Rank revealing factorizations, and low rank approximations

L. Grigori

Inria Paris, Sorbonne University

December 2020

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

Plan

Low rank matrix approximation

Low rank approximation based on max-vol

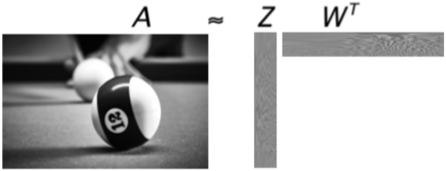
Rank revealing QR factorization

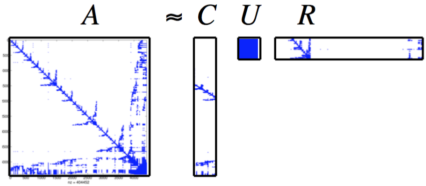
LU_CRTP: Truncated LU factorization with column and row tournament pivoting

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, ...

$$Ax \rightarrow ZW^T x$$

$$\text{Flops} \quad 2mn \rightarrow 2(m+n)k$$

Singular value decomposition

For any given $A \in \mathbb{R}^{m \times n}$, $m \geq n$ its singular value decomposition is

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

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 $\sigma_1(A) \geq \ldots \geq \sigma_n(A) \geq 0$ Σ_1 is $k \times k$, Σ_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
  $Range(A) = span(U_1)$, $Null(A) = span(V_2)$,
  $Range(A^T) = span(V_1)$, $Null(A) = span(U_2\ U_3)$.

Norms

$$||A||_p = \max_{||x||_p=1} ||Ax||_p$$

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

$$||A||_2 = \sigma_{max}(A) = \sigma_1(A)$$

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]

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

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

Image, size 1190 × 1920 | Rank-10 approximation, SVD | Rank-50 approximation, SVD

- Image source: `https://pixabay.com/photos/billiards-ball-play-number-half-4345870/`

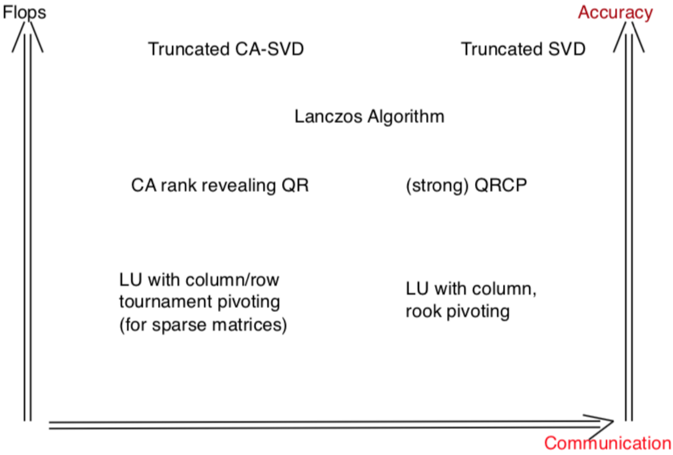
Large data sets

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.

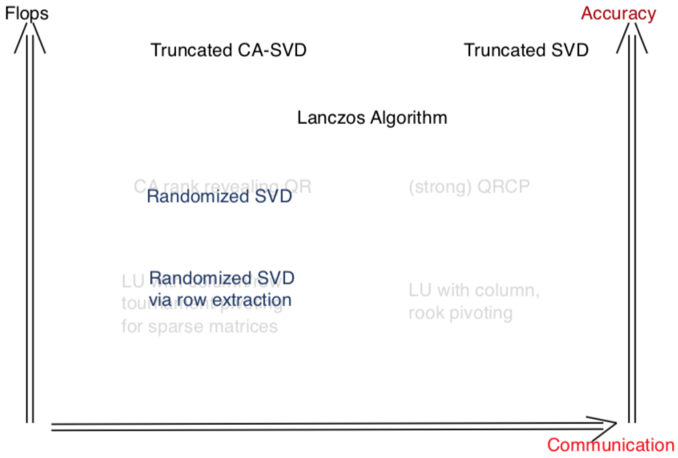
Low rank matrix approximation: trade-offs



Communication optimal if computing a rank-k approximation on P processors requires

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

Low rank matrix approximation: trade-offs



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

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

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

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

$$\mathcal{P}^o = A\Omega_1 (A\Omega_1)^+ = Q_1 Q_1^T, \text{ or equiv } \mathcal{P}^o a_j := arg \min_{x \in X} \|x - a_j\|_2$$

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

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

Idea underlying many algorithms

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

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

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

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

   $$\mathcal{P}^o = A\Omega_1 (A\Omega_1)^+ = Q_1 Q_1^T, \text{ or equiv } \mathcal{P}^o a_j := arg \min_{x \in X} \|x - a_j\|_2$$

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

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

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, γ) *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, γ) *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, γ) *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

Low rank approximation based on max-vol

Theorem

([Goreinov and Tyrtshnikov, 2001, Thm. 2.1]) Given the matrix

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{3}$$

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

$$\|S(A_{11})\|_{\max} \leq (k+1)\sigma_{k+1}, \tag{4}$$

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

$$AP_c = QR = Q \begin{bmatrix} R_{11} & R_{12} \\ & R_{22} \end{bmatrix}, \tag{5}$$

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],

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

  for $1 \leq i \leq k$ and $1 \leq j \leq n-k$.
- $\sigma_{k+1}(A) \leq \sigma_{max}(R_{22}) = ||R_{22}||$

Rank revealing QR factorization

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

$$AP_c = QR = Q \begin{bmatrix} R_{11} & R_{12} \\ & R_{22} \end{bmatrix}. \tag{6}$$

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

- $Q(:, 1:k)$ forms an approximate orthogonal basis for the range of A,

$$\begin{aligned} A(:,j) &= \sum_{i=1}^{min(j,k)} R(i,j)Q(:,i) \ \in span\{Q(:,1), \ldots Q(:,k)\} \\ Range(A) &\in span\{Q(:,1), \ldots Q(:,k)\} \end{aligned}$$

- $P_c \begin{bmatrix} -R_{11}^{-1}R_{12} \\ I \end{bmatrix}$ is an approximate right null space of A.

