Low rank approximation of a sparse matrix

L. Grigori

Inria Paris and UPMC
with A. Ayala, S. Cayrols, J. Demmel

February 2017

Plan

Low rank matrix approximation

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP

Plan

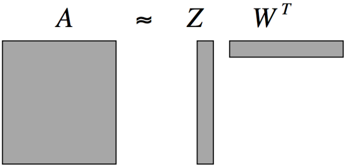
Low rank matrix approximation

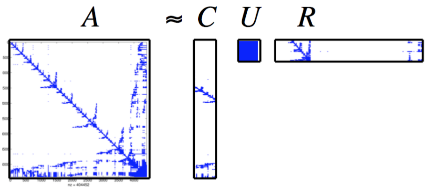
LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP

Low rank matrix approximation

- Problem: given $m \times n$ matrix A, 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, ...

$$Ax \rightarrow ZW^T x$$

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

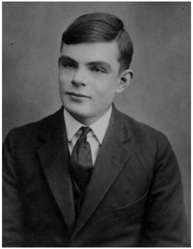
Low rank matrix approximation

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

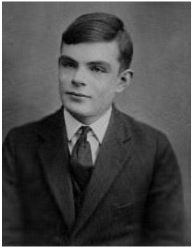
$$\min_{rank(\tilde{A}_k) \leq k} ||A - \tilde{A}_k||_2 = ||A - A_k||_2 = \sigma_{k+1}(A) \quad (1)$$

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

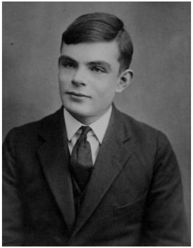
Original image of size 919×707



Rank-38 approximation, SVD

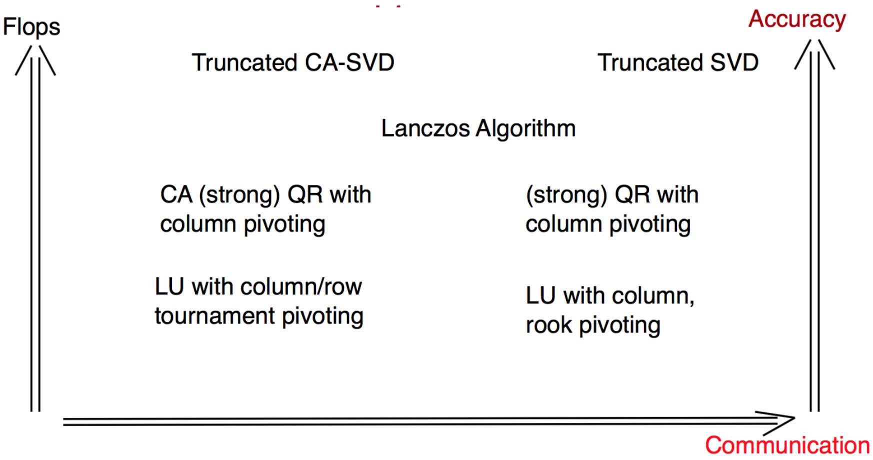


Rank-75 approximation, SVD



- Image source: `https://upload.wikimedia.org/wikipedia/commons/a/a1/Alan_Turing_Aged_16.jpg`

Low rank matrix approximation: trade-offs



Rank revealing QR factorization

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

$$AP_c = QR = Q \begin{bmatrix} R_{11} & R_{12} \\ & R_{22} \end{bmatrix}, \tag{3}$$

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.

- $Q(:, 1:k)$ forms an approximate orthogonal basis for the range of A,
- $P_c \begin{bmatrix} R_{11}^{-1} R_{12} \\ -I \end{bmatrix}$ is an approximate right null space of A.

Rank revealing QR factorization

The factorization from equation (3) is rank revealing if

$$1 \leq \frac{\sigma_i(A)}{\sigma_i(R_{11})}, \frac{\sigma_j(R_{22})}{\sigma_{k+j}(A)} \leq q_1(n,k),$$

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

$$||R_{11}^{-1} R_{12}||_{max} \leq q_2(n,k)$$

- Gu and Eisenstat show that given k and f, there exists a P_c such that $q_1(n,k) = \sqrt{1 + f^2 k(n-k)}$ and $q_2(n,k) = f$.
- Factorization computed in $O(mnk)$ flops.

