Exploiting multiprecision in Krylov subspace methods

David Titley-Peloquin
McGill University, Montreal, Canada

Joint work with S. Gratton, E. Simon, and P. Toint

SIAM CSE 2019

Outline

- Motivation
- Multiprecision in GMRES
- Numerical experiments
- Conclusions

Outline

- Motivation
- Multiprecision in GMRES
- Numerical experiments
- Conclusions

Why multiprecision?

Why multiprecision?

siam news

MORE AT SIAM

SUBSCRIBE

HOME | HAPPENING NOW | GET INVOLVED | RESEARCH | CAREERS | CURRENT ISSUE

SIAM NEWS OCTOBER 2017

Research | October 02, 2017

Print

A Multiprecision World

By Nicholas Higham

Traditionally, floating-point arithmetic has come in two precisions: single and double. But with the introduction of support for other precisions, thanks in part to the influence of applications, the floating-point landscape has become much richer in recent years.

To see how today's multiprecision world came about, we need to start with two important events from the 1980s. The IEEE standard for

Why multiprecision?

Paraphrasing [Higham, 2017]:

- Variable precision is becoming more and more accessible in hardware and software.
- Using lower precision can drastically reduce computational running time (e.g. IEEE single up to 14 times faster than IEEE double).
- Our challenge is to better understand the accuracy of algorithms in low precision.

Why multiprecision?

Paraphrasing [Higham, 2017]:

- Variable precision is becoming more and more accessible in hardware and software.
- Using lower precision can drastically reduce computational running time (e.g. IEEE single up to 14 times faster than IEEE double).
- Our challenge is to better understand the accuracy of algorithms in low precision.

How does multiprecision arithmetic affect the convergence rate and final accuracy of Krylov subspace methods?

Outline

- Motivation
- Multiprecision in GMRES
- Numerical experiments
- Conclusions

Arnoldi algorithm

$\beta = \sqrt{\langle b, b\rangle}$
$v_1 = b/\beta$
for $k = 1, 2, \ldots$ **do**
  $w_k = Av_k$
  for $j = 1, \ldots, k$ **do**
    $h_{jk} = \langle v_j, w_k\rangle$
    $w_k = w_k - h_{jk} v_j$
  end for
  $h_{k+1,k} = \sqrt{\langle w_k, w_k\rangle}$
  $v_{k+1} = w_k / h_{k+1,k}$
end for

This is equivalent to MGS applied to $[b, A]$.

Arnoldi algorithm

$\beta = \sqrt{\langle b, b \rangle}$
$v_1 = b/\beta$
for $k = 1, 2, \ldots$ **do**
  $w_k = Av_k$
  for $j = 1, \ldots, k$ **do**
    $h_{jk} = \langle v_j, w_k \rangle$
    $w_k = w_k - h_{jk} v_j$
  end for
  $h_{k+1,k} = \sqrt{\langle w_k, w_k \rangle}$
  $v_{k+1} = w_k / h_{k+1,k}$
end for

This is equivalent to MGS applied to $[b, A]$.

After k steps, the algorithm has produced $V_{k+1} \in \mathbb{R}^{n \times (k+1)}$ and $H_k \in \mathbb{R}^{(k+1) \times k}$ upper-Hessenberg such that

$$AV_k = V_{k+1} H_k, \quad V_k^T V_k = I.$$

Arnoldi algorithm

$\beta = \sqrt{\langle b, b \rangle}$
$v_1 = b/\beta$
for $k = 1, 2, \ldots$ **do**
  $w_k = Av_k$
  for $j = 1, \ldots, k$ **do**
    $h_{jk} = \langle v_j, w_k \rangle$
    $w_k = w_k - h_{jk} v_j$
  end for
  $h_{k+1,k} = \sqrt{\langle w_k, w_k \rangle}$
  $v_{k+1} = w_k / h_{k+1,k}$
end for

This is equivalent to MGS applied to $[b, A]$.