Rank revealing QR factorization

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 $R_{11} \in \mathbb{R}^{k \times k}$,

$$
\begin{aligned}
AP_c &= QR = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix} \begin{pmatrix} R_{11} & R_{12} \\ & R_{22} \end{pmatrix}, \\
\tilde{A}_{qr} &= Q_1 \begin{pmatrix} R_{11} & R_{12} \end{pmatrix} P_c^T = Q_1 Q_1^T A = \mathcal{P}^o A
\end{aligned}
$$

- 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,

$$
\begin{aligned}
1 \leq \frac{\sigma_i(A)}{\sigma_i(R_{11})}, \frac{\sigma_j(R_{22})}{\sigma_{k+j}(A)} &\leq \gamma(n,k), \quad \gamma(n,k) = \sqrt{1 + f^2 k(n-k)} \\
||R_{11}^{-1} R_{12}||_{max} &\leq f
\end{aligned}
$$

  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.

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

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

$$Q^T v = w = \begin{bmatrix} w_1 \\ w(2:n) \end{bmatrix}, \;\; ||w(2:n)||_2^2 = ||v||_2^2 - w_1^2$$

QR with column pivoting [Businger and Golub, 1965]

Sketch of the algorithm

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

for j = 1 : n **do**

Find column p of largest norm

if $colnrm[p] > \epsilon$ **then**

1. Pivot: swap columns j and p in A and modify $colnrm$.
2. Compute Householder matrix H_j s.t. $H_j A(j:m,j) = \pm||A(j:m,j)||_2 e_1$.
3. Update $A(j:m, j+1:n) = H_j A(j:m, j+1:n)$.
4. Norm downdate $colnrm(j+1:n)^2 -= A(j, j+1:n)^2$.

else Break

end if

end for

If algorithm stops after k steps

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

Strong RRQR [Gu and Eisenstat, 1996]

Since

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

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

Strong RRQR (contd)

It can be shown that

$$\frac{det(\tilde{R}_{11})}{det(R_{11})} = \sqrt{\left(R_{11}^{-1}R_{12}\right)_{i,j}^2 + \rho_i^2\left(R_{11}\right)\chi_j^2\left(R_{22}\right)} \tag{7}$$

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

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

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.

Strong RRQR (contd)

Theorem

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

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

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

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

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 $\alpha = \sigma_{max}(R_{22})/\sigma_{min}(R_{11})$, and let

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

We have

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

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

$$\begin{aligned} ||W_1||_2^2 &\leq 1 + ||R_{11}^{-1}R_{12}||_2^2 + \alpha^2 = 1 + ||R_{11}^{-1}R_{12}||_2^2 + ||R_{22}||_2^2||R_{11}^{-1}||_2^2 \\ &\leq 1 + ||R_{11}^{-1}R_{12}||_F^2 + ||R_{22}||_F^2||R_{11}^{-1}||_F^2 \\ &= 1 + \sum_{i=1}^{k}\sum_{j=1}^{n-k}\left((R_{11}^{-1}R_{12})_{i,j}^2 + \rho_i^2\left(R_{11}\right)\chi_j^2\left(R_{22}\right)\right) \leq 1 + f^2k(n-k) \end{aligned}$$

We obtain,

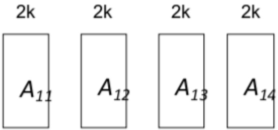
$$\frac{\sigma_i(A)}{\sigma_i(R_{11})} \leq \sqrt{1 + f^2k(n-k)}$$

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

$$\begin{array}{cccc} (A_{11} & A_{12} & A_{13} & A_{14}) \\ \| & \| & \| & \| \\ (Q_{00}R_{00}P_{c\,00}^T & Q_{10}R_{10}P_{c\,10}^T & Q_{20}R_{20}P_{c\,20}^T & Q_{30}R_{30}P_{c\,30}^T \\ \downarrow & \downarrow & \downarrow & \downarrow \\ I_{00} & I_{10} & I_{20} & I_{30} \end{array}$$



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

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

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

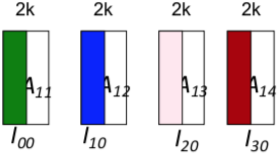
- Return selected columns $A(:, I_{02})$

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

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



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

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

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

- 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

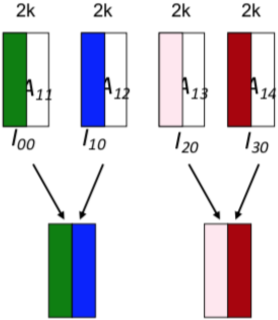
$$\begin{array}{cccc} (A_{11} & A_{12} & A_{13} & A_{14}) \\ \| & \| & \| & \| \\ (Q_{00}R_{00}P_{c00}^T & Q_{10}R_{10}P_{c10}^T & Q_{20}R_{20}P_{c20}^T & Q_{30}R_{30}P_{c30}^T) \\ \downarrow & \downarrow & \downarrow & \downarrow \\ I_{00} & I_{10} & I_{20} & I_{30} \end{array}$$

