



# Backward stability of FGMRES

(i.e. Using FGMRES to recover backward stability in mixed precision)

M. Arioli, I. S. Duff,

<http://www.numerical.rl.ac.uk/people/marioli/marioli.html>



# Outline

- Mixed precision
- Iterative refinement, GMRES and Flexible GMRES
- Roundoff error analysis
- Numerical experiments



# Linear system

We wish to solve large sparse systems

$$Ax = b \quad A \in \mathbf{R}^{N \times N}$$

to **high accuracy** using **mixed precision** arithmetic. For example, we might want to achieve double precision accuracy while using a single precision factorization of the matrix  $A$ . We will use this lower accuracy factorization as a preconditioner for **FGMRES**.



# Mixed precision arithmetic

- Very fast 32-bit arithmetic unit



# Mixed precision arithmetic

- Very fast 32-bit arithmetic unit
- We use 32-bit arithmetic for factorization and triangular solves  
 $M$  is the  $fl(LU)$  of  $A$  and  $\|M - A\| \leq c(N)\sqrt{\varepsilon}\|A\|$   
( $\varepsilon = 2.2 \times 10^{-16}$ )



# Mixed precision arithmetic

- Very fast 32-bit arithmetic unit
- We use 32-bit arithmetic for factorization and triangular solves  
 $M$  is the  $fl(LU)$  of  $A$  and  $\|M - A\| \leq c(N)\sqrt{\varepsilon}\|A\|$   
( $\varepsilon = 2.2 \times 10^{-16}$ )
- If  $\kappa(A)\sqrt{\varepsilon} > 1$  then Iterative Refinement may not converge. **FGMRES**  
**does**



# Mixed precision arithmetic

- Very fast 32-bit arithmetic unit
- We use 32-bit arithmetic for factorization and triangular solves  
 $M$  is the  $fl(LU)$  of  $A$  and  $\|M - A\| \leq c(N)\sqrt{\varepsilon}\|A\|$   
( $\varepsilon = 2.2 \times 10^{-16}$ )
- If  $\kappa(A)\sqrt{\varepsilon} > 1$  then Iterative Refinement may not converge. **FGMRES** does
- **FGMRES** backward stable



# Mixed precision arithmetic

- Very fast 32-bit arithmetic unit
- We use 32-bit arithmetic for factorization and triangular solves  
 $M$  is the  $fl(LU)$  of  $A$  and  $\|M - A\| \leq c(N)\sqrt{\varepsilon}\|A\|$   
( $\varepsilon = 2.2 \times 10^{-16}$ )
- If  $\kappa(A)\sqrt{\varepsilon} > 1$  then Iterative Refinement may not converge. **FGMRES** does
- **FGMRES** backward stable
- **GMRES** is not always backward stable



# GMRES and FGMRES

Let  $r_0 = b - Ax_0$  and  $\mathcal{K}_k(A, r_0)$  be the usual Krylov space  
GMRES

$$\min_{x \in x_0 + \mathcal{K}_k(A, r_0)} \|r_0 - Ax\|_2 \quad r_0 - Ax_k \perp A\mathcal{K}_k(A, r_0)$$



# GMRES and FGMRES

Let  $r_0 = b - Ax_0$  and  $\mathcal{K}_k(A, r_0)$  be the usual Krylov space  
GMRES

$$\min_{x \in x_0 + \mathcal{K}_k(A, r_0)} \|r_0 - Ax\|_2 \quad r_0 - Ax_k \perp A\mathcal{K}_k(A, r_0)$$

GMRES Right preconditioning

$$AM^{-1}y = b \quad \left\{ \begin{array}{l} (AM^{-1}, r_0) \longrightarrow (A, r_0) \\ \mathcal{K}_k(AM^{-1}, r_0) \longrightarrow \mathcal{K}_k(A, r_0) \\ x_k = M^{-1}y_k \\ AM^{-1}V_k = V_{k+1}H_k \end{array} \right.$$



# GMRES and FGMRES

Let  $r_0 = b - Ax_0$  and  $\mathcal{K}_k(A, r_0)$  be the usual Krylov space  
GMRES

$$\min_{x \in x_0 + \mathcal{K}_k(A, r_0)} \|r_0 - Ax\|_2 \quad r_0 - Ax_k \perp A\mathcal{K}_k(A, r_0)$$

