pmatrix}
\quad
\begin{pmatrix}
A_{12} \\
\| \\
Q_{10}R_{10}P_{c_{10}}^T \\
\downarrow \\
l_{10}
\end{pmatrix}
\quad
\begin{pmatrix}
A_{13} \\
\| \\
Q_{20}R_{20}P_{c_{20}}^T \\
\downarrow \\
l_{20}
\end{pmatrix}
\quad
\begin{pmatrix}
A_{14} \\
\| \\
Q_{30}R_{30}P_{c_{30}}^T \\
\downarrow \\
l_{30}
\end{pmatrix}
\]

- Reduction tree to select $k$ cols from sets of $2k$ cols,

\[
\begin{pmatrix}
A(:, l_{00} \cup l_{10}) \\
\| \\
Q_{01}R_{01}P_{c_{01}}^T \\
\downarrow \\
l_{01}
\end{pmatrix}
\quad
\begin{pmatrix}
A(:, l_{20} \cup l_{30}); \\
\| \\
Q_{11}R_{11}P_{c_{11}}^T \\
\downarrow \\
l_{11}
\end{pmatrix}
\]

\[A(:, l_{01} \cup l_{11}) = Q_{02}R_{02}P_{c_{02}}^T \rightarrow l_{02}\]

- Return selected columns $A(:, l_{02})$
Deterministic column selection: tournament pivoting

1D tournament pivoting (1Dc-TP)

- 1D column block partition of $A$, select $k$ cols from each block with strong RRQR

$$\begin{pmatrix} A^{11} \\ A^{12} \\ A^{13} \\ A^{14} \end{pmatrix} = \begin{pmatrix} Q_00 & R_00 & P_{c00}^T \\ Q_{10} & R_{10} & P_{c10}^T \\ Q_{20} & R_{20} & P_{c20}^T \\ Q_{30} & R_{30} & P_{c30}^T \end{pmatrix} \downarrow \downarrow \downarrow \downarrow \begin{pmatrix} I_{00} \\ I_{10} \\ I_{20} \\ I_{30} \end{pmatrix}$$

- Reduction tree to select $k$ cols from sets of $2k$ cols,

$$\begin{pmatrix} A(:, I_{00} \cup I_{10}) \\ A(:, I_{20} \cup I_{30}) \end{pmatrix} = \begin{pmatrix} Q_{01} & R_{01} & P_{c01}^T \\ Q_{11} & R_{11} & P_{c11}^T \end{pmatrix} \downarrow \downarrow \begin{pmatrix} I_{01} \\ I_{11} \end{pmatrix}$$

$$A(:, I_{01} \cup I_{11}) = Q_{02} R_{02} P_{c02}^T \rightarrow I_{02}$$

- Return selected columns $A(:, I_{02})$
Deterministic column selection: tournament pivoting

1D tournament pivoting (1Dc-TP)

1D column block partition of $A$, select $k$ cols from each block with strong RRQR

$$
\begin{pmatrix}
A_{11} & A_{12} & A_{13} & A_{14} \\
\| & \| & \| & \\
Q_{00}R_{00}P_{c00}^T & Q_{10}R_{10}P_{c10}^T & Q_{20}R_{20}P_{c20}^T & Q_{30}R_{30}P_{c30}^T
\end{pmatrix}
\begin{pmatrix}
l_00 \\
l_{10} \\
l_{20} \\
l_{30}
\end{pmatrix}
$$

Reduction tree to select $k$ cols from sets of $2k$ cols,

$$
\begin{pmatrix}
A(:, l_{00} \cup l_{10}) \\
\| & \|
\end{pmatrix}
\begin{pmatrix}
A(:, l_{20} \cup l_{30});
\end{pmatrix}
$$

$$
\begin{pmatrix}
Q_{01}R_{01}P_{c01}^T \\
\|
\end{pmatrix}
\begin{pmatrix}
Q_{11}R_{11}P_{c11}^T
\end{pmatrix}
\begin{pmatrix}
l_{01} \\
l_{11}
\end{pmatrix}
$$

$$A(:, l_{01} \cup l_{11}) = Q_{02}R_{02}P_{c02}^T \rightarrow l_{02}$$

Return selected columns $A(:, l_{02})$
Deterministic column selection: tournament pivoting

1D tournament pivoting (1Dc-TP)

- 1D column block partition of $A$, select $k$ cols from each block with strong RRQR

$$
\begin{pmatrix}
A_{11} & A_{12} & A_{13} & A_{14} \\
\| & \| & \| & \\
Q_{00}R_{00}P_{c00}^T & Q_{10}R_{10}P_{c10}^T & Q_{20}R_{20}P_{c20}^T & Q_{30}R_{30}P_{c30}^T \\
\downarrow & \downarrow & \downarrow & \\
I_{00} & I_{10} & I_{20} & I_{30}
\end{pmatrix}
$$

- Reduction tree to select $k$ cols from sets of $2k$ cols,

$$
\begin{pmatrix}
A(:, I_{00} \cup I_{10}) & A(:, I_{20} \cup I_{30}); \\
\| & \|
\end{pmatrix}
\begin{pmatrix}
Q_{01}R_{01}P_{c01}^T & Q_{11}R_{11}P_{c11}^T \\
\downarrow & \downarrow \\
I_{01} & I_{11}
\end{pmatrix}
$$

$$A(:, I_{01} \cup I_{11}) = Q_{02}R_{02}P_{c02}^T \rightarrow I_{02}$$

- Return selected columns $A(:, I_{02})$
Deterministic column selection: tournament pivoting

1D tournament pivoting (1Dc-TP)