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

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

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

- 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

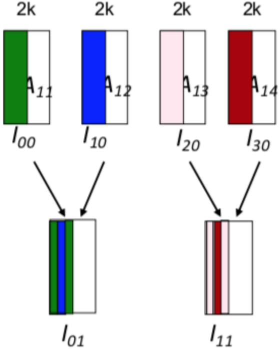
$$
\begin{pmatrix} A_{11} & A_{12} & A_{13} & A_{14} \end{pmatrix}
$$
$$
\begin{pmatrix} Q_{00}R_{00}P_{c00}^T & Q_{10}R_{10}P_{c10}^T & Q_{20}R_{20}P_{c20}^T & Q_{30}R_{30}P_{c30}^T \end{pmatrix}
$$
$$
\downarrow \quad I_{00} \qquad \downarrow \quad I_{10} \qquad \downarrow \quad I_{20} \qquad \downarrow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

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

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

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

- Return selected columns $A(:, I_{02})$



1D tournament pivoting (1Dc-TP)

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

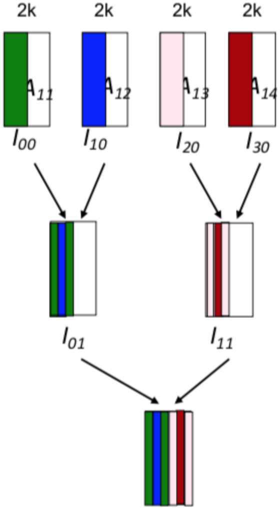
$$\begin{array}{cccc} (A_{11} & A_{12} & A_{13} & A_{14}) \\ \| & \| & \| & \| \\ (Q_{00}R_{00}P_{c00}^T & Q_{10}R_{10}P_{c10}^T & Q_{20}R_{20}P_{c20}^T & Q_{30}R_{30}P_{c30}^T) \\ \downarrow & \downarrow & \downarrow & \downarrow \\ I_{00} & I_{10} & I_{20} & I_{30} \end{array}$$

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

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

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

- 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

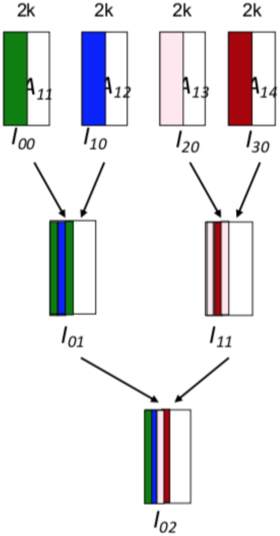
$$\begin{pmatrix} A_{11} & A_{12} & A_{13} & A_{14} \end{pmatrix}$$
$$\| \quad \| \quad \| \quad \|$$
$$\begin{pmatrix} Q_{00}R_{00}P_{c00}^T & Q_{10}R_{10}P_{c10}^T & Q_{20}R_{20}P_{c20}^T & Q_{30}R_{30}P_{c30}^T \end{pmatrix}$$
$$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$$
$$I_{00}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

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

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

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

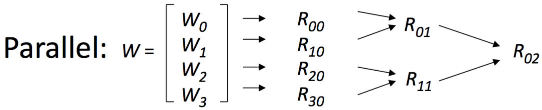
- Return selected columns $A(:, I_{02})$



Select k columns from a tall and skinny matrix

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

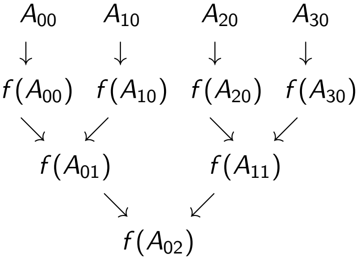
$W = QR_{02}$ using TSQR
$R_{02} P_c = Q_2 R_2$ using QRCP
Return $WP_c(:, 1:k)$



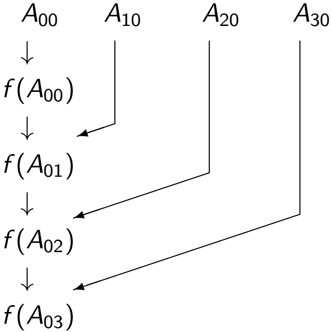
Reduction trees

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

- Binary tree:



- Flat tree:



Notation: at each node of the reduction tree, $f(A_{ij})$ returns the first b columns obtained after performing (strong) RRQR of A_{ij}.

Rank revealing properties of tournament pivoting

It is shown in [Demmel et al., 2015] that the column permutation computed by CA-RRQR satisfies

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

where F_{TP} depends on k, f, n, the shape of reduction tree used during tournament pivoting, and the number of iterations of CARRQR.

CA-RRQR - bounds for one tournament

Selecting k columns by using tournament pivoting reveals the rank of A with the following bounds:

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

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

- Binary tree of depth $\log_2(n/k)$,

$$F_{TP} \leq \frac{1}{\sqrt{2k}} (n/k)^{\log_2\left(\sqrt{2}fk\right)}. \quad (9)$$

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

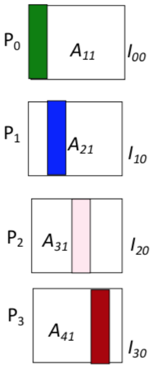
- Flat tree of depth n/k,

$$F_{TP} \leq \frac{1}{\sqrt{2k}} \left(\sqrt{2}fk\right)^{n/k}. \quad (10)$$

Tournament pivoting for 1D row partitioning - 1Dr TP

- Row block partition A as e.g.