GMRES Right preconditioning

$$AM^{-1}y = b \quad \left\{ \begin{array}{l} (AM^{-1}, r_0) \longrightarrow (A, r_0) \\ \mathcal{K}_k(AM^{-1}, r_0) \longrightarrow \mathcal{K}_k(A, r_0) \\ x_k = M^{-1}y_k \\ AM^{-1}V_k = V_{k+1}H_k \end{array} \right.$$

Flexible GMRES Right preconditioning

$$Z_k \longrightarrow \mathcal{K}_k(A, r_0) \quad x_k = x_0 + Z_k y_k \quad AZ_k = V_{k+1}H_k$$

$$Z_k = \text{span}(r_0, AM_1^{-1}r_0, \dots, \left( \prod_{j=0}^{k-1} AM_j^{-1} \right) r_0)$$



# Right preconditioned GMRES and Flexible GMRES

```

procedure [x] = right_Prec_GMRES(A,M,b)
   $x_0 = M^{-1}b$ ,  $r_0 = b - Ax_0$  and  $\beta = \|r_0\|$ 
   $v_1 = r_0/\beta$ ;  $k=0$ ;
  while  $\|r_k\| > \mu(\|b\| + \|A\| \|x_k\|)$ 
     $k = k + 1$ ;
     $z_k = M^{-1}v_k$ ;  $w = Az_k$ ;
    for  $i = 1, \dots, k$  do
       $h_{i,k} = v_i^T w$ ;
       $w = w - h_{i,k}v_i$ ;
    end for;
     $h_{k+1,k} = \|w\|$ ;
     $v_{k+1} = w/h_{k+1,k}$ ;
     $V_k = [v_1, \dots, v_k]$ ;
     $H_k = \{h_{i,j}\}_{1 \leq i \leq j+1; 1 \leq j \leq k}$ ;
     $y_k = \arg \min_y \|\beta e_1 - H_k y\|$ ;
     $x_k = x_0 + M^{-1}V_k y_k$  and  $r_k = b - Ax_k$ ;
  end while ;
end procedure.

```

```

procedure [x] =FGMRES(A,M_i,b)
   $x_0 = M_0^{-1}b$ ,  $r_0 = b - Ax_0$  and  $\beta = \|r_0\|$ 
   $v_1 = r_0/\beta$ ;  $k=0$ ;
  while  $\|r_k\| > \mu(\|b\| + \|A\| \|x_k\|)$ 
     $k = k + 1$ ;
     $z_k = M_k^{-1}v_k$ ;  $w = Az_k$ ;
    for  $i = 1, \dots, k$  do
       $h_{i,k} = v_i^T w$ ;
       $w = w - h_{i,k}v_i$ ;
    end for;
     $h_{k+1,k} = \|w\|$ ;
     $v_{k+1} = w/h_{k+1,k}$ ;
     $Z_k = [z_1, \dots, z_k]$ ;  $V_k = [v_1, \dots, v_k]$ ;
     $H_k = \{h_{i,j}\}_{1 \leq i \leq j+1; 1 \leq j \leq k}$ ;
     $y_k = \arg \min_y \|\beta e_1 - H_k y\|$ ;
     $x_k = x_0 + Z_k y_k$  and  $r_k = b - Ax_k$ ;
  end while ;
end procedure.

```



# Roundoff error

The roundoff error analysis of both FGMRES and GMRES can be done in three stages:



# Roundoff error

The **roundoff error analysis** of both **FGMRES** and **GMRES** can be done in **three stages**:

1. Error analysis of the Arnoldi-Krylov process (Giraud and Langou, Björck and Paige, and Paige, Rozložník, and Strakoš).  
MGS applied to

$$z_1 = M_1^{-1}r_0/\|r_0\|, \quad z_j = M_j^{-1}v_j$$

$$C^{(k)} = (r_0, Az_1, Az_2, \dots, Az_k) = V_{k+1}R_k$$

$$R_k = \begin{bmatrix} \|r_0\| & H_{1,1} & \dots & H_{1,k} \\ 0 & H_{2,1} & \dots & H_{2,k} \\ 0 & 0 & \dots & H_{3,k} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & H_{k+1,k} \end{bmatrix}$$



# Roundoff error

The **roundoff error analysis** of both **FGMRES** and **GMRES** can be done in **three stages**:

1. Error analysis of the Arnoldi-Krylov process (Giraud and Langou, Björck and Paige, and Paige, Rozložník, and Strakoš).
2. Error analysis of the **Givens process used on the upper Hessenberg matrix**  $H_k$  in order to reduce it to upper triangular form.