After k steps, the algorithm has produced $V_{k+1} \in \mathbb{R}^{n\times(k+1)}$ and $H_k \in \mathbb{R}^{(k+1)\times k}$ upper-Hessenberg such that

$$AV_k = V_{k+1} H_k, \quad V_k^T V_k = I.$$

In **GMRES**, $x_k \in \mathrm{Range}(V_k) = \mathrm{Span}\{b, Ab, \ldots, A^{k-1}b\} = \mathcal{K}_k(A, b)$
is chosen to minimize the residual norm $\|b - Ax_k\|_2$ over the Krylov subspace.

Inexact Arnoldi

- Arnoldi with inexact matvecs:

$$AV_k + E_k = V_{k+1} H_k, \quad V_k^T V_k = I$$

- Arnoldi with inexact inner products:

$$AV_k = V_{k+1} H_k, \quad V_k^T V_k = I + F_k$$

Inexact Arnoldi

- Arnoldi with inexact matvecs:

$$AV_k + E_k = V_{k+1}H_k, \quad V_k^T V_k = I$$

- Arnoldi with inexact inner products:

$$AV_k = V_{k+1}H_k, \quad V_k^T V_k = I + F_k$$

- Our goal is to analyze GMRES in inexact arithmetic, with the floating point precision varying at each iteration:

$$AV_k + E_k = V_{k+1}H_k, \quad V_k^T V_k = I + F_k$$

Pieces of the puzzle

- GMRES with inexact matrix-vector products
  [Bouras & Fraysse, 2000], [Bouras, Fraysse & Giraud, 2000],
  [Sleijpen & van den Eshof, 2002], [Simoncini & Szyld, 2003],
  [Giraud, Gratton & Langou, 2007]

Pieces of the puzzle

- GMRES with inexact matrix-vector products
  [Bouras & Fraysse, 2000], [Bouras, Fraysse & Giraud, 2000],
  [Sleijpen & van den Eshof, 2002], [Simoncini & Szyld, 2003],
  [Giraud, Gratton & Langou, 2007]
- convergence of GMRES in IEEE double floating point arithmetic
  [Drkosova, Greenbaum, Rozloznik & Strakos, 1995],
  [Greenbaum, Rozloznik & Strakos, 1997], [Paige, Rozloznik & Strakos, 2006]
- loss of orthogonality in MGS
  [Bjorck, 1967], [Bjorck & Paige, 1992]

Pieces of the puzzle

- GMRES with inexact matrix-vector products
  [Bouras & Fraysse, 2000], [Bouras, Fraysse & Giraud, 2000], [Sleijpen & van den Eshof, 2002], [Simoncini & Szyld, 2003], [Giraud, Gratton & Langou, 2007]
- convergence of GMRES in IEEE double floating point arithmetic
  [Drkosova, Greenbaum, Rozloznik & Strakos, 1995], [Greenbaum, Rozloznik & Strakos, 1997], [Paige, Rozloznik & Strakos, 2006]
- loss of orthogonality in MGS
  [Bjorck, 1967], [Bjorck & Paige, 1992]
- GMRES in non-standard inner products
  [Pestana & Wathen, 2013], [Guttel & Pestana, 2014]

Dealing with loss of orthogonality

Suppose $Q \in \mathbb{R}^{n \times k}$ has rank k. If

$$Q^T Q = I_k - F, \qquad \|F\|_2 \leq \delta < 1,$$

then there exists a SPD matrix $M \in \mathbb{R}^{n \times n}$ such that

$$Q^T (I_n + M) Q = I_k.$$

Dealing with loss of orthogonality

Suppose $Q \in \mathbb{R}^{n\times k}$ has rank k. If

$$Q^T Q = I_k - F, \qquad \|F\|_2 \leq \delta < 1,$$

then there exists a SPD matrix $M \in \mathbb{R}^{n\times n}$ such that

$$Q^T (I_n + M) Q = I_k.$$

Additionally,

$$\kappa_2(I_n + M) \leq \frac{(1+\delta)^2}{(1-\delta)^2}.$$

Even if δ is quite large, $I_n + M$ remains well conditioned,

$$\text{e.g.} \;\; \delta = 1/2 \;\Rightarrow\; \kappa_2(I_n + M) \leq 9.$$

Dealing with loss of orthogonality

If after k steps of Arnoldi with inexact inner products

$$AV_k = V_{k+1}H_k, \qquad V_k^T V_k = I_k - F, \qquad \|F\|_2 \leq \delta < 1,$$