- 1D column block partition of $A$, select $k$ cols from each block with strong RRQR:

$$
\begin{pmatrix}
A_{11} & A_{12} & A_{13} & A_{14} \\
\end{pmatrix}
\begin{pmatrix}
Q_{00}R_{00}P_{c00}^T & Q_{10}R_{10}P_{c10}^T & Q_{20}R_{20}P_{c20}^T & Q_{30}R_{30}P_{c30}^T \\
\end{pmatrix}
\begin{pmatrix}
I_{00} & I_{10} & I_{20} & I_{30} \\
\end{pmatrix}
$$

- Reduction tree to select $k$ cols from sets of $2k$ cols:

$$
\begin{pmatrix}
A(:, I_{00} \cup I_{10}) & A(:, I_{20} \cup I_{30}); \\
\end{pmatrix}
\begin{pmatrix}
Q_{01}R_{01}P_{c01}^T & Q_{11}R_{11}P_{c11}^T \\
\end{pmatrix}
\begin{pmatrix}
I_{01} & I_{11} \\
\end{pmatrix}
$$

$$
A(:, I_{01} \cup I_{11}) = Q_{02}R_{02}P_{c02}^T \rightarrow I_{02}
$$

- Return selected columns $A(:, I_{02})$
Select $k$ columns from a tall and skinny matrix

Given $W$ of size $m \times 2k$, $m \gg k$, $k$ columns are selected as:

$$W = QR_{02} \text{ using TSQR}$$

$$R_{02}P_c = Q_2R_2 \text{ using QRCP}$$

Return $WP_c(:,1:k)$
Reduction trees

Any shape of reduction tree can be used during CA\_RRQR, depending on the underlying architecture.

- **Binary tree:**

  \[
  \begin{array}{cccc}
  A_{00} & A_{10} & A_{20} & A_{30} \\
  \downarrow & \downarrow & \downarrow & \downarrow \\
  f(A_{00}) & f(A_{10}) & f(A_{20}) & f(A_{30}) \\
  \downarrow & \downarrow & \downarrow & \downarrow \\
  f(A_{01}) & f(A_{11}) & & \\
  \downarrow & \downarrow & \downarrow & \downarrow \\
  f(A_{02}) & & & \\
  \end{array}
  \]

- **Flat tree:**

  \[
  \begin{array}{cccc}
  A_{00} & A_{10} & A_{20} & A_{30} \\
  \downarrow & \downarrow & \downarrow & \downarrow \\
  f(A_{00}) & f(A_{01}) & f(A_{02}) & f(A_{03}) \\
  \end{array}
  \]

Notation: at each node of the reduction tree, \( f(A_{ij}) \) returns the first \( b \) columns obtained after performing (strong) RRQR of \( A_{ij} \).
It is shown in [Demmel et al., 2015] that the column permutation computed by CA-RRQR satisfies

\[
\chi_j^2 (R_{11}^{-1} R_{12}) + \left(\chi_j (R_{22}) / \sigma_{\min}(R_{11})\right)^2 \leq F_{TP}^2, \text{ for } j = 1, \ldots, n - k. \tag{8}
\]

where \( F_{TP} \) depends on \( k, f, n, \) the shape of reduction tree used during tournament pivoting, and the number of iterations of CARRQR.
Selecting $k$ columns by using tournament pivoting reveals the rank of $A$ with the following bounds:

\[ 1 \leq \frac{\sigma_i(A)}{\sigma_i(R_{11})}, \quad \frac{\sigma_j(R_{22})}{\sigma_{k+j}(A)} \leq \sqrt{1 + F_{TP}^2(n - k)}, \]

\[ \|R_{11}^{-1}R_{12}\|_{max} \leq F_{TP} \]

- Binary tree of depth $\log_2(n/k)$,
  \[ F_{TP} \leq \frac{1}{\sqrt{2k}} (n/k)^{\log_2(\sqrt{2fk})}. \]  
  (9)

  The upper bound is a decreasing function of $k$ when $k > \sqrt{n/(\sqrt{2f})}$.

- Flat tree of depth $n/k$,
  \[ F_{TP} \leq \frac{1}{\sqrt{2k}} \left(\sqrt{2fk}\right)^{n/k}. \]  
  (10)
Tournament pivoting for 1D row partitioning - 1Dr TP

- Row block partition $A$ as e.g.

$$A = \begin{pmatrix} A_{11} \\ A_{21} \\ A_{31} \\ A_{41} \end{pmatrix} = \begin{pmatrix} Q_{00}R_{00}P_{c_{00}^{-1}} \\ Q_{10}R_{10}P_{c_{10}^{-1}} \\ Q_{20}R_{20}P_{c_{20}^{-1}} \\ Q_{30}R_{30}P_{c_{30}^{-1}} \end{pmatrix} \rightarrow \text{select k cols } l_{00}$$

- Apply 1D-TP on sets of 2k sub-columns

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} (:, l_{00} \cup l_{10}) = \begin{pmatrix} Q_{01}R_{01}P_{c_{01}^{-1}} \\ Q_{11}R_{11}P_{c_{11}^{-1}} \end{pmatrix} \rightarrow l_{01}$$

$$\begin{pmatrix} A_{31} \\ A_{41} \end{pmatrix} (:, l_{20} \cup l_{30}) = \begin{pmatrix} Q_{02}R_{02}P_{c_{02}^{-1}} \\ \end{pmatrix} \rightarrow l_{02}$$

- Return columns $A(:, l_{02})$
Tournament pivoting for 1D row partitioning - 1Dr TP

- Row block partition $A$ as e.g.