# Roundoff error

The **roundoff error analysis** of both **FGMRES** and **GMRES** can be done in **three stages**:

1. Error analysis of the Arnoldi-Krylov process (Giraud and Langou, Björck and Paige, and Paige, Rozložník, and Strakoš).
2. Error analysis of the **Givens process used on the upper Hessenberg matrix**  $H_k$  in order to reduce it to upper triangular form.
3. Error analysis of the computation of  $x_k$  in FGMRES and GMRES.



# Roundoff error

The **roundoff error analysis** of both **FGMRES** and **GMRES** can be done in **three stages**:

1. Error analysis of the Arnoldi-Krylov process (Giraud and Langou, Björck and Paige, and Paige, Rozložník, and Strakoš).
2. Error analysis of the **Givens process used on the upper Hessenberg matrix**  $H_k$  in order to reduce it to upper triangular form.
3. Error analysis of the computation of  $x_k$  in FGMRES and GMRES.

The first two stages of the roundoff error analysis are the same for both FGMRES and GMRES. The **last stage is specific** to each algorithm.



# Roundoff error analysis of FGMRES

**Theorem 1.** *If we apply FGMRES to solve  $Ax = b$ , using finite-precision arithmetic conforming to IEEE standard with relative precision  $\varepsilon$  and under the hypotheses:*

$$2.12(n + 1)\varepsilon < 0.01 \quad \text{and} \quad c_0(n)\varepsilon\kappa(C^{(k)}) < 0.1 \quad \forall k$$

where

$$c_0(n) = 18.53n^{\frac{3}{2}}$$

and

$$|\bar{s}_k| < 1 - \varepsilon, \quad \forall k,$$

where  $\bar{s}_k$  are the sines computed during the Givens algorithm applied to  $\bar{H}_k$  in order to compute  $\bar{y}_k$ , then there exists  $\hat{k}$ ,  $\hat{k} \leq n$  such that,  $\forall k \geq \hat{k}$ , we have

$$\|b - A\bar{x}_k\| \leq c_1(n, k)\varepsilon \left( \|b\| + \|A\| \|\bar{x}_0\| + \right. \\ \left. \|A\| \|\bar{Z}_k\| \|\bar{y}_k\| + \|A\bar{Z}_k\| \|\bar{y}_k\| \right) + \mathcal{O}(\varepsilon^2).$$



## The symmetric indefinite case

A particular and important case arises in saddle-point problems where the coefficient matrix is of the form

$$A = \begin{bmatrix} H & B \\ B^T & 0 \end{bmatrix}$$



## Why FGMRES for symmetric case?

- The computed values of Gaussian factorization  $\hat{L} \hat{D}$  are affected by roundoff:  $M = \hat{L} \hat{D} \hat{L}^T$  and  $\|E\| = \|M - A\| \leq c(n) \text{eps} \|A\|$  with  $E \neq E^T$



## Why FGMRES for symmetric case?

- The computed values of Gaussian factorization  $\hat{L} \hat{D}$  are affected by roundoff:  $M = \hat{L}\hat{D}\hat{L}^T$  and  $\|E\| = \|M - A\| \leq c(n) \text{eps}\|A\|$  with  $E \neq E^T$
- Thus  $M^{-1}A \neq AM^{-1}$  and the preconditioned matrix is non symmetric



## Why FGMRES for symmetric case?

- The computed values of Gaussian factorization  $\hat{L} \hat{D}$  are affected by roundoff:  $M = \hat{L} \hat{D} \hat{L}^T$  and  $\|E\| = \|M - A\| \leq c(n) \text{eps} \|A\|$  with  $E \neq E^T$
- Thus  $M^{-1}A \neq AM^{-1}$  and the preconditioned matrix is non symmetric
- FGMRES is then the only way