then

$$AV_k = V_{k+1}H_k, \qquad V_k^T (I_n + M) V_k = I_k.$$

Dealing with loss of orthogonality

If after k steps of Arnoldi with inexact inner products

$$AV_k = V_{k+1}H_k, \qquad V_k^T V_k = I_k - F, \qquad \|F\|_2 \leq \delta < 1,$$

then

$$AV_k = V_{k+1}H_k, \qquad V_k^T(I_n + M)V_k = I_k.$$

- The Arnoldi algorithm with inexact inner products has exactly computed an $(I_n + M)$-orthonormal basis for $\mathcal{K}_k(A, b)$.
- The resulting inexact implementation of GMRES is equivalent to exact GMRES in the $(I_n + M)$ inner product.

Dealing with loss of orthogonality

Let r_k denote the residual of exact GMRES and $\tilde{r}_k$ the residual of GMRES implemented with inexact inner products.

If after k steps of Arnoldi with inexact inner products

$$AV_k = V_{k+1}H_k, \qquad V_k^T V_k = I_k - F, \qquad \|F\|_2 \leq \delta < 1,$$

then

$$1 \leq \frac{\|\tilde{r}_k\|_2}{\|r_k\|_2} \leq \frac{(1+\delta)^2}{(1-\delta)^2}.$$

Dealing with loss of orthogonality

Let r_k denote the residual of exact GMRES and $\tilde{r}_k$ the residual of GMRES implemented with inexact inner products.

If after k steps of Arnoldi with inexact inner products

$$AV_k = V_{k+1}H_k, \qquad V_k^T V_k = I_k - F, \qquad \|F\|_2 \leq \delta < 1,$$

then

$$1 \leq \frac{\|\tilde{r}_k\|_2}{\|r_k\|_2} \leq \frac{(1+\delta)^2}{(1-\delta)^2}.$$

Provided there exists such a δ not too close to 1, the inexact inner products do not significantly affect the convergence rate or final achievable accuracy of GMRES.

Bounding $\|F_k\|_2$

In the Arnoldi algorithm, use η_{jk} to denote the error in each inner product:

$$h_{jk} = v_j^T w_k + \eta_{jk}.$$

Let

$$N_k = \begin{bmatrix} \eta_{11} & \eta_{12} & \dots & \eta_{1k} \\ & \eta_{22} & \dots & \eta_{2k} \\ & & \ddots & \vdots \\ & & & \eta_{kk} \end{bmatrix}, \quad R_k = \begin{bmatrix} h_{21} & h_{22} & \dots & h_{2,k} \\ & h_{32} & \dots & h_{3,k} \\ & & \ddots & \vdots \\ & & & h_{k+1,k} \end{bmatrix}.$$

Bounding $\|F_k\|_2$

In the Arnoldi algorithm, use η_{jk} to denote the error in each inner product:

$$h_{jk} = v_j^T w_k + \eta_{jk}.$$

Let

$$N_k = \begin{bmatrix} \eta_{11} & \eta_{12} & \dots & \eta_{1k} \\ & \eta_{22} & \dots & \eta_{2k} \\ & & \ddots & \vdots \\ & & & \eta_{kk} \end{bmatrix}, \quad R_k = \begin{bmatrix} h_{21} & h_{22} & \dots & h_{2,k} \\ & h_{32} & \dots & h_{3,k} \\ & & \ddots & \vdots \\ & & & h_{k+1,k} \end{bmatrix}.$$