$$
A = \begin{pmatrix}
A_{11} \\
A_{21} \\
A_{31} \\
A_{41}
\end{pmatrix} = 
\begin{pmatrix}
Q_{00} R_{00} P_{c_{00}^{-1}} \\
Q_{10} R_{10} P_{c_{10}^{-1}} \\
Q_{20} R_{20} P_{c_{20}^{-1}} \\
Q_{30} R_{30} P_{c_{30}^{-1}}
\end{pmatrix} \rightarrow \text{select k cols } I_{00} \\
\text{select k cols } I_{10} \\
\text{select k cols } I_{20} \\
\text{select k cols } I_{30}
$$

- Apply 1D-TP on sets of 2k sub-columns

$$
\begin{pmatrix}
(A_{11})(:, I_{00} \cup I_{10}) \\
(A_{21})(:, I_{20} \cup I_{30})
\end{pmatrix} = 
\begin{pmatrix}
(Q_{01} R_{01} P_{c_{01}^{-1}}) \\
(Q_{11} R_{11} P_{c_{11}^{-1}})
\end{pmatrix} \rightarrow I_{01}
$$

- Return columns $A(:, I_{02})$
Row block partition $A$ as e.g.

$$
A = \begin{pmatrix}
A_{11} \\
A_{21} \\
A_{31} \\
A_{41}
\end{pmatrix}
= \begin{pmatrix}
Q_{00} R_{00} P_{c_{00}}^{-1} \\
Q_{10} R_{10} P_{c_{10}}^{-1} \\
Q_{20} R_{20} P_{c_{20}}^{-1} \\
Q_{30} R_{30} P_{c_{30}}^{-1}
\end{pmatrix}
\rightarrow \text{select k cols } l_{00}
\rightarrow \text{select k cols } l_{10}
\rightarrow \text{select k cols } l_{20}
\rightarrow \text{select k cols } l_{30}
$$

Apply 1D-TP on sets of $2k$ sub-columns

$$
\begin{pmatrix}
(A_{11})_{(:, l_{00} \cup l_{10})} \\
(A_{21})_{(:, l_{00} \cup l_{10})} \\
(A_{31})_{(:, l_{20} \cup l_{30})} \\
(A_{41})_{(:, l_{20} \cup l_{30})}
\end{pmatrix}
= \begin{pmatrix}
Q_{01} R_{01} P_{c_{01}}^{-1} \\
Q_{11} R_{11} P_{c_{11}}^{-1}
\end{pmatrix}
\rightarrow l_{01}
\rightarrow l_{11}
$$

$$
A(:, l_{01} \cup l_{11}) = (Q_{02} R_{02} P_{c_{02}}^{-1}) \rightarrow l_{02}
$$

Return columns $A(:, l_{02})$
Tournament pivoting for 1D row partitioning - 1Dr TP

- **Row block partition** $A$ as e.g.

  $$A = \begin{pmatrix} A_{11} \\ A_{21} \\ A_{31} \\ A_{41} \end{pmatrix} = \begin{pmatrix} Q_{00}R_{00}P_{c_{00}}^{-1} \\ Q_{10}R_{10}P_{c_{10}}^{-1} \\ Q_{20}R_{20}P_{c_{20}}^{-1} \\ Q_{30}R_{30}P_{c_{30}}^{-1} \end{pmatrix} \rightarrow \text{select k cols } l_{00} \rightarrow \text{select k cols } l_{10} \rightarrow \text{select k cols } l_{20} \rightarrow \text{select k cols } l_{30}$$

- Apply 1D-TP on sets of $2k$ sub-columns

  $$\begin{pmatrix} A_{11} \\ A_{21} \\ A_{31} \\ A_{41} \end{pmatrix} \begin{pmatrix} \cdot, l_{00} \cup l_{10} \\ \cdot, l_{20} \cup l_{30} \end{pmatrix} = \begin{pmatrix} Q_{01}R_{01}P_{c_{01}}^{-1} \\ Q_{11}R_{11}P_{c_{11}}^{-1} \end{pmatrix} \rightarrow l_{01} \rightarrow l_{11}$$

  $$\Rightarrow A(\cdot, l_{01} \cup l_{11}) = (Q_{02}R_{02}P_{c_{02}}^{-1}) \rightarrow l_{02}$$

- **Return columns** $A(\cdot, l_{02})$
Row block partition $A$ as e.g.

$$A = \begin{pmatrix} A_{11} \\ A_{21} \\ A_{31} \\ A_{41} \end{pmatrix} = \begin{pmatrix} Q_{00} R_{00} P_{c_{00}}^{-1} \\ Q_{10} R_{10} P_{c_{10}}^{-1} \\ Q_{20} R_{20} P_{c_{20}}^{-1} \\ Q_{30} R_{30} P_{c_{30}}^{-1} \end{pmatrix}$$

- select k cols $I_{00}$
- select k cols $I_{10}$
- select k cols $I_{20}$
- select k cols $I_{30}$

Apply 1D-TP on sets of $2k$ sub-columns

$$\begin{pmatrix} (A_{11}) (::, I_{00} \cup I_{10}) \\ (A_{21}) (::, I_{20} \cup I_{30}) \end{pmatrix} = \begin{pmatrix} Q_{01} R_{01} P_{c_{01}}^{-1} \\ Q_{11} R_{11} P_{c_{11}}^{-1} \end{pmatrix} \rightarrow I_{01}$$

- $I_{01}$

$$A(::, I_{01} \cup I_{11}) = (Q_{02} R_{02} P_{c_{02}}^{-1}) \rightarrow I_{02}$$

Return columns $A(::, I_{02})$
CA-RRQR : 2D tournament pivoting

- $A$ distributed on $P_r \times P_c$ procs as e.g.

\[
A = \begin{pmatrix}
A_{11} & A_{12} & A_{13} & A_{14} \\
A_{21} & A_{22} & A_{23} & A_{24}
\end{pmatrix}
\]