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$
- We weaken the numerical pivot strategy by using a threshold



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$
- We weaken the numerical pivot strategy by using a threshold
- However, also this can be unsatisfactory: the numerical pivot strategy is still disrupting the ordering we have chosen and increases the fill-in



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$
- We weaken the numerical pivot strategy by using a threshold
- However, also this can be unsatisfactory: the numerical pivot strategy is still disrupting the ordering we have chosen and increases the fill-in
- An **ALTERNATIVE** is to use **Static Pivoting**, by replacing the pivot  $a_k$  failing the test by  $a_k + \tau$  and CONTINUE.



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$
- We weaken the numerical pivot strategy by using a threshold
- However, also this can be unsatisfactory: the numerical pivot strategy is still disrupting the ordering we have chosen and increases the fill-in
- An **ALTERNATIVE** is to use **Static Pivoting**, by replacing the pivot  $a_k$  failing the test by  $a_k + \tau$  and CONTINUE.
- We thus have factorized  $A + E = LDL^T = M$  where  $|E| \leq \tau I$



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$
- We weaken the numerical pivot strategy by using a threshold
- However, also this can be unsatisfactory: the numerical pivot strategy is still disrupting the ordering we have chosen and increases the fill-in
- An **ALTERNATIVE** is to use **Static Pivoting**, by replacing the pivot  $a_k$  failing the test by  $a_k + \tau$  and CONTINUE.
- We thus have factorized  $A + E = LDL^T = M$  where  $|E| \leq \tau I$
- Several codes use (or have an option for) this device:



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$
- We weaken the numerical pivot strategy by using a threshold
- However, also this can be unsatisfactory: the numerical pivot strategy is still disrupting the ordering we have chosen and increases the fill-in
- An **ALTERNATIVE** is to use **Static Pivoting**, by replacing the pivot  $a_k$  failing the test by  $a_k + \tau$  and CONTINUE.
- We thus have factorized  $A + E = LDL^T = M$  where  $|E| \leq \tau I$
- Several codes use (or have an option for) this device:
  - SuperLU (Demmel and Li)



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$
- We weaken the numerical pivot strategy by using a threshold
- However, also this can be unsatisfactory: the numerical pivot strategy is still disrupting the ordering we have chosen and increases the fill-in
- An **ALTERNATIVE** is to use **Static Pivoting**, by replacing the pivot  $a_k$  failing the test by  $a_k + \tau$  and CONTINUE.
- We thus have factorized  $A + E = LDL^T = M$  where  $|E| \leq \tau I$
- Several codes use (or have an option for) this device:
  - SuperLU (Demmel and Li)
  - PARDISO (Gärtner and Schenk)



In order to reduce the fill-in during the  $LDL^T$  factorization

- We scale and reorder the entries of  $A$
- We weaken the numerical pivot strategy by using a threshold
- However, also this can be unsatisfactory: the numerical pivot strategy is still disrupting the ordering we have chosen and increases the fill-in
- An **ALTERNATIVE** is to use **Static Pivoting**, by replacing the pivot  $a_k$  failing the test by  $a_k + \tau$  and CONTINUE.
- We thus have factorized  $A + E = LDL^T = M$  where  $|E| \leq \tau I$
- Several codes use (or have an option for) this device:
  - SuperLU (Demmel and Li)
  - PARDISO (Gärtner and Schenk)
  - MA57 (Duff and Pralet)



# GMRES vs FGMRES

- GMRES error bounds depend on  $\| |\hat{L}| |\hat{D}| |\hat{L}^T| \|$ . (Arioli, Duff, Gratton, and Pralet SISC 2007)



# GMRES vs FGMRES

- GMRES error bounds depend on  $\| |\hat{L}| |\hat{D}| |\hat{L}^T| \|$ . (Arioli, Duff, Gratton, and Pralet SISC 2007)
- For sparse matrices  $\| |\hat{L}| |\hat{D}| |\hat{L}^T| \|$  can be much larger than  $\|A\|$ .



