MANCHESTER 1824

The University of Manchester

M\crNA
Manchester Numerical Analysis

Mixed Precision Tall and Thin QR Factorization

Françoise Tisseur
Department of Mathematics
The University of Manchester

Joint work with Srikara Pranesh

New Directions in NLA and HPC: Celebrating the 70th Birthday of Jack Dongarra, July 7-8, 2021

Objective

Let $A \in \mathbb{R}^{m \times n}$, $m \gg n$ have full rank. $\kappa_2(A) = \|A\|_2 \|A^+\|_2$.

Reduced QR factorization: $A = QR$, $Q \in \mathbb{R}^{m \times n}$, $Q^T Q = I_n$, and $R \in \mathbb{R}^{n \times n}$ upper triangular.

Want to compute $\widehat{Q}$, $\widehat{R}$ s.t.

$$\|I_n - \widehat{Q}^T \widehat{Q}\|_2 \le c_1 u, \quad \|\widehat{Q}\widehat{R} - A\|_2 \le c_2 \|A\|_2 u$$

exploiting mixed precision arithmetics (u is the unit roundoff of working precision.)

Objective

Let $A \in \mathbb{R}^{m\times n}$, $m \gg n$ have full rank. $\kappa_2(A) = \|A\|_2\|A^+\|_2$.

Reduced QR factorization: $A = QR$, $Q \in \mathbb{R}^{m\times n}$, $Q^TQ = I_n$, and $R \in \mathbb{R}^{n\times n}$ upper triangular.

Want to compute $\widehat{Q}$, $\widehat{R}$ s.t.

$$\|I_n - \widehat{Q}^T\widehat{Q}\|_2 \le c_1 u, \quad \|\widehat{Q}\widehat{R} - A\|_2 \le c_2\|A\|_2 u$$

exploiting mixed precision arithmetics (u is the unit roundoff of working precision.)

Two classes of algorithms:

- orthogonal triangularization (Householder QR, TSQR),
- triangular orthogonalization (CGS, MGS, **Cholesky-QR**).

Cholesky-QR Algorithm

$A \in \mathbb{R}^{m \times n}$, $m \gg n$, full rank.

CholeskyQR algorithm:

1. $G = A^T A$
2. $R = \text{chol}(G)$ (upper triang. Cholesky factor of G)
3. $Q = AR^{-1}$

- Rich in BLAS 3 operations.
- Comm. avoiding alg. with similar cost to TSQR.
- Step 2 may fail when $\kappa_2(A) \gtrsim u^{-1/2}$.
- $\|I_n - Q^T Q\|_2 = O(u\kappa_2(A)^2)$.

Cholesky-QR Algorithm

$A \in \mathbb{R}^{m\times n}$, $m \gg n$, full rank.

CholeskyQR algorithm:

1 $G = A^T A$
2 $R = \mathrm{chol}(G)$ (upper triang. Cholesky factor of G)
3 $Q = AR^{-1}$

- Rich in BLAS 3 operations.
- Comm. avoiding alg. with similar cost to TSQR.
- Step 2 may fail when $\kappa_2(A) \gtrsim u^{-1/2}$.
- $\|I_n - Q^T Q\|_2 = O(u\kappa_2(A)^2)$.

[Yamazaki, Tomov, Dongarra, SISC 2015] use double-double precision in steps 1–2.

ShiftedCholeskyQR3

It follows from [Yamamoto et al. ETNA 2015] that if $R = \text{chol}(A^T A)$ succeeds then $\kappa(AR^{-1}) \approx \max\left(1, u^{1/2}\kappa_2(A)\right)$.

Apply Cholesky-QR

- twice if $1 < \kappa_2(A) \lesssim u^{-1/2}$ [Yamamoto et al., ETNA 2015];
- thrice if $u^{-1/2} \lesssim \kappa_2(A) \lesssim u^{-1}$ [Fukaya et al., SISC 2020].

ShiftedCholeskyQR3 algorithm:

1 $[Q_0, R_0]$ = ShiftedCholeskyQR(A)
2 $[Q_1, R_1]$ = CholeskyQR(Q_0), $R = R_1 R_0$
3 $[Q, R_2]$ = CholeskyQR(Q_1), $R = R_2 R$

ShiftedCholeskyQR3

It follows from [Yamamoto et al. ETNA 2015] that if $R = \text{chol}(A^T A)$ succeeds then $\kappa(AR^{-1}) \approx \max\left(1, u^{1/2}\kappa_2(A)\right)$.

Apply Cholesky-QR

- twice if $1 < \kappa_2(A) \lesssim u^{-1/2}$ [Yamamoto et al., ETNA 2015];
- thrice if $u^{-1/2} \lesssim \kappa_2(A) \lesssim u^{-1}$ [Fukaya et al., SISC 2020].