Plan

Low rank matrix approximation

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP

Low rank approximation based on LU factorization

- Given desired rank k, the factorization has the form

$$P_r A P_c = \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{pmatrix} = \begin{pmatrix} I & \\ \bar{A}_{21}\bar{A}_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ & S(\bar{A}_{11}) \end{pmatrix}, \tag{4}$$

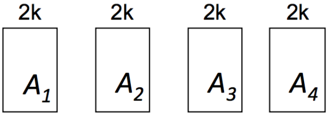
  where $A \in \mathbb{R}^{m \times n}$, $\bar{A}_{11} \in \mathbb{R}^{k,k}$, $S(\bar{A}_{11}) = \bar{A}_{22} - \bar{A}_{21}\bar{A}_{11}^{-1}\bar{A}_{12}$.
- The rank-k approximation matrix $\tilde{A}_k$ is

$$\tilde{A}_k = \begin{pmatrix} I \\ \bar{A}_{21}\bar{A}_{11}^{-1} \end{pmatrix} \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \end{pmatrix} = \begin{pmatrix} \bar{A}_{11} \\ \bar{A}_{21} \end{pmatrix} \bar{A}_{11}^{-1} \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \end{pmatrix}. \tag{5}$$

- $\bar{A}_{11}^{-1}$ is never formed, its factorization is used when $\tilde{A}_k$ is applied to a vector.
- In randomized algorithms, $U = C^+ A R^+$, where C^+, R^+ are Moore-Penrose generalized inverses.

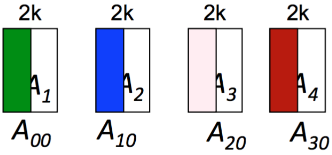
Select k cols using tournament pivoting

- 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}



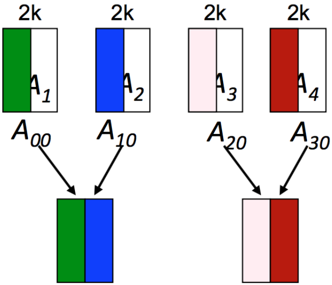
Select k cols using tournament pivoting

- 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}



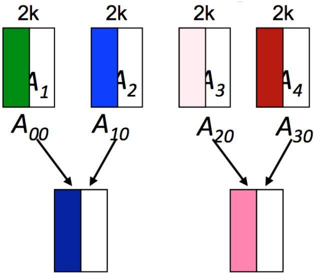
Select k cols using tournament pivoting

- 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}



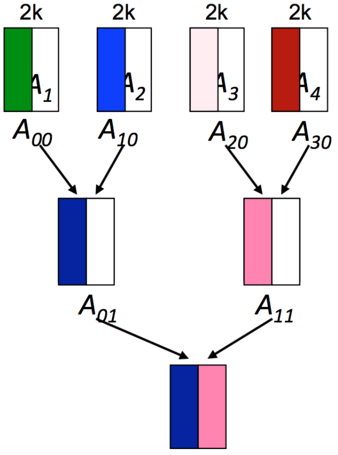
Select k cols using tournament pivoting

- 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}



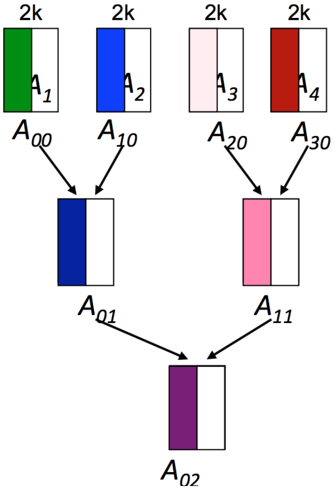
Select k cols using tournament pivoting

- 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}



Our 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 $||\bar{Q}_{21}\bar{Q}_{11}^{-1}||_{max} \le F_{TP}$ and bounds for s.v. governed by $q_2(m, k)$.
Binary tree of depth $\log_2(n/k)$,
$$F_{TP} \le \frac{1}{\sqrt{2k}} (n/k)^{\log_2\left(\sqrt{2}fk\right)}. \quad (6)$$

Orthogonal matrices

The factorization

$$P_r Q = \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ \bar{Q}_{21} & \bar{Q}_{22} \end{pmatrix} = \begin{pmatrix} I & \\ \bar{Q}_{21}\bar{Q}_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ & S(\bar{Q}_{11}) \end{pmatrix} \tag{7}$$

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

$$\rho_j(\bar{Q}_{21}\bar{Q}_{11}^{-1}) \leq F_{TP}, \tag{8}$$

$$\frac{1}{q_2(m,k)} \leq \sigma_i(\bar{Q}_{11}) \leq 1, \tag{9}$$

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, $q_2(m,k) = \sqrt{1 + F_{TP}^2(m-k)}$.