# GMRES vs FGMRES

- GMRES error bounds depend on  $|| |\hat{L}| |\hat{D}| |\hat{L}^T| ||$ . (Arioli, Duff, Gratton, and Pralet SISC 2007)
- For sparse matrices  $|| |\hat{L}| |\hat{D}| |\hat{L}^T| ||$  can be much larger than  $||A||$ .
- For the static pivot the growth can be dramatic.



# GMRES vs FGMRES

- GMRES error bounds depend on  $|| |\hat{L}| |\hat{D}| |\hat{L}^T| ||$ . (Arioli, Duff, Gratton, and Pralet SISC 2007)
- For sparse matrices  $|| |\hat{L}| |\hat{D}| |\hat{L}^T| ||$  can be much larger than  $||A||$ .
- For the static pivot the growth can be dramatic.
- Theorem 1 shows that FGMRES does not depend on  $|| |\hat{L}| |\hat{D}| |\hat{L}^T| ||$ .



# Roundoff error right preconditioned GMRES

## Theorem 2

We assume of applying Iterative Refinement for solving  $M(\bar{x}_k - \bar{x}_0) = \bar{V}_k \bar{y}_k$  at last step.

Under the Hypotheses of Theorem 1 and  $c(n)\varepsilon \kappa(M) < 1$

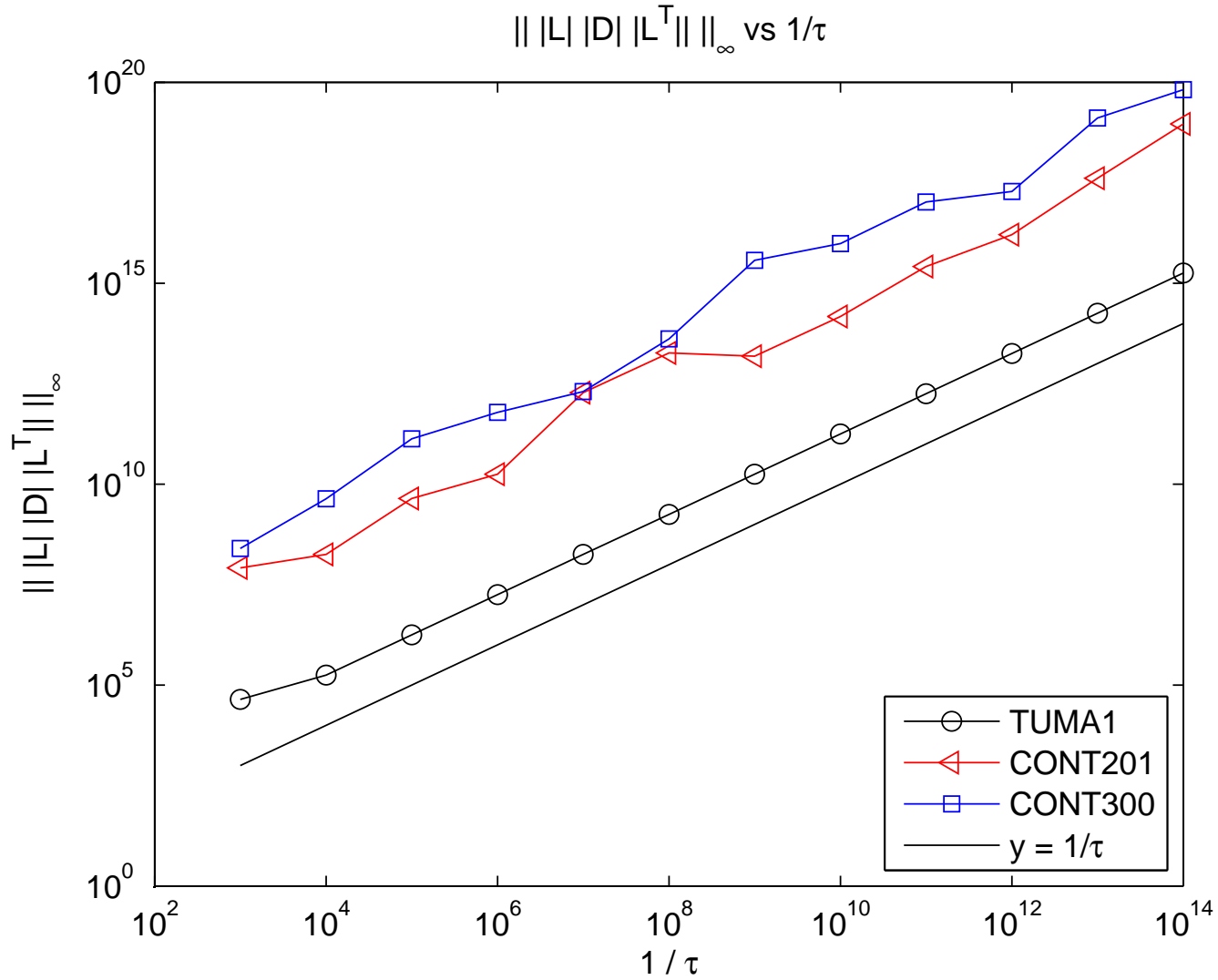
$$\exists \hat{k}, \quad \hat{k} \leq n$$