ShiftedCholeskyQR3 algorithm: [Assume $\kappa_2(A) \approx u^{-1}$]

1 $[Q_0, R_0]$ = ShiftedCholeskyQR(A) [$\kappa_2(Q_0) \approx \sqrt{u^{-1}}$]
2 $[Q_1, R_1]$ = CholeskyQR(Q_0), $R = R_1 R_0$ [$\kappa_2(Q_1) \approx O(1)$]
3 $[Q, R_2]$ = CholeskyQR(Q_1), $R = R_2 R$

ShiftedCholeskyQR3

It follows from [Yamamoto et al. ETNA 2015] that if $R = \text{chol}(A^T A)$ succeeds then $\kappa(AR^{-1}) \approx \max\left(1, u^{1/2}\kappa_2(A)\right)$.

Apply Cholesky-QR

- twice if $1 < \kappa_2(A) \lesssim u^{-1/2}$ [Yamamoto et al., ETNA 2015];
- thrice if $u^{-1/2} \lesssim \kappa_2(A) \lesssim u^{-1}$ [Fukaya et al., SISC 2020].

ShiftedCholeskyQR3 algorithm: [Assume $\kappa_2(A) \approx u^{-1}$]

1. $[Q_0, R_0]$ = ShiftedCholeskyQR(A) [$\kappa_2(Q_0) \approx \sqrt{u^{-1}}$]
2. $[Q_1, R_1]$ = CholeskyQR(Q_0), $R = R_1 R_0$ [$\kappa_2(Q_1) \approx O(1)$]
3. $[Q, R_2]$ = CholeskyQR(Q_1), $R = R_2 R$

Steps1-2 construct **preconditioner** $R_1 R_0$ s.t.
$Q_1 = A(R_1 R_0)^{-1}$, $\kappa_2(Q_1) \approx 1$ at twice the cost of step 3!

LU-Cholesky QR

[Terao, Ozaki, Ogita, Parallel Computing 2020] precondition A with

$[\widetilde{Q}, \widetilde{R}]$ = LU-CholeskyQR(A):

1. $[L, U, P] = \mathrm{LU}(A)$ (LU fact. with partial piv.)
2. $S = \mathrm{chol}(L^T L)$
3. $\widetilde{R} = SU$, $\widetilde{Q} = A\widetilde{R}^{-1}$.

[Note: $PA = LU$ and $A^T A = U^T L^T P P^T L U = U^T S^T S U =: \widetilde{R}^T \widetilde{R}$.]

LU-Cholesky QR

[Terao, Ozaki, Ogita, Parallel Computing 2020] precondition A with

$[\widetilde{Q}, \widetilde{R}]$ = LU-CholeskyQR(A): [Assume $\kappa_2(A) \lesssim u^{-1} \approx 10^{16}$]

1. $[L, U, P] = \mathrm{LU}(A)$ (LU fact. with partial piv.)
2. $S = \mathrm{chol}(L^T L)$
3. $\widetilde{R} = SU$, $\widetilde{Q} = A\widetilde{R}^{-1}$. [$\kappa_2(\widetilde{Q}) \approx 1$]

[Note: $PA = LU$ and $A^T A = U^T L^T P P^T L U = U^T S^T S U =: \widetilde{R}^T \widetilde{R}$.]

LU-Cholesky QR

[Terao, Ozaki, Ogita, Parallel Computing 2020] precondition A with

$[\widetilde{Q}, \widetilde{R}]$ = LU-CholeskyQR(A): [Assume $\kappa_2(A) \lesssim u^{-1} \approx 10^{16}$]

1. $[L, U, P] = \mathrm{LU}(A)$ (LU fact. with partial piv.)
2. $S = \mathrm{chol}(L^T L)$
3. $\widetilde{R} = SU$, $\widetilde{Q} = A\widetilde{R}^{-1}$. [$\kappa_2(\widetilde{Q}) \approx 1$]

[Note: $PA = LU$ and $A^T A = U^T L^T P P^T L U = U^T S^T S U =: \widetilde{R}^T \widetilde{R}$.]

LU-CholQR2 algorithm:

1. $[\widetilde{Q}, \widetilde{R}]$ = LU-CholeskyQR(A) (preconditioning)
2. $[Q, R_1]$ = CholeskyQR($\widetilde{Q}$), $R = R_1 \widetilde{R}$

- Similar cost to CholeskyQR3 but not comm. avoiding.

Preconditioning of A in Mixed Precision

Let $u \leq u_\ell$
(e.g., u is rounding error for fp64, u_ℓ is fp32, fp16 or bfloat16).