- Select $k$ cols from each column block by 1Dr-TP,

\[
\begin{pmatrix}
A_{11} \\
A_{21}
\end{pmatrix}, \begin{pmatrix}
A_{12} \\
A_{22}
\end{pmatrix}, \begin{pmatrix}
A_{13} \\
A_{23}
\end{pmatrix}, \begin{pmatrix}
A_{14} \\
A_{24}
\end{pmatrix}
\]

\[
\downarrow \downarrow \downarrow \downarrow
\]

$I_{00}$ $I_{10}$ $I_{20}$ $I_{30}$

- Apply 1Dc-TP on sets of $k$ selected cols,

\[
A(:, I_{00}), A(:, I_{10}), A(:, I_{20}), A(:, I_{30})
\]

- Return columns selected by 1Dc-TP $A(:, I_{02})$
CA-RRQR : 2D tournament pivoting

- $A$ distributed on $P_r \times P_c$ procs as e.g.

$$A = \begin{pmatrix}
A_{11} & A_{12} & A_{13} & A_{14} \\
A_{21} & A_{22} & A_{23} & A_{24}
\end{pmatrix}$$

- Select $k$ cols from each column block by 1Dr-TP,

$$\begin{pmatrix}
A_{11} \\
A_{21}
\end{pmatrix} \downarrow \quad \begin{pmatrix}
A_{12} \\
A_{22}
\end{pmatrix} \downarrow \quad \begin{pmatrix}
A_{13} \\
A_{23}
\end{pmatrix} \downarrow \quad \begin{pmatrix}
A_{14} \\
A_{24}
\end{pmatrix}$$

$$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$$

$$l_{00} \quad l_{10} \quad l_{20} \quad l_{30}$$

- Apply 1Dc-TP on sets of $k$ selected cols,

$$A(:, l_{00}) \quad A(:, l_{10}) \quad A(:, l_{20}) \quad A(:, l_{30})$$

- Return columns selected by 1Dc-TP $A(:, l_{02})$
CA-RRQR : 2D tournament pivoting

- $A$ distributed on $P_r \times P_c$ procs as e.g.

$$A = \begin{pmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \end{pmatrix}$$

- Select $k$ cols from each column block by 1Dr-TP,

$$\begin{align*}
&\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} \quad \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} \quad \begin{pmatrix} A_{13} \\ A_{23} \end{pmatrix} \quad \begin{pmatrix} A_{14} \\ A_{24} \end{pmatrix} \\
&\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
&l_{00} \quad l_{10} \quad l_{20} \quad l_{30}
\end{align*}$$

- Apply 1Dc-TP on sets of $k$ selected cols,

$$\begin{align*}
&\begin{pmatrix} A(:, l_{00}) \\ A(:, l_{10}) \\ A(:, l_{20}) \\ A(:, l_{30}) \end{pmatrix}
\end{align*}$$

- Return columns selected by 1Dc-TP $A(:, l_{02})$
Numerical results

- Stability close to QRCP for many tested matrices.
- Absolute value of diagonals of R referred to as R-values.

Methods compared

- RRQR: QR with column pivoting
- CA-RRQR-B with tournament pivoting 1Dc-TP based on binary tree
- CA-RRQR-F with tournament pivoting 1Dc-TP based on flat tree
- SVD
Numerical results (contd)

- Left: exponent - exponential Distribution, $\sigma_1 = 1$, $\sigma_i = \alpha^{i-1}$ ($i = 2, \ldots, n$), $\alpha = 10^{-1/11}$ [Bischof, 1991]
- Right: shaw - 1D image restoration model [Hansen, 2007]

\[
\epsilon \min\{\|A\Pi_0(:, i)\|_2, \|A\Pi_1(:, i)\|_2, \|A\Pi_2(:, i)\|_2\} \tag{11}
\]
\[
\epsilon \max\{\|A\Pi_0(:, i)\|_2, \|A\Pi_1(:, i)\|_2, \|A\Pi_2(:, i)\|_2\} \tag{12}
\]

where $\Pi_j (j = 0, 1, 2)$ are the permutation matrices obtained by QRCP, CARRQR-B, and CARRQR-F, and $\epsilon$ is the machine precision.
CA-RRQR : 2D tournament pivoting
Numerical experiments

Original image, size 1190 × 1920

Rank-10 approx, 2D TP 8 × 8 procs

Rank-50 approx, 2D TP 8 × 8 procs

Singular values and ratios

Image source: https://pixabay.com/photos/billiards-ball-play-number-half-4345870/
Numerical results - a set of 18 matrices

- Ratios $|R(i, i)|/\sigma_i(R)$, for QRCP (top plot), CARRQR-B (second plot), and CARRQR-F (third plot).
- The number along x-axis represents the index of test matrices.
Plan

Low rank matrix approximation

Low rank approximation based on max-vol

Rank revealing QR factorization

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP
LU versus QR - filled graph $G^+(A)$

- Consider $A$ is SPD and $A = LL^T$
- Given $G(A) = (V, E)$, $G^+(A) = (V, E^+)$ is defined as: there is an edge $(i, j) \in G^+(A)$ iff there is a path from $i$ to $j$ in $G(A)$ going through lower numbered vertices.
- $G(L + L^T) = G^+(A)$, ignoring cancellations.
- Definition holds also for directed graphs (LU factorization).

$$A = \begin{pmatrix}
    1 & x & x & x & x \\
    x & x & x & x \\
    x & x & x & x & x \\
    x & x & x & x & x \\
    x & x & x & x & x & x \\
    x & x & x & x & x & x & x \\
    x & x & x & x & x & x & x & x \\
    x & x & x & x & x & x & x & x & x \\
\end{pmatrix}$$

$$L + L^T = \begin{pmatrix}
    1 & x & x & x & x & x & x & x & x \\
    x & x & x & x & x & x & x \\
    x & x & x & x & x & x & x & x \\
    x & x & x & x & x & x & x & x \\
    x & x & x & x & x & x & x & x & x \\
\end{pmatrix}$$
LU versus QR