such that,  $\forall k \geq \hat{k}$ , we have

$$\|b - A\bar{x}_k\| \leq c_1(n, k)\varepsilon \left\{ \|b\| + \|A\| \|\bar{x}_0\| + \|A\| \|\bar{Z}_k\| \|M(\bar{x}_k - \bar{x}_0)\| + \right. \\ \left. \|AM^{-1}\| \|M\| \|\bar{x}_k - \bar{x}_0\| + \right. \\ \left. \|AM^{-1}\| \|\hat{L}\| \|\hat{D}\| \|\hat{L}^T\| \|M(\bar{x}_k - \bar{x}_0)\| \right\} + \mathcal{O}(\varepsilon^2).$$

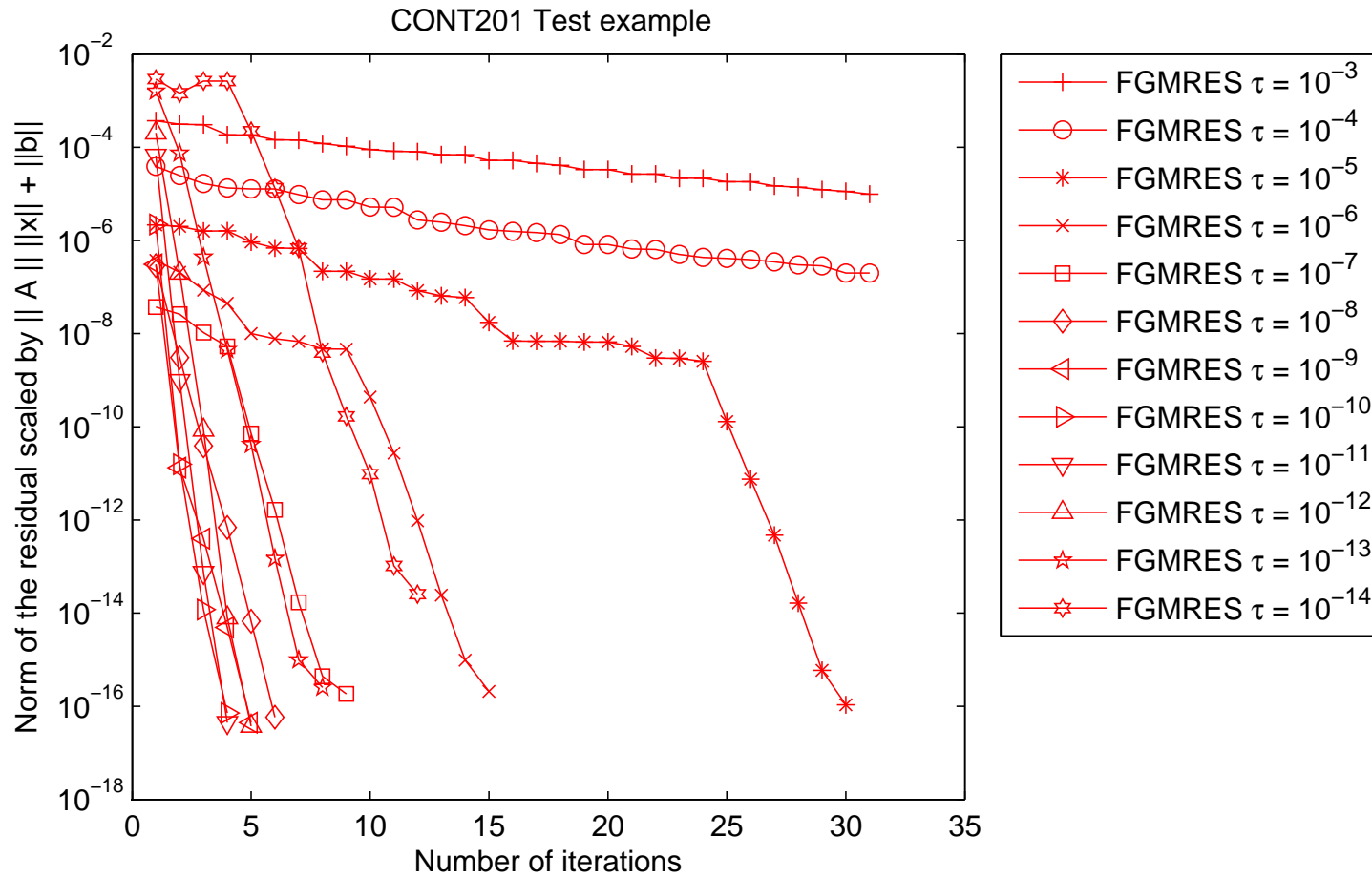


# $|| |\hat{L}| |\hat{D}| |\hat{L}^T| ||$ vs $1/\tau$





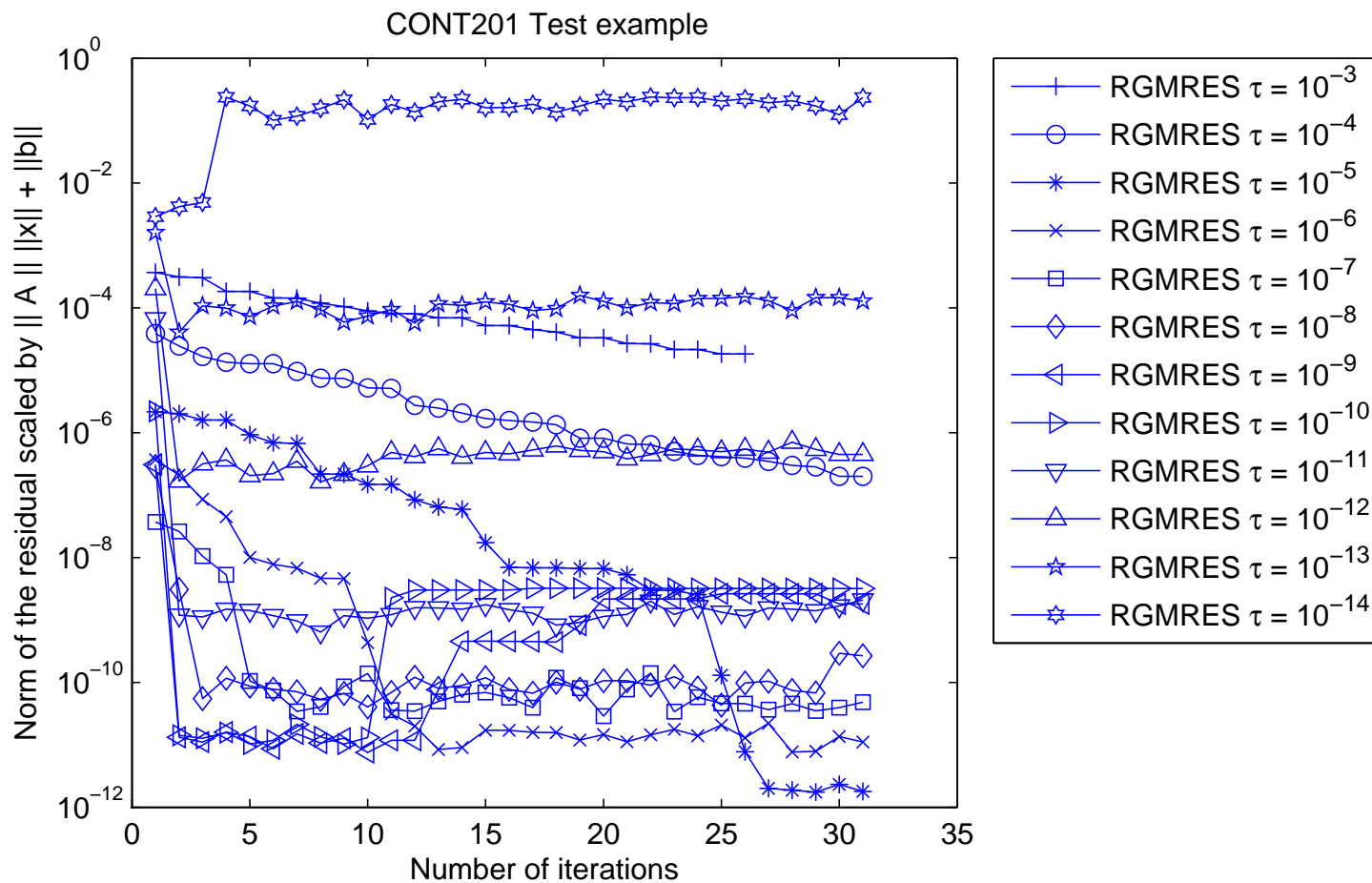
# Numerical experiments



FGMRES on CONT-201 test example



# Numerical experiments using mixed precision



GMRES on CONT-201 test example



# Test Environment

■  $A = QDW$  with  $Q$  and  $V$  random orthogonal matrices  $D = \text{diag}\{d_i\}$

$$d_i = 10^{-c\left(\frac{i-1}{n-1}\right)^\gamma}$$