Assume $\kappa_2(\mathrm{fl}_\ell(A)) \approx \min\{u_\ell^{-1}, \kappa_2(A)\}$.

The steps

1. $[L, U, P] = \mathrm{LU}(A)$ in precision u_ℓ
2. $G = L^T L$ in precision u_ℓ
3. $S = \mathrm{chol}(G)$ in precision u
4. $\widetilde{R} = SU$ in precision u

compute preconditioner $\widetilde{R}$ s.t.

$$\kappa_2(A\widetilde{R}^{-1}) \approx \max\{1, u_\ell \kappa_2(A)\}.$$

Experiment 1, $m = 1000$, $n = 10$

Preconditioner $\widetilde{R}$ computed in fp16, $u_\ell = 4.88 \times 10^{-4}$.

Q, R computed by CholeskyQR applied to $A\widetilde{R}^{-1}$.

res $= \|A - QR\|_2 / \|A\|_2$. A = randsvd matrix.

$\kappa_2(A)$	$\kappa_2(\mathrm{fl}_\ell(A))$	$\kappa_2(A\widetilde{R}^{-1})$	$\|I - Q^TQ\|_2$	res
1.0e+2	1.0e+2	1.3e+0	2.7e-16	1.8e-16
1.0e+3	1.0e+3	1.3e+0	6.7e-16	1.4e-16
1.0e+4	8.9e+3	1.3e+0	4.7e-16	2.2e-16
1.0e+5	3.6e+4	3.4e+0	1.1e-15	1.2e-16
1.0e+6	2.4e+4	2.6e+1	1.8e-14	1.8e-16
1.0e+7	3.0e+4	4.3e+2	9.0e-12	9.9e-17
1.0e+8	3.3e+4	2.4e+3	5.6e-11	1.1e-16

$\kappa_2(\mathrm{fl}_\ell(A)) \approx \min\{u_\ell^{-1}, \kappa_2(A)\}$, $\kappa_2(A\widetilde{R}^{-1}) \approx \max\{1, u_\ell \kappa_2(A)\}$.

Precondition A in Three Precisions

Let $u \le u_s \le u_h$ (double/single/half).

```
1 Q = A, R = I
2 for iter = 1:4
3     R_h = LU-Chol(Q, u_h) in precision u_h
4     Update R ← R_h R in precision u
5     if estimate of κ_2(Q) < c u_h^{-1}, break,end
6     Q = A R^{-1} in precision u_s if iter=1 and u otherwise
7 end
```

- Return R such that $\kappa_2(AR^{-1}) \approx 1$.
- No knowledge of $\kappa_2(A)$ is required.
- Estimate of $\kappa_2(Q)$ relies on R_h and is cheap to compute.
- Expensive step 3 performed at half precision.

Experiment 2, $m = 1000$, $n = 10$

$u = 1.11 \times 10^{-16}$, $u_s = 5.96 \times 10^{-8}$, $u_h = 4.88 \times 10^{-4}$.
res $= \|A - QR\|_2 / \|A\|_2$. A = randsvd matrix.

$\kappa_2(A)$	iter	$\kappa_2(A\widetilde{R}^{-1})$	$\|I - Q^T Q\|_2$	res
1.0e+2	1	1.2	4.9e-16	1.7e-16
1.0e+3	1	1.4	8.9e-16	1.6e-16
1.0e+4	2	1.3	4.5e-16	1.7e-16
1.0e+5	2	1.2	2.6e-16	1.9e-16
1.0e+6	2	1.6	4.5e-16	1.6e-16
1.0e+7	2	1.4	5.9e-16	1.4e-16
1.0e+8	2	2.8	9.0e-16	1.2e-16
1.0e+9	3	1.3	7.8e-16	1.3e-16
1.0e+10	3	1.5	4.7e-16	1.4e-16
1.0e+12	4	1.5	4.5e-16	1.3e-16
1.0e+13	4	1.3	6.7e-16	1.3e-16

Summary

- Cholesky-QR requires a preconditioner $\widetilde{R}$ s.t. $\kappa(A\widetilde{R}^{-1}) \approx 1$.
- Propose to use mixed-precision to compute an LU-CholeskyQR preconditioner (expected to be 4 times faster than ShiftedCholeskyQR3 when $\kappa(A) \lesssim 10^4$).

Future work:

- Need a good estimate of $\kappa(A)$ for practical implementations of preconditioned LU-Cholesky-QR.
- Random butterfly transformations to avoid pivoting in LU.
- Use of preconditioned LU-Cholesky-QR in reduction to tridiagonal form of symmetric A.