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



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

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

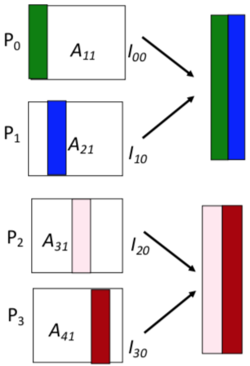
$$A(:, I_{01} \cup I_{11}) = \left(Q_{02} R_{02} P_{c_{02}}^{-1} \right) \rightarrow I_{02}$$

- Return columns $A(:, I_{02})$

Tournament pivoting for 1D row partitioning - 1Dr TP

- Row block partition A as e.g.

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



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

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

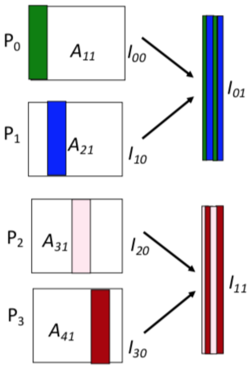
$$A(:, I_{01} \cup I_{11}) = \left(Q_{02} R_{02} P_{c_{02}}^{-1}\right) \rightarrow I_{02}$$

- Return columns $A(:, I_{02})$

Tournament pivoting for 1D row partitioning - 1Dr TP

- Row block partition A as e.g.

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



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

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

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

- Return columns $A(:, I_{02})$

Tournament pivoting for 1D row partitioning - 1Dr TP

- Row block partition A as e.g.

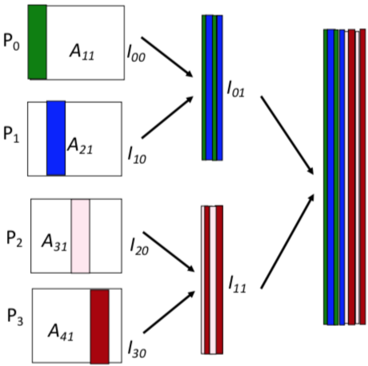
$$A = \begin{pmatrix} A_{11} \\ \hline A_{21} \\ \hline A_{31} \\ \hline A_{41} \end{pmatrix} = \begin{pmatrix} Q_{00} R_{00} P_{c_{00}}^{-1} \\ Q_{10} R_{10} P_{c_{10}}^{-1} \\ Q_{20} R_{20} P_{c_{20}}^{-1} \\ Q_{30} R_{30} P_{c_{30}}^{-1} \end{pmatrix} \begin{matrix} \rightarrow \text{select k cols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

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

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

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

- Return columns $A(:, I_{02})$



Tournament pivoting for 1D row partitioning - 1Dr TP

- Row block partition A as e.g.

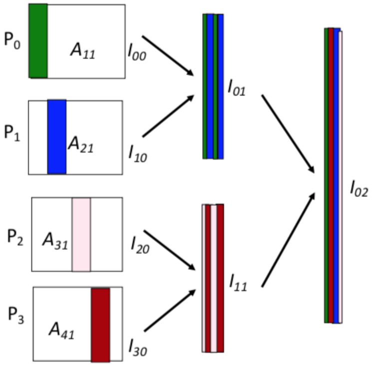
$$A = \begin{pmatrix} A_{11} \\ \hline A_{21} \\ \hline A_{31} \\ \hline A_{41} \end{pmatrix} = \begin{pmatrix} Q_{00} R_{00} P_{c_{00}}^{-1} \\ Q_{10} R_{10} P_{c_{10}}^{-1} \\ Q_{20} R_{20} P_{c_{20}}^{-1} \\ Q_{30} R_{30} P_{c_{30}}^{-1} \end{pmatrix} \begin{matrix} \rightarrow \text{select k cols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

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

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

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

- Return columns $A(:, I_{02})$



CA-RRQR : 2D tournament pivoting

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

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

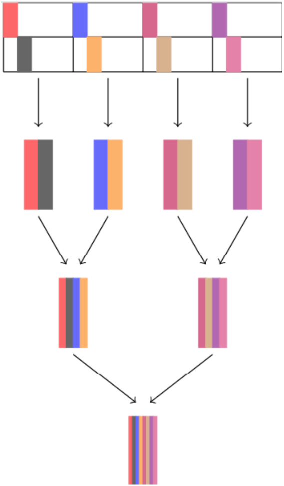
- Select k cols from each column block by 1Dr-TP,

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} \quad \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} \quad \begin{pmatrix} A_{13} \\ A_{23} \end{pmatrix} \quad \begin{pmatrix} A_{14} \\ A_{24} \end{pmatrix}$$
$$\downarrow \qquad \downarrow \qquad \downarrow \qquad \downarrow$$
$$I_{00} \qquad I_{10} \qquad I_{20} \qquad I_{30}$$

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

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

- 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.

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

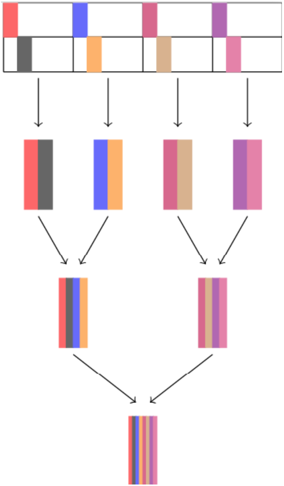
- Select k cols from each column block by 1Dr-TP,

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} \quad \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} \quad \begin{pmatrix} A_{13} \\ A_{23} \end{pmatrix} \quad \begin{pmatrix} A_{14} \\ A_{24} \end{pmatrix}$$
$$\downarrow \qquad \downarrow \qquad \downarrow \qquad \downarrow$$
$$I_{00} \qquad I_{10} \qquad I_{20} \qquad I_{30}$$

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

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

- 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.