Filled column intersection graph $G^+_n(A)$

- Graph of the Cholesky factor of $A^T A$
- $G(R) \subseteq G^+_n(A)$
- $A^T A$ can have many more nonzeros than $A$
LU versus QR

Numerical stability

Let $\hat{L}$ and $\hat{U}$ be the computed factors of the block LU factorization. Then

$$\hat{L}\hat{U} = A + E, \quad \|E\|_{\text{max}} \leq c(n)\varepsilon \left(\|A\|_{\text{max}} + \|\hat{L}\|_{\text{max}}\|\hat{U}\|_{\text{max}}\right). \quad (13)$$

For partial pivoting, $\|L\|_{\text{max}} \leq 1, \|U\|_{\text{max}} \leq 2^n\|A\|_{\text{max}}$

In practice, $\|U\|_{\text{max}} \leq \sqrt{n}\|A\|_{\text{max}}$
Low rank approximation based on LU factorization

- Given desired rank $k$, the factorization has the form

$$P_r A P_c = \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{pmatrix} = \begin{pmatrix} I & \bar{A}_{21} \bar{A}_{11}^{-1} \\ \bar{A}_{21} \bar{A}_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ S(\bar{A}_{11}) \end{pmatrix},$$

(14)

where $A \in \mathbb{R}^{m \times n}$, $\bar{A}_{11} \in \mathbb{R}^{k \times k}$, $S(\bar{A}_{11}) = \bar{A}_{22} - \bar{A}_{21} \bar{A}_{11}^{-1} \bar{A}_{12}$.

- The rank-$k$ approximation matrix $\bar{A}_k$ is

$$\bar{A}_k = \begin{pmatrix} I \\ \bar{A}_{21} \bar{A}_{11}^{-1} \end{pmatrix} \bar{A}_{11} \bar{A}_{12} = \begin{pmatrix} \bar{A}_{11} \\ \bar{A}_{21} \end{pmatrix} \bar{A}_{11}^{-1} \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \end{pmatrix}. \quad (15)$$

- $\bar{A}_{11}^{-1}$ is never formed, its factorization is used when $\bar{A}_k$ is applied to a vector.
Design space

Non-exhaustive list for selecting $k$ columns and rows:

1. Select $k$ linearly independent columns of $A$ (call result $B$), by using
   1.1 (strong) QRCP/tournament pivoting using QR,
   1.2 LU / tournament pivoting based on LU, with some form of pivoting (column, complete, rook),
   1.3 randomization: premultiply $X = ZA$ where random matrix $Z$ is short and fat, then pick $k$ rows from $X^T$, by some method from 2) below,
   1.4 tournament pivoting based on randomized algorithms to select columns at each step.

2. Select $k$ linearly independent rows of $B$, by using
   2.1 (strong) QRCP / tournament pivoting based on QR on $B^T$, or on $Q^T$, the rows of the thin $Q$ factor of $B$,
   2.2 LU / tournament pivoting based on LU, with pivoting (row, complete, rook) on $B$,
   2.3 tournament pivoting based on randomized algorithms to select rows.
Select $k$ cols using tournament pivoting

- Partition $A = (A_1, A_2, A_3, A_4)$.
- Select $k$ cols from each column block, by using QR with column pivoting.
- At each level $i$ of the tree
  - At each node $j$ do in parallel
    - Let $A_{v,i-1}, A_{w,i-1}$ be the cols selected by the children of node $j$
    - Select $k$ cols from $(A_{v,i-1}, A_{w,i-1})$, by using QR with column pivoting
- Return columns in $A_{ji}$
Select $k$ cols using tournament pivoting

- Partition $A = (A_1, A_2, A_3, A_4)$.
- Select $k$ cols from each column block, by using QR with column pivoting.
- At each level $i$ of the tree:
  - At each node $j$ do in parallel
    - Let $A_{v,i-1}, A_{w,i-1}$ be the cols selected by the children of node $j$
    - Select $k$ cols from $(A_{v,i-1}, A_{w,i-1})$, by using QR with column pivoting
- Return columns in $A_{ji}$"
Select $k$ cols using tournament pivoting

- Partition $A = (A_1, A_2, A_3, A_4)$.
- Select $k$ cols from each column block, by using QR with column pivoting.
- At each level $i$ of the tree
  - At each node $j$ do in parallel
    - Let $A_{v,i-1}, A_{w,i-1}$ be the cols selected by the children of node $j$
    - Select $k$ cols from $(A_{v,i-1}, A_{w,i-1})$, by using QR with column pivoting
- Return columns in $A_{ji}$
Select $k$ cols using tournament pivoting

- Partition $A = (A_1, A_2, A_3, A_4)$.
- Select $k$ cols from each column block, by using QR with column pivoting.
- At each level $i$ of the tree
  - At each node $j$ do in parallel
    - Let $A_{v,i-1}, A_{w,i-1}$ be the cols selected by the children of node $j$
    - Select $k$ cols from $(A_{v,i-1}, A_{w,i-1})$, by using QR with column pivoting
- Return columns in $A_{ji}$
Select $k$ cols using tournament pivoting

- Partition $A = (A_1, A_2, A_3, A_4)$.
- Select $k$ cols from each column block, by using QR with column pivoting.
- At each level $i$ of the tree:
  - At each node $j$ do in parallel:
    - Let $A_{v,i-1}, A_{w,i-1}$ be the cols selected by the children of node $j$.
    - Select $k$ cols from $(A_{v,i-1}, A_{w,i-1})$, by using QR with column pivoting.
- Return columns in $A_{j,i}$. 
**Select $k$ cols using tournament pivoting**