The obtained factorization

$$
\begin{aligned}
P_r A P_c &= \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{pmatrix} = \begin{pmatrix} I & \\ \bar{A}_{21}\bar{A}_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ & S(\bar{A}_{11}) \end{pmatrix} \\
&= \begin{pmatrix} I & \\ \bar{Q}_{21}\bar{Q}_{11}^{-1} & I \end{pmatrix} \begin{pmatrix} \bar{Q}_{11} & \bar{Q}_{12} \\ & S(\bar{Q}_{11}) \end{pmatrix} \begin{pmatrix} R_{11} & R_{12} \\ & R_{22} \end{pmatrix} \qquad (10)
\end{aligned}
$$

where

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

- Similarities with the proof of existence of a RRLU by [Pan, LAA, 2000].

LU_CRTP factorization - bounds if $rank = k$

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

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

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

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

It satisfies the following properties:

$$\rho_l(\bar{A}_{21}\bar{A}_{11}^{-1}) = \rho_l(\bar{Q}_{21}\bar{Q}_{11}^{-1}) \leq F_{TP}, \tag{13}$$

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

$$1 \leq \frac{\sigma_i(A)}{\sigma_i(\bar{A}_{11})}, \frac{\sigma_j(S(\bar{A}_{11}))}{\sigma_{k+j}(A)} \leq q(m,n,k), \tag{14}$$

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

Tournament pivoting for sparse matrices

Arithmetic complexity

A has arbitrary sparsity structure

$$flops(TP_{FT}) \leq 2nnz(A)k^2$$
$$flops(TP_{BT}) \leq 8\frac{nnz(A)}{P}k^2 \log \frac{n}{k}$$

$G(A^T A)$ is an $n^{1/2}$- separable graph

$$flops(TP_{FT}) \leq O(nnz(A)k^{3/2})$$
$$flops(TP_{BT}) \leq O(\frac{nnz(A)}{P}k^{3/2} \log \frac{n}{k})$$

Randomized algorithm by Clarkson and Woodruff, STOC'13

- Given $n \times n$ matrix A, it computes LDW^T, where D is $k \times k$ such that $||A - LDW^T||_F \leq (1+\epsilon)||A - A_k||_F$, A_k is best rank-k approximation.

$$flops \leq O(nnz(A)) + n\epsilon^{-4} log^{O(1)}(n\epsilon^{-4})$$

- Tournament pivoting is faster if $\epsilon \leq \frac{1}{(nnz(A)/n)^{1/4}}$
or if $\epsilon = 0.1$ and $nnz(A)/n \leq 10^4$.

Tournament pivoting for sparse matrices

Arithmetic complexity

A has arbitrary sparsity structure	$G(A^TA)$ is an $n^{1/2}$- separable graph
$flops(TP_{FT}) \leq 2nnz(A)k^2$	$flops(TP_{FT}) \leq O(nnz(A)k^{3/2})$
$flops(TP_{BT}) \leq 8\frac{nnz(A)}{P}k^2 \log \frac{n}{k}$	$flops(TP_{BT}) \leq O(\frac{nnz(A)}{P}k^{3/2} \log \frac{n}{k})$

Randomized algorithm by Clarkson and Woodruff, STOC'13

- Given $n \times n$ matrix A, it computes LDW^T, where D is $k \times k$ such that $||A - LDW^T||_F \leq (1+\epsilon)||A - A_k||_F$, A_k is best rank-k approximation.

$$flops \leq O(nnz(A)) + n\epsilon^{-4} log^{O(1)}(n\epsilon^{-4})$$

- Tournament pivoting is faster if $\epsilon \leq \frac{1}{(nnz(A)/n)^{1/4}}$ or if $\epsilon = 0.1$ and $nnz(A)/n \leq 10^4$.