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

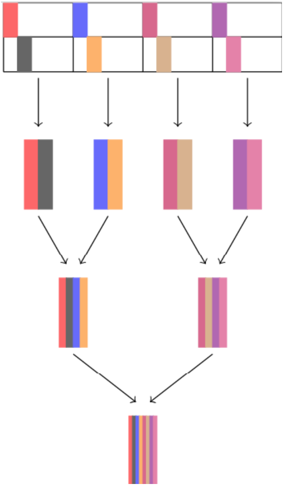
- Select k cols from each column block by 1Dr-TP,

$$\begin{pmatrix} A_{11} \\ A_{21} \end{pmatrix} \quad \begin{pmatrix} A_{12} \\ A_{22} \end{pmatrix} \quad \begin{pmatrix} A_{13} \\ A_{23} \end{pmatrix} \quad \begin{pmatrix} A_{14} \\ A_{24} \end{pmatrix}$$
$$\downarrow \qquad \downarrow \qquad \downarrow \qquad \downarrow$$
$$I_{00} \qquad I_{10} \qquad I_{20} \qquad I_{30}$$

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

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

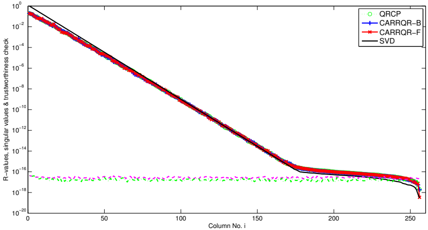
- Return columns selected by 1Dc-TP $A(:, I_{02})$

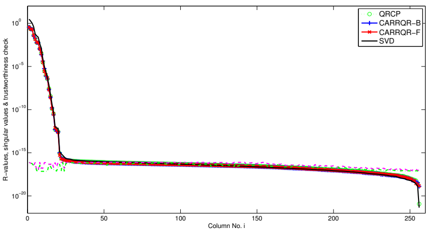


Numerical results

- 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)





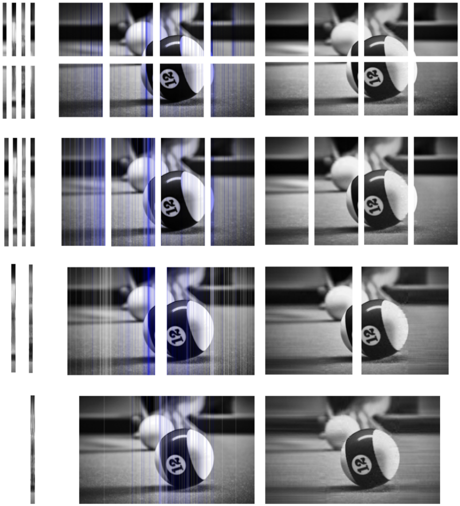
- Left: exponent - exponential Distribution, $\sigma_1 = 1$, $\sigma_i = \alpha^{i-1}$ $(i = 2, \ldots, n)$, $\alpha = 10^{-1/11}$ [Bischof, 1991]
- Right: shaw - 1D image restoration model [Hansen, 2007]

$$\epsilon \min\{||(A\Pi_0)(:,i)||_2\,, ||(A\Pi_1)(:,i)||_2\,, ||(A\Pi_2)(:,i)||_2\} \quad (11)$$

$$\epsilon \max\{||(A\Pi_0)(:,i)||_2\,, ||(A\Pi_1)(:,i)||_2\,, ||(A\Pi_2)(:,i)||_2\} \quad (12)$$

where $\Pi_j (j = 0, 1, 2)$ are the permutation matrices obtained by QRCP, CARRQR-B, and CARRQR-F, and ϵ is the machine precision.

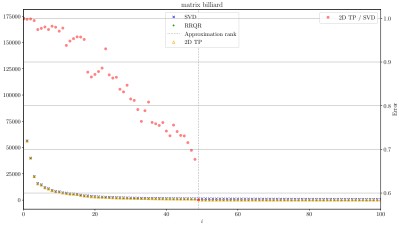
CA-RRQR : 2D tournament pivoting



Numerical experiments

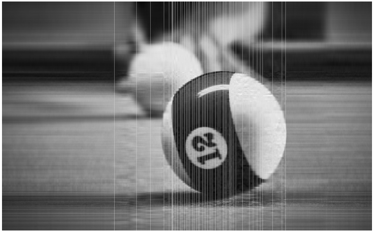
Original image, size 1190 $\times$ 1920

Singular values and ratios



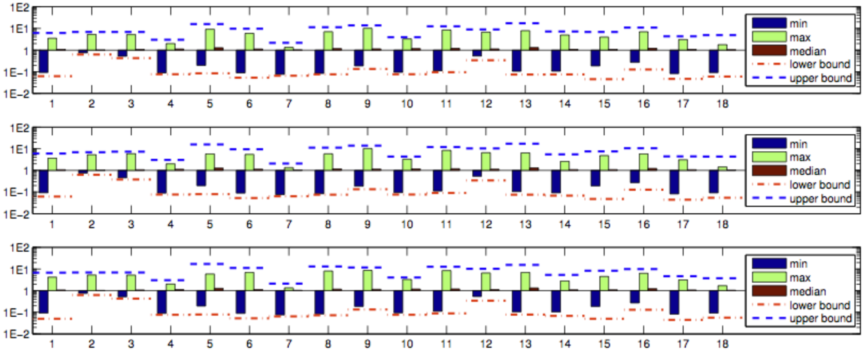
Rank-10 approx, 2D TP 8 $\times$ 8 procs