- Partition $A = (A_1, A_2, A_3, A_4)$.
- Select $k$ cols from each column block, by using QR with column pivoting.
- At each level $i$ of the tree:
  - At each node $j$ do in parallel:
    - Let $A_{v,i-1}, A_{w,i-1}$ be the cols selected by the children of node $j$.
    - Select $k$ cols from $(A_{v,i-1}, A_{w,i-1})$, by using QR with column pivoting.
- Return columns in $A_{ji}$.
LU_CRTP factorization - one block step

One step of truncated block LU based on column/row tournament pivoting on matrix $A$ of size $m \times n$:

1. Select $k$ columns by using tournament pivoting, permute them in front, bounds for s.v. governed by $q_1(n, k)$

   $$AP_c = Q \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix}$$

2. Select $k$ rows from $(Q_{11}; Q_{21})^T$ of size $m \times k$ by using tournament pivoting,

   $$P_r Q = \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ \bar{Q}_{21} & \bar{Q}_{22} \end{pmatrix}$$

   such that $\|\bar{Q}_{21} \bar{Q}_{11}^{-1}\|_{max} \leq F_{TP}$ and bounds for s.v. governed by $q_2(m, k)$. 
Orthogonal matrices

Given orthogonal matrix $Q \in \mathbb{R}^{m \times m}$ and its partitioning

$$Q = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix}, \quad (16)$$

the selection of $k$ cols by tournament pivoting from $(Q_{11}; Q_{21})^T$ leads to the factorization

$$P_r Q = \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ \bar{Q}_{21} & \bar{Q}_{22} \end{pmatrix} = \begin{pmatrix} I & \bar{Q}_{21} Q_{11}^{-1} \\ \bar{Q}_{21} & \bar{Q}_{22} \end{pmatrix} \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ S(\bar{Q}_{11}) \end{pmatrix} \quad (17)$$

where $S(\bar{Q}_{11}) = \bar{Q}_{22} - \bar{Q}_{21} \bar{Q}_{11}^{-1} \bar{Q}_{12} = \bar{Q}_{22}^{-T}$ since

$$S(\bar{Q}_{11}) \bar{Q}_{22}^T = \bar{Q}_{22} \bar{Q}_{22}^T - \bar{Q}_{21} \bar{Q}_{11}^{-1} \bar{Q}_{12} \bar{Q}_{22}^T = I - \bar{Q}_{21} \bar{Q}_{21}^T - \bar{Q}_{21} \bar{Q}_{11}^{-1} \bar{Q}_{12} \bar{Q}_{22}^T$$

$$= I - \bar{Q}_{21} (\bar{Q}_{21}^T - \bar{Q}_{11}^{-1} \bar{Q}_{11} \bar{Q}_{21}^T) = I$$
Orthogonal matrices (contd)

The factorization
\[
P_r Q = \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ \bar{Q}_{21} & \bar{Q}_{22} \end{pmatrix} = \begin{pmatrix} I & \bar{Q}_{21} \bar{Q}_{11}^{-1} \\ \bar{Q}_{21} & I \end{pmatrix} \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ S(\bar{Q}_{11}) \end{pmatrix}
\] (18)

satisfies:
\[
\rho_j(\bar{Q}_{21} \bar{Q}_{11}^{-1}) \leq F_{TP},
\]
(19)
\[
\frac{1}{q_2(m,k)} \leq \sigma_i(\bar{Q}_{11}) \leq 1,
\]
(20)
\[
\sigma_{\text{min}}(\bar{Q}_{11}) = \sigma_{\text{min}}(\bar{Q}_{22})
\]
(21)

for all \(1 \leq i \leq k, 1 \leq j \leq m - k\), where \(\rho_j(A)\) is the 2-norm of the j-th row of \(A\), \(q_2(m,k) = \sqrt{1 + F_{TP}^2(m - k)}\).

Exercice: show that \(\sigma_{\text{min}}(\bar{Q}_{11}) = \sigma_{\text{min}}(\bar{Q}_{22})\) by considering unit vectors \(x \in \mathbb{R}^k, y \in \mathbb{R}^{m-k}\)
\[
1 = \|\bar{Q}_{11}x\|^2 + \|\bar{Q}_{21}x\|^2, 
1 = \|\bar{Q}^T_{22}y\|^2 + \|\bar{Q}^T_{21}y\|^2
\]

and showing \(\min_{\|x\|=1} \|\bar{Q}_{11}x\|^2 = \min_{\|y\|=1} \|\bar{Q}^T_{22}y\|^2\)
Sketch of the proof

\[ P_r A P_c = \left( \begin{array}{cc} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{array} \right) = \left( \begin{array}{cc} I & \bar{A}_{21} \bar{A}_{11}^{-1} \\ \bar{Q}_{21} \bar{Q}_{11}^{-1} & I \end{array} \right) \left( \begin{array}{cc} \bar{A}_{11} & \bar{A}_{12} \\ S(\bar{A}_{11}) \end{array} \right) \left( \begin{array}{cc} R_{11} & R_{12} \\ R_{21} & R_{22} \end{array} \right) \]

where

\[ \bar{Q}_{21} \bar{Q}_{11}^{-1} = \bar{A}_{21} \bar{A}_{11}^{-1}, \]

\[ S(\bar{A}_{11}) = S(\bar{Q}_{11}) R_{22} = \bar{Q}_{22}^T R_{22}. \]
Sketch of the proof (contd)

\[ \bar{A}_{11} = \bar{Q}_{11}R_{11}, \quad \text{(23)} \]
\[ S(\bar{A}_{11}) = S(\bar{Q}_{11})R_{22} = \bar{Q}_{22}^{-T}R_{22}. \quad \text{(24)} \]