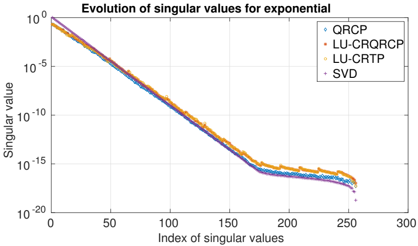
Plan

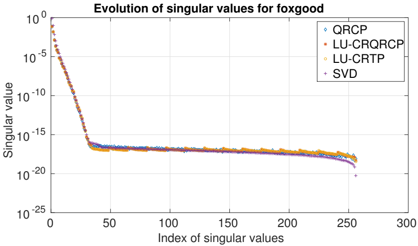
Low rank matrix approximation

LU_CRTP: Truncated LU factorization with column and row tournament pivoting

Experimental results, LU_CRTP

Numerical results

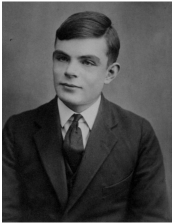




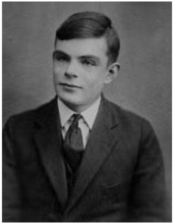
- Left: exponent - exponential Distribution, $\sigma_1 = 1$, $\sigma_i = \alpha^{i-1}$ $(i = 2, \ldots, n)$, $\alpha = 10^{-1/11}$ [Bischof, 1991]
- Right: foxgood - Severely ill-posed test problem of the 1st kind Fredholm integral equation used by Fox and Goodwin

Results for image of size 919 × 707

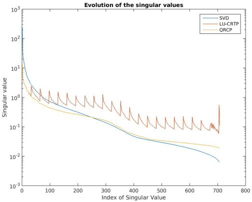
Original image



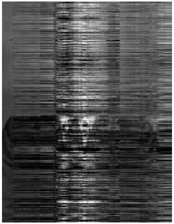
Rank-38 approx, SVD



Singular value distribution

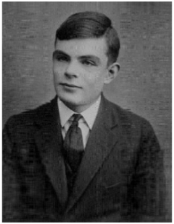


Rank-38 approx, LUPP



Rank-38 approx, LU_CRTP

Rank-75 approx, LU_CRTP



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*
*gemat*11	1232	128	2.1	2.2
4929	4895	512	3.3	2.6
	9583	1024	11.5	3.2
*wang*3	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	0.9
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

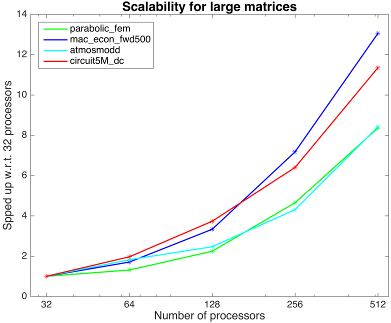
Selection of 256 columns by tournament pivoting

- Mesu (UPMC): Intel Xeon Sandy Bridge (8 cores) 28 nodes, 24 cores per node
- Tournament pivoting uses SPQR (T. Davis) + dGEQP3 (Lapack), time in secs

Matrices:

- Parab_fem: 528825×528825, $3,674,625$ nnz
- Mac_econ: 206500×206500 $1,273,389$ nnz
- atmosmodd: $1,270,432 \times 1,270,432$ $8,814,880$ nnz
- circuit5M_dc: $3,523,317 \times 3,523,317$ $19,194,193$ nnz

Performance results (contd)



	Number of MPI processes				
	32	64	128	256	512
Parab_fem	57.6	44.0	25.7	12.4	6.9
Mac_econ	94.0	55.1	28.2	13.1	7.2
atmosmodd	370.3	203.3	150.1	86.0	44.0
circuit5M_dc	916.0	465.9	245.4	143.1	80.7

- Tournament pivoting: flat local + binary between processes
- SPQR + DGEQP3 for one block of Parab_fem (dimension 528825 × 16432) is 0.26 + 1129 secs.

Talk based on the papers

- [Demmel et al., 2012] Communication-optimal parallel and sequential QR and LU factorizations, J. W. Demmel, L. Grigori, M. Hoemmen, and J. Langou, SIAM Journal on Scientific Computing, Vol. 34, No 1, 2012.
- [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 technical report 9023, 2016.

References (1)

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

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.

Demmel, J. W., Grigori, L., Hoemmen, M., and Langou, J. (2012).
Communication-optimal parallel and sequential QR and LU factorizations.
SIAM Journal on Scientific Computing, (1):206–239.
short version of technical report UCB/EECS-2008-89 from 2008.

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

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.