After k steps of Arnoldi with the above inexact inner products,

$$AV_k = V_{k+1} H_k, \qquad V_k^T V_k = I_k - F_k,$$

with

$$\|F_k\|_2 \leq 2\|N_k R_k^{-1}\|_2 \equiv \delta.$$

Bounding $\|F_k\|_2$

For any $\epsilon \in (0,1)$, if at all steps $j = 1, \ldots, k$ the inner product h_{ij} is computed with error

$$\eta_j \leq \frac{\epsilon \delta \sigma_{min}(A)}{2} \frac{\|A\|_2 \|x_{j-1}\|_2 + \|b\|_2}{\|r_{j-1}\|_2}$$

then either GMRES has converged to a backward error of ϵ at step $k-1$, or $\|F_k\|_2 \leq \delta$.

Bounding $\|F_k\|_2$

For any $\epsilon \in (0,1)$, if at all steps $j = 1, \ldots, k$ the inner product h_{ij} is computed with error

$$\eta_j \leq \frac{\epsilon \delta \sigma_{min}(A)}{2} \frac{\|A\|_2 \|x_{j-1}\|_2 + \|b\|_2}{\|r_{j-1}\|_2}$$

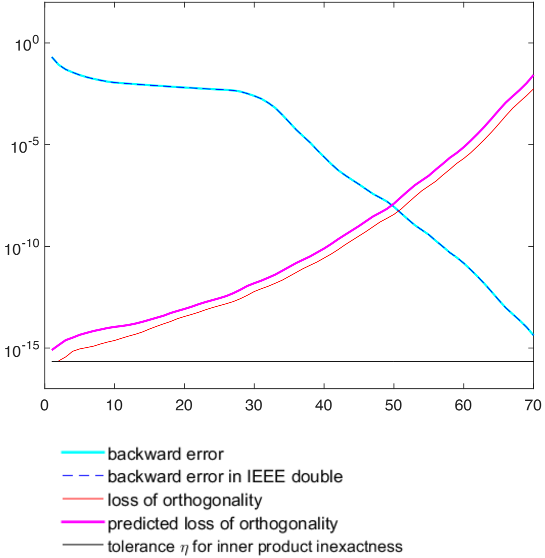
then either GMRES has converged to a backward error of ϵ at step $k-1$, or $\|F_k\|_2 \leq \delta$.

- This result is similar to the one in [Simoncini & Szyld, 2003] for GMRES with inexact matvecs.
- Because the residual norm is decreasing, the threshold increases as the iterations proceed.
- The $\sigma_{\min}(A)$ seems overly pessimistic, but we haven't (yet) been able to remove it from the bound.