Rank-50 approx, 2D TP 8 $\times$ 8 procs



- 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

LU versus QR - filled graph $G^+(A)$

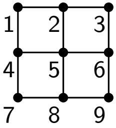
- Consider A is SPD and $A = LL^T$
- Given $G(A) = (V, E)$, $G^+(A) = (V, E^+)$ is defined as:
  there is an edge $(i, j) \in G^+(A)$ iff there is a path from i to j in $G(A)$ going through lower numbered vertices.
- $G(L + L^T) = G^+(A)$, ignoring cancellations.
- Definition holds also for directed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

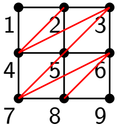
$A =$

	1	2	3	4	5	6	7	8	9
1	x	x		x					
2	x	x	x		x				
3		x	x			x			
4	x			x	x		x		
5		x		x	x	x		x	
6			x		x	x			x
7				x			x	x	
8					x		x	x	x
9						x		x	x

$L + L^T =$

	1	2	3	4	5	6	7	8	9
1	x	x		x					
2	x	x	x	$\color{red}{x}$	x				
3		x	x	$\color{red}{x}$	$\color{red}{x}$	x			
4	x	$\color{red}{x}$	$\color{red}{x}$	x	x		x		
5		x	$\color{red}{x}$	x	x	x	$\color{red}{x}$	x	
6			x		x	x	$\color{red}{x}$	$\color{red}{x}$	x
7				x	$\color{red}{x}$	$\color{red}{x}$	x	x	
8					x	$\color{red}{x}$	x	x	x
9						x		x	x





LU versus QR

Filled column intersection graph $G_{\cap}^{+}(A)$

- Graph of the Cholesky factor of $A^T A$
- $G(R) \subseteq G_{\cap}^{+}(A)$
- $A^T A$ 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

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

- For partial pivoting, $\|L\|_{max} \leq 1$, $\|U\|_{max} \leq 2^n \|A\|_{max}$
  In practice, $\|U\|_{max} \leq \sqrt{n}\|A\|_{max}$

Low rank approximation based on LU factorization

- Given desired rank k, the factorization has the form

$$P_r A P_c = \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{pmatrix} = \begin{pmatrix} I & \\ \bar{A}_{21}\bar{A}_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ & S(\bar{A}_{11}) \end{pmatrix}, \qquad (14)$$

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

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

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

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

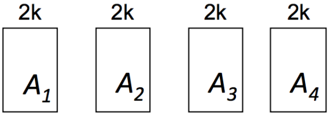
Design space

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.

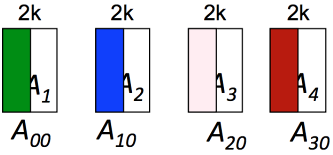
Select k cols using tournament pivoting

- 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}



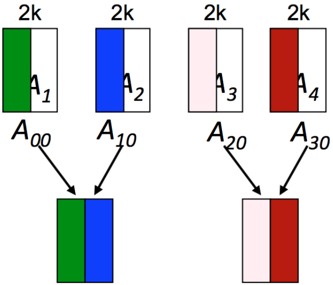
Select k cols using tournament pivoting

- 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}



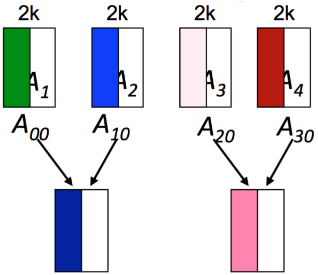
Select k cols using tournament pivoting

- 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}



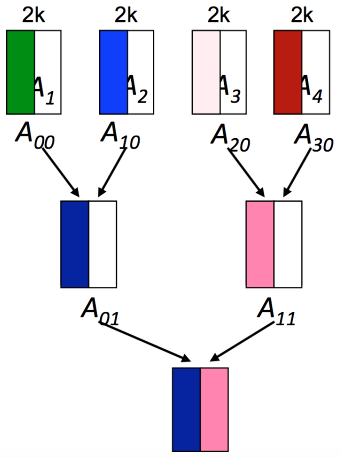
Select k cols using tournament pivoting

- 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}



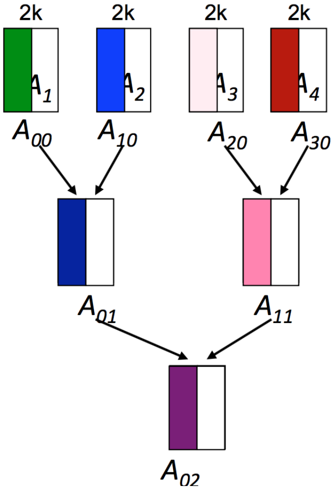
Select k cols using tournament pivoting

- 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_{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}



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 $q_1(n,k)$

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

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

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

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

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

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

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

Orthogonal matrices (contd)

The factorization

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

satisfies:

$$\rho_j(\bar{Q}_{21}\bar{Q}_{11}^{-1}) \leq F_{TP}, \quad (19)$$

$$\frac{1}{q_2(m,k)} \leq \sigma_i(\bar{Q}_{11}) \leq 1, \quad (20)$$

$$\sigma_{min}(\bar{Q}_{11}) = \sigma_{min}(\bar{Q}_{22}) \quad (21)$$

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 $\sigma_{min}(\bar{Q}_{11}) = \sigma_{min}(\bar{Q}_{22})$ by considering unit vectors $x \in \mathbb{R}^k, y \in \mathbb{R}^{m-k}$

$$1 = ||\bar{Q}_{11}x||^2 + ||\bar{Q}_{21}x||^2, \; 1 = ||\bar{Q}_{22}^T y||^2 + ||\bar{Q}_{21}^T y||^2$$

and showing $\min_{||x||=1} ||\bar{Q}_{11}x||^2 = \min_{||y||=1} ||\bar{Q}_{22}^T y||^2$