We obtain

\[ \sigma_i(A) \geq \sigma_i(\bar{A}_{11}) \geq \sigma_{\min}(\bar{Q}_{11})\sigma_i(R_{11}) \geq \frac{1}{q_1(n, k)q_2(m, k)}\sigma_i(A), \]

We also have that

\[ \sigma_{k+j}(A) \leq \sigma_j(S(\bar{A}_{11})) = \sigma_j(S(\bar{Q}_{11})R_{22}) \leq ||S(\bar{Q}_{11})||_2\sigma_j(R_{22}) \leq q_1(n, k)q_2(m, k)\sigma_{k+j}(A), \]

where \( q_1(n, k) = \sqrt{1 + F_{TP}^2(n - k)} \), \( q_2(m, k) = \sqrt{1 + F_{TP}^2(m - k)} \).
LU_CRTP factorization - bounds if \(\text{rank} = k\)

Given \(A\) of size \(m \times n\), one step of LU_CRTP computes the decomposition

\[
\tilde{A} = P_r A P_c = \begin{pmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ \tilde{A}_{21} & \tilde{A}_{22} \end{pmatrix} = \begin{pmatrix} I & \tilde{Q}_{21} \tilde{Q}_{11}^{-1} \\ \tilde{Q}_{21} \tilde{Q}_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ S(\tilde{A}_{11}) \end{pmatrix}
\]

(25)

where \(\tilde{A}_{11}\) is of size \(k \times k\) and

\[
S(\tilde{A}_{11}) = \tilde{A}_{22} - \tilde{A}_{21} \tilde{A}_{11}^{-1} \tilde{A}_{12} = \tilde{A}_{22} - \tilde{Q}_{21} \tilde{Q}_{11}^{-1} \tilde{A}_{12}.
\]

(26)

It satisfies the following properties:

\[
\rho_l(\tilde{A}_{21} \tilde{A}_{11}^{-1}) = \rho_l(\tilde{Q}_{21} \tilde{Q}_{11}^{-1}) \leq F_{TP},
\]

(27)

\[
\|S(\tilde{A}_{11})\|_{max} \leq \min((1 + F_{TP} \sqrt{k})\|A\|_{max}, F_{TP} \sqrt{1 + F_{TP}^2 (m - k) \sigma_k(A)})
\]

\[
1 \leq \frac{\sigma_i(A)}{\sigma_i(\tilde{A}_{11})}, \frac{\sigma_j(S(\tilde{A}_{11}))}{\sigma_{k+j}(A)} \leq q(m, n, k),
\]

(28)

for any \(1 \leq l \leq m - k\), \(1 \leq i \leq k\), and \(1 \leq j \leq \min(m, n) - k\),

\[
q(m, n, k) = q_1(n, k)q_2(m, k) = \sqrt{(1 + F_{TP}^2 (n - k))(1 + F_{TP}^2 (m - k))}.
\]
Details on the pivot growth

First bound: $\rho_l(\bar{A}_{21}\bar{A}_{11}^{-1}) = \rho_l(\bar{Q}_{21}\bar{Q}_{11}^{-1}) \leq F_{TP}$, for each row $l$ of $\bar{A}_{21}\bar{A}_{11}^{-1}$.

Element growth in $S(\bar{A}_{11})$ is bounded as follows.

$$|S(\bar{A}_{11})(i,j)| = |\bar{A}_{22}(i,j) - (\bar{A}_{21}\bar{A}_{11}^{-1})(i,:)\bar{A}_{12}(:,j)|$$

$$\leq \|A\|_{\text{max}} + \|(\bar{A}_{21}\bar{A}_{11}^{-1})(i,:)\|_2\|\bar{A}_{12}(:,j)\|_2$$

$$\leq \|A\|_{\text{max}} + \rho_i(\bar{A}_{21}\bar{A}_{11}^{-1})\sqrt{k}\|A\|_{\text{max}}$$

$$\leq (1 + F_{TP}\sqrt{k})\|A\|_{\text{max}}$$

Second bound: $\chi_j(R_{22}) = \|R_{22}(:,j)\|_2 \leq F_{TP}\sigma_{\text{min}}(R_{11}) \leq F_{TP}\sigma_k(A)$. The absolute value of an element of $S(\bar{A}_{11})$ can be bounded as follows,

$$|S(\bar{A}_{11})(i,j)| = |\bar{Q}_{22}^{-T}(i,:)R_{22}(:,j)| \leq \|\bar{Q}_{22}^{-1}(:,i)\|_2\|R_{22}(:,j)\|_2$$

$$\leq \|\bar{Q}_{22}^{-1}\|_2\|R_{22}(:,j)\|_2 = \|R_{22}(:,j)\|_2/\sigma_{\text{min}}(\bar{Q}_{22})$$

$$\leq q_2(m,k)F_{TP}\sigma_k(A).$$

Hence:

$$\|S(\bar{A}_{11})\|_{\text{max}} \leq \min((1 + F_{TP}\sqrt{k})\|A\|_{\text{max}}, F_{TP}\sqrt{1 + F_{TP}^2(m - k)\sigma_k(A)})$$
Plan

Low rank matrix approximation

Low rank approximation based on max-vol

Rank revealing QR factorization

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP
Numerical results

- **Left**: exponent - exponential Distribution, \( \sigma_1 = 1, \sigma_i = \alpha^{i-1} \) \((i = 2, \ldots, n)\), \(\alpha = 10^{-1/11} \) [Bischof, 1991]

- **Right**: foxgood - Severely ill-posed test problem of the 1st kind Fredholm integral equation used by Fox and Goodwin
Here $k = 16$ and the factorization is truncated at $K = 128$ (bars) or $K = 240$ (red lines).