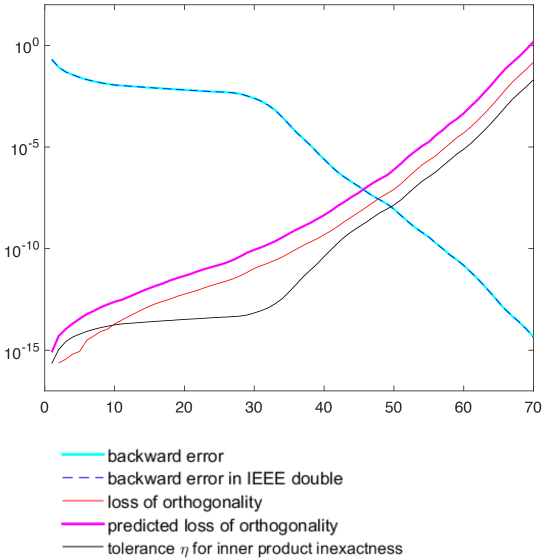
Outline

- Motivation
- Multiprecision in GMRES
- Numerical experiments
- Conclusions

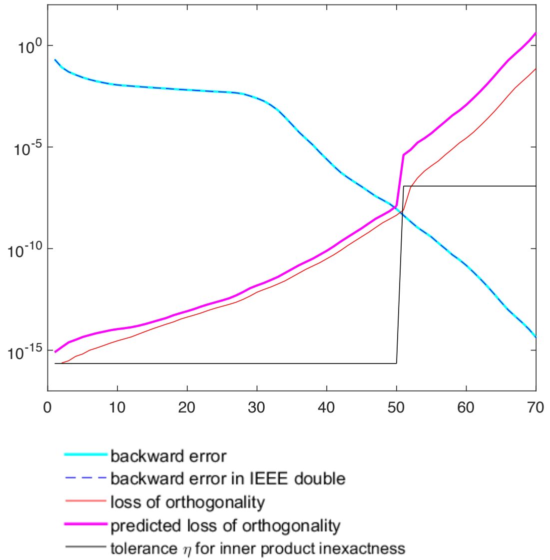
GMRES in IEEE double



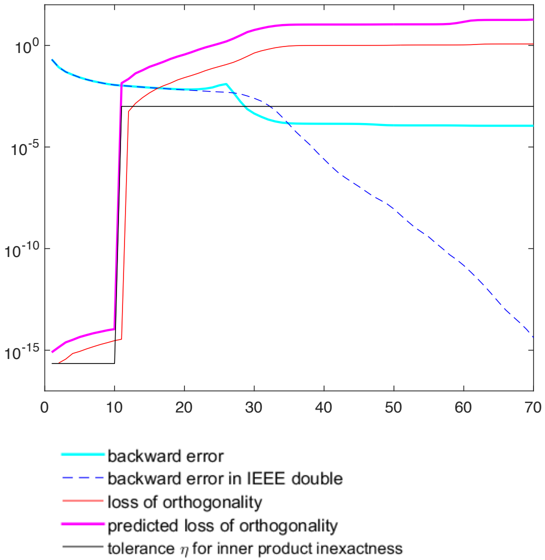
GMRES with inexact inner products (1)



GMRES with inexact inner products (2)



GMRES with inexact inner products (3)



Outline

- Motivation
- Multiprecision in GMRES
- Numerical experiments
- Conclusions

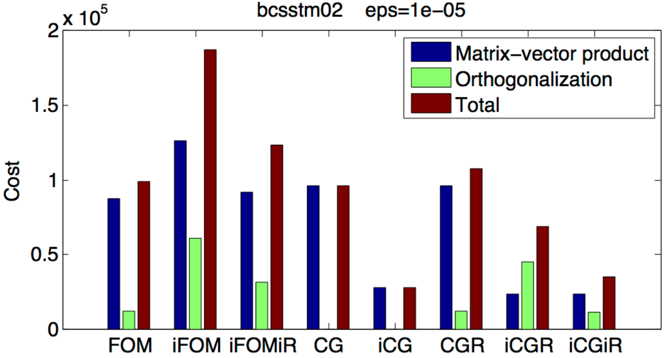
Conclusions

- Multiprecision arithmetic can be used in GMRES for computing both matvecs and inner products, without affecting the convergence rate or final achievable accuracy.
- The inner products at step j can be computed inexactly with error

$$h_{ij} = v_i^T w_j + \eta_j, \qquad \eta_j \leq \epsilon \sigma_{min}(A) \frac{\|A\|_2 \|x_{j-1}\|_2 + \|b\|_2}{\|r_{j-1}\|_2}$$

- Because the residual norm is decreasing, this threshold *increases* as the iterations proceed.
- The $\sigma_{\min}(A)$ seems overly pessimistic, but we haven't (yet) been able to remove it from the bound.
- The full result for inexact matvecs, saxpys & inner products is work in progress.

Extensions to CG/Lanczos?