Sketch of the proof

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

where

$$
\begin{aligned}
\bar{Q}_{21}\bar{Q}_{11}^{-1} &= \bar{A}_{21}\bar{A}_{11}^{-1}, \\
S(\bar{A}_{11}) &= S(\bar{Q}_{11})R_{22} = \bar{Q}_{22}^{-T} R_{22}.
\end{aligned}
$$

Sketch of the proof (contd)

$$\begin{aligned} \bar{A}_{11} &= \bar{Q}_{11} R_{11}, & (23) \\ S(\bar{A}_{11}) &= S(\bar{Q}_{11}) R_{22} = \bar{Q}_{22}^{-T} R_{22}. & (24) \end{aligned}$$

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

$$\begin{aligned} \sigma_{k+j}(A) \leq \sigma_j(S(\bar{A}_{11})) &= \sigma_j(S(\bar{Q}_{11}) R_{22}) \leq ||S(\bar{Q}_{11})||_2 \sigma_j(R_{22}) \\ &\leq q_1(n,k) q_2(m,k) \sigma_{k+j}(A), \end{aligned}$$

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 $rank = k$

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

$$\bar{A} = P_r A P_c = \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{pmatrix} = \begin{pmatrix} I & \\ \bar{Q}_{21}\bar{Q}_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ & S(\bar{A}_{11}) \end{pmatrix} \quad (25)$$

where $\bar{A}_{11}$ is of size $k \times k$ and

$$S(\bar{A}_{11}) = \bar{A}_{22} - \bar{A}_{21}\bar{A}_{11}^{-1}\bar{A}_{12} = \bar{A}_{22} - \bar{Q}_{21}\bar{Q}_{11}^{-1}\bar{A}_{12}. \quad (26)$$

It satisfies the following properties:

$$\begin{aligned} \rho_l(\bar{A}_{21}\bar{A}_{11}^{-1}) &= \rho_l(\bar{Q}_{21}\bar{Q}_{11}^{-1}) \le F_{TP}, & (27) \\ ||S(\bar{A}_{11})||_{max} &\le \min((1 + F_{TP}\sqrt{k})||A||_{max}, F_{TP}\sqrt{1 + F_{TP}^2(m-k)}\sigma_k(A)) \end{aligned}$$

$$1 \le \frac{\sigma_i(A)}{\sigma_i(\bar{A}_{11})}, \frac{\sigma_j(S(\bar{A}_{11}))}{\sigma_{k+j}(A)} \le q(m,n,k), \quad (28)$$

for any $1 \le l \le m-k$, $1 \le i \le k$, and $1 \le j \le \min(m,n) - k$,
$q(m,n,k) = q_1(n,k)q_2(m,k) = \sqrt{(1 + F_{TP}^2(n-k))\,(1 + F_{TP}^2(m-k))}$.

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.

$$\begin{aligned}
|S(\bar{A}_{11})(i,j)| &= |\bar{A}_{22}(i,j) - (\bar{A}_{21}\bar{A}_{11}^{-1})(i,:)\bar{A}_{12}(:,j)| \\
&\leq ||A||_{max} + ||(\bar{A}_{21}\bar{A}_{11}^{-1})(i,:)||_2||\bar{A}_{12}(:,j)||_2 \\
&\leq ||A||_{max} + \rho_i(\bar{A}_{21}\bar{A}_{11}^{-1})\sqrt{k}||A||_{max} \\
&\leq (1 + F_{TP}\sqrt{k})||A||_{max}
\end{aligned}$$

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

$$\begin{aligned}
|S(\bar{A}_{11})(i,j)| &= |\bar{Q}_{22}^{-T}(i,:)R_{22}(:,j)| \leq ||\bar{Q}_{22}^{-1}(:,i)||_2||R_{22}(:,j)||_2 \\
&\leq ||\bar{Q}_{22}^{-1}||_2||R_{22}(:,j)||_2 = ||R_{22}(:,j)||_2/\sigma_{min}(\bar{Q}_{22}) \\
&\leq q_2(m,k)F_{TP}\sigma_k(A).
\end{aligned}$$

Hence:

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

Plan

Low rank matrix approximation

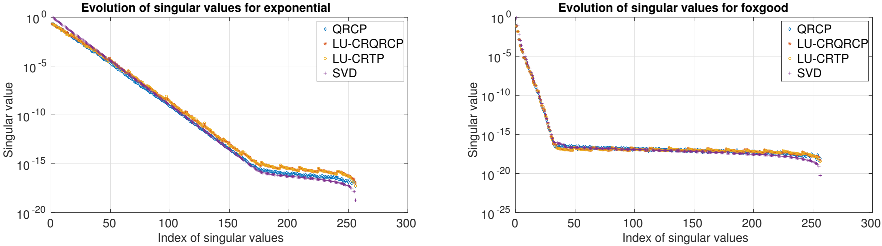
Low rank approximation based on max-vol

Rank revealing QR factorization

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

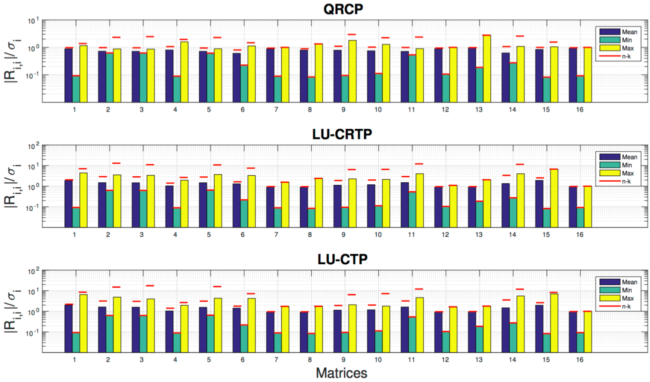
Experimental results, LU_CRTP

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

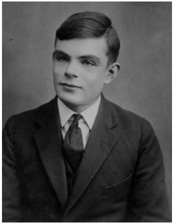
Numerical results



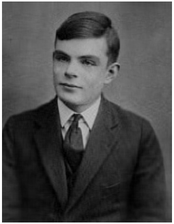
- Here $k = 16$ and the factorization is truncated at $K = 128$ (bars) or $K = 240$ (red lines).
- LU_CTP: Column tournament pivoting + partial pivoting
- All singular values smaller than machine precision, ϵ, are replaced by ϵ.
- The number along x-axis represents the index of test matrices.

Results for image of size 919 × 707

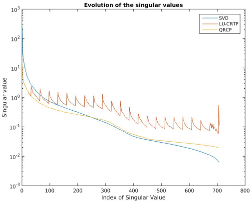
Original image



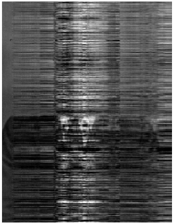
Rank-38 approx, SVD



Singular value distribution

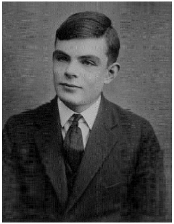


Rank-38 approx, LUPP



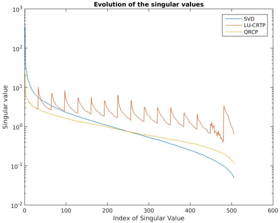
Rank-38 approx, LU_CRTP

Rank-75 approx, LU_CRTP



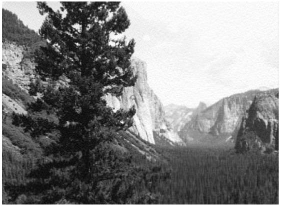
Results for image of size 691 × 505

Singular value distribution



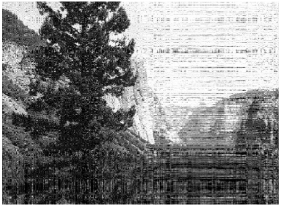
Original image

Rank-105 approx, SVD

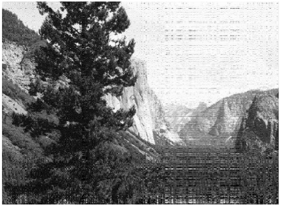


Rank-105 approx, LUPP

Rank-105 approx, LU_CRTP



Rank-209 approx, LU_CRTP



Comparing nnz in the factors L, U versus Q, R

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

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

	Time 2k cols	*Time leaves 32procs SPQR + dGEQP3*	*Number of MPI processes* 16	32	64	128	256	512	1024
Parab_fem	0.26	0.26 + 1129	46.7	24.5	13.7	8.4	5.9	4.8	4.4
Mac_econ	0.46	25.4 + 510	132.7	86.3	111.4	59.6	27.2	–	–

More details on CA deterministic algorithms

- [Demmel et al., 2015] Communication avoiding rank revealing QR factorization with column pivoting Demmel, Grigori, Gu, Xiang, SIAM J. Matrix Analysis and Applications, 2015.
- 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.

References (1)

Bischof, C. H. (1991).
A parallel QR factorization algorithm with controlled local pivoting.
SIAM J. Sci. Stat. Comput., 12:36–57.

Businger, P. A. and Golub, G. H. (1965).
Linear least squares solutions by Householder transformations.
Numer. Math., 7:269–276.

Civril, A. and Magdon-Ismail, M. (2013).
Exponential inapproximability of selecting a maximum volume sub-matrix.
Algorithmica, 65(1):159–176.

Demmel, J., Grigori, L., Gu, M., and Xiang, H. (2015).
Communication-avoiding rank-revealing qr decomposition.
SIAM Journal on Matrix Analysis and its Applications, 36(1):55–89.

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

Eisenstat, S. C. and Ipsen, I. C. F. (1995).
Relative perturbation techniques for singular value problems.
SIAM J. Numer. Anal., 32(6):1972–1988.

Goreinov, S. and Tyrtshnikov, E. (2001).
The maximal-volume concept in approximation by low-rank matrices.
Contemporary Mathematics, 280:47–52.

References (2)

Gu, M. and Eisenstat, S. C. (1996).
Efficient algorithms for computing a strong rank-revealing QR factorization.
SIAM J. Sci. Comput., 17(4):848–869.

Hansen, P. C. (2007).
Regularization tools: A matlab package for analysis and solution of discrete ill-posed problems.
Numerical Algorithms, (46):189–194.

Results used in the proofs

- Interlacing property of singular values [Golub, Van Loan, 4th edition, page 487]
  Let $A = [a_1| \dots |a_n]$ be a column partitioning of an $m \times n$ matrix with $m \geq n$. If $A_r = [a_1| \dots |a_r]$, then for $r = 1 : n-1$

  $$\sigma_1(A_{r+1}) \geq \sigma_1(A_r) \geq \sigma_2(A_{r+1}) \geq \dots \geq \sigma_r(A_{r+1}) \geq \sigma_r(A_r) \geq \sigma_{r+1}(A_{r+1}).$$

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

  $$\sigma_{min}(B)\sigma_j(C) \leq \sigma_j(BC) \leq \sigma_{max}(B)\sigma_j(C), j = 1, \dots, k.$$