- **LU\_CTP**: Column tournament pivoting + partial pivoting
- All singular values smaller than machine precision, $\epsilon$, are replaced by $\epsilon$.
- The number along $x$-axis represents the index of test matrices.
Results for image of size $919 \times 707$
Results for image of size 691 × 505

Original image

Rank-105 approx, SVD

Rank-105 approx, LUPP

Rank-105 approx, LU_CRTP

Rank-209 approx, LU_CRTP

Singular value distribution
Comparing nnz in the factors $L, U$ versus $Q, R$

<table>
<thead>
<tr>
<th>Name/size</th>
<th>Nnz $A(:, 1:K)$</th>
<th>Rank K</th>
<th>Nnz QRCP/Nnz LU_CRTP</th>
<th>Nnz LU_CRTP/Nnz LUPP</th>
</tr>
</thead>
<tbody>
<tr>
<td>gemat11</td>
<td>1232</td>
<td>128</td>
<td>2.1</td>
<td>2.2</td>
</tr>
<tr>
<td>4929</td>
<td>4895</td>
<td>512</td>
<td>3.3</td>
<td>2.6</td>
</tr>
<tr>
<td></td>
<td>9583</td>
<td>1024</td>
<td>11.5</td>
<td>3.2</td>
</tr>
<tr>
<td>wang3</td>
<td>896</td>
<td>128</td>
<td>3.0</td>
<td>2.1</td>
</tr>
<tr>
<td>26064</td>
<td>3536</td>
<td>512</td>
<td>2.9</td>
<td>2.1</td>
</tr>
<tr>
<td></td>
<td>7120</td>
<td>1024</td>
<td>2.9</td>
<td>1.2</td>
</tr>
<tr>
<td>Rfdevice</td>
<td>633</td>
<td>128</td>
<td>10.0</td>
<td>1.1</td>
</tr>
<tr>
<td>74104</td>
<td>2255</td>
<td>512</td>
<td>82.6</td>
<td>0.9</td>
</tr>
<tr>
<td></td>
<td>4681</td>
<td>1024</td>
<td>207.2</td>
<td>0.0</td>
</tr>
<tr>
<td>Parab_fem</td>
<td>896</td>
<td>128</td>
<td>–</td>
<td>0.5</td>
</tr>
<tr>
<td>525825</td>
<td>3584</td>
<td>512</td>
<td>–</td>
<td>0.3</td>
</tr>
<tr>
<td></td>
<td>7168</td>
<td>1024</td>
<td>–</td>
<td>0.2</td>
</tr>
<tr>
<td>Mac_econ</td>
<td>384</td>
<td>128</td>
<td>–</td>
<td>0.3</td>
</tr>
<tr>
<td>206500</td>
<td>1535</td>
<td>512</td>
<td>–</td>
<td>0.3</td>
</tr>
<tr>
<td></td>
<td>5970</td>
<td>1024</td>
<td>–</td>
<td>0.2</td>
</tr>
</tbody>
</table>
Performance results

Selection of 256 columns by tournament pivoting

- Edison, Cray XC30 (NERSC): 2x12-core Intel Ivy Bridge (2.4 GHz)
- Tournament pivoting uses SPQR (T. Davis) + dGEQP3 (Lapack), time in secs

Matrices: dimension at leaves on 32 procs

- Parab_fem: 528825 × 528825 528825 × 16432
- Mac_econ: 206500 × 206500 206500 × 6453

<table>
<thead>
<tr>
<th></th>
<th>Time 2k cols</th>
<th>Time leaves 32procs</th>
<th></th>
<th>Number of MPI processes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SPQR + dGEQP3</td>
<td></td>
<td></td>
<td>16</td>
</tr>
<tr>
<td>Parab_fem</td>
<td>0.26</td>
<td>0.26 + 1129</td>
<td></td>
<td>46.7</td>
</tr>
<tr>
<td>Mac_econ</td>
<td>0.46</td>
<td>25.4 + 510</td>
<td></td>
<td>132.7</td>
</tr>
</tbody>
</table>

60 of 64
More details on CA deterministic algorithms


- Low rank approximation of a sparse matrix based on LU factorization with column and row tournament pivoting, with S. Cayrols and J. Demmel, Inria TR 8910.
A parallel QR factorization algorithm with controlled local pivoting.

Linear least squares solutions by Householder transformations.

Exponential inapproximability of selecting a maximum volume sub-matrix.

Communication-avoiding rank-revealing qr decomposition.

Eckart, C. and Young, G. (1936).
The approximation of one matrix by another of lower rank.
*Psychometrika*, 1:211–218.

Relative perturbation techniques for singular value problems.

The maximal-volume concept in approximation by low-rank matrices.
Efficient algorithms for computing a strong rank-revealing QR factorization.

Regularization tools: A matlab package for analysis and solution of discrete ill-posed problems.
Results used in the proofs

  Let \( A = [a_1 \ldots | a_n] \) be a column partitioning of an \( m \times n \) matrix with \( m \geq n \). If \( A_r = [a_1 \ldots | a_r] \), then for \( r = 1 : n - 1 \)

\[
\sigma_1(A_{r+1}) \geq \sigma_1(A_r) \geq \sigma_2(A_{r+1}) \geq \ldots \geq \sigma_r(A_{r+1}) \geq \sigma_r(A_r) \geq \sigma_{r+1}(A_{r+1}).
\]

- Given \( n \times n \) matrix \( B \) and \( n \times k \) matrix \( C \), then ([Eisenstat and Ipsen, 1995], p. 1977)

\[
\sigma_{\min}(B)\sigma_j(C) \leq \sigma_j(BC) \leq \sigma_{\max}(B)\sigma_j(C), j = 1, \ldots, k.
\]