Extensions to CG/Lanczos?

method	Matrix	n_{it}	cost	r.res.gap	r.sol.err	r.val.err.	Matrix	n_{it}	cost	r.res.gap	r.sol.err	r.val.err.
FOM	*bcsstm02.mat*	9	9.0e+00	4.9e-32	2.2e-06	4.2e-16	*nos4.mat*	53	5.3e+01	2.6e-28	2.3e-06	1.9e-15
iFOM		21	1.4e+01	9.0e-32	5.1e-07	7.0e-05		63	5.2e+01	9.0e-29	2.7e-09	8.9e-09
CG		10	1.0e+01	2.4e-32	4.9e-07	0.0e+00		59	5.9e+01	3.2e-29	5.4e-08	1.9e-08
CGR		10	1.0e+01	1.1e-31	4.9e-07	0.0e+00		59	5.9e+01	1.4e-28	5.4e-08	3.0e-15
iCG		26	2.2e+00	7.1e-07	7.1e-07	1.8e-04		64	1.7e+01	1.5e-10	4.2e-09	8.6e-06
iCGR		19	1.8e+00	4.9e-07	4.9e-07	3.0e-05		63	1.6e+01	2.0e-09	4.6e-09	1.5e-05
FOM	*bcsstk09.mat*	153	1.5e+02	8.6e-27	2.5e-06	1.4e-13	*bcsstk05.mat*	119	1.2e+02	2.5e-27	2.2e-06	5.1e-13
iFOM		152	4.3e+01	4.4e-27	2.6e-06	8.7e-13		129	2.6e+01	4.0e-27	2.7e-09	9.1e-11
CG		80	8.0e+01	4.3e-27	6.1e-04	9.7e-14		113	1.1e+02	3.8e-28	7.8e-05	4.0e-07
CGR		80	8.0e+01	3.5e-27	6.1e-04	1.0e-14		89	8.9e+01	8.5e-28	6.9e-05	4.4e-15
iCG		152	1.1e+01	8.1e-19	2.7e-06	4.2e-09		179	1.2e+01	2.5e-15	1.0e-05	3.2e-10
iCGR		152	1.1e+01	4.5e-18	2.7e-06	3.2e-10		129	8.6e+00	2.7e-13	2.7e-09	5.8e-08
FOM	*bcsstk27.mat*	302	3.0e+02	5.6e-27	2.4e-06	4.5e-13	685_*bus.mat*	182	1.8e+02	7.4e-26	2.3e-06	3.6e-12
iFOM		293	2.7e+02	4.6e-27	4.1e-06	1.4e-13		188	1.2e+02	3.9e-26	1.0e-06	2.5e-11
CG		305	3.0e+02	1.8e-29	1.0e-04	2.5e-07		322	3.2e+02	4.8e-27	7.0e-07	4.4e-09
CGR		235	2.4e+02	1.9e-27	1.0e-04	2.3e-15		191	1.9e+02	1.7e-26	6.5e-07	3.2e-14
iCG		395	9.4e+01	2.5e-17	7.6e-06	6.4e-08		370	7.1e+01	2.6e-13	5.0e-06	3.4e-07
iCGR		293	6.8e+01	2.7e-17	4.1e-06	3.5e-10		188	2.9e+01	7.6e-12	1.0e-06	8.7e-07
FOM	*nos1.mat*	220	2.2e+02	4.7e-22	2.1e-06	1.4e-10	*nos7.mat*	270	2.7e+02	1.0e-18	1.8e-06	2.2e-08
iFOM		230	1.9e+02	2.0e-21	9.0e-09	1.4e-10		252	2.5e+02	4.2e-18	1.4e-05	2.0e-08
CG		711	7.1e+02	3.6e-23	3.1e-01	6.8e-07		2102	2.1e+03	1.5e-22	1.7e-07	2.7e-08
CGR		199	2.0e+02	1.1e-23	5.6e-05	1.7e-12		300	3.0e+02	1.2e-18	9.6e-08	1.0e-09
iCG		711	1.7e+02	1.1e-18	3.6e-01	7.7e-07		1097	1.5e+02	1.1e-07	1.0e-02	1.5e-04
iCGR		230	4.5e+01	2.4e-16	9.0e-09	6.4e-09		274	3.4e+01	1.7e-06	6.4e-06	9.9e-05

Table 4.15: NIST Matrix Market: using practical algorithms in multi-precision with $\epsilon = 10^{-5}$