## Test Environment

- $A = QDW$  with  $Q$  and  $V$  random orthogonal matrices  $D = \text{diag}\{d_i\}$

$$d_i = 10^{-c \left( \frac{i-1}{n-1} \right)^\gamma}$$

The **singular values** lie between 1 and  $10^{-c}$ , the **condition number** is  $10^c$ , and the distribution can be skewed by altering  $\gamma$ .  
 $\gamma$  equal to 1 gives a log-linear uniform distribution, values of  $\gamma$  greater than 1 skew towards 1 and values of  $\gamma$  less than 1 towards  $10^{-c}$ .



# Test Environment

- $A = QDW$  with  $Q$  and  $V$  random orthogonal matrices  $D = \text{diag}\{d_i\}$

$$d_i = 10^{-c\left(\frac{i-1}{n-1}\right)^\gamma}$$

- Selected **Sparse** Matrices



# Test Environment

- $A = QDW$  with  $Q$  and  $V$  random orthogonal matrices  $D = \text{diag}\{d_i\}$

$$d_i = 10^{-c\left(\frac{i-1}{n-1}\right)^\gamma}$$

- Selected **Sparse** Matrices
- Forward and backward substitution



# Test Environment

- $A = QDW$  with  $Q$  and  $V$  random orthogonal matrices  $D = \text{diag}\{d_i\}$

$$d_i = 10^{-c\left(\frac{i-1}{n-1}\right)^\gamma}$$

- Selected **Sparse** Matrices
- Forward and backward substitution
  - the vector  $\bar{z}_k$  is computed using the forward and backward substitution algorithm in **single precision** on the single precision conversion of vector  $\bar{v}_k$ ,
  - the vector  $\bar{z}_k$  is computed using the forward and backward substitution algorithm in **double precision** on  $\bar{v}_k$  after we converted the factors  $L$  and  $U$  to double precision.



# Random dense matrices.

$$n = 200, c = 8.2, \gamma = 1$$

Single Precision					Double Precision				
Total It	Inner it	SR	$  A \bar{z}_{\hat{k}}  $	$   \bar{z}_{\hat{k}}   \bar{y}_{\hat{k}}   $	Total It	Inner it	SR	$  A \bar{z}_{\hat{k}}  $	$   \bar{z}_{\hat{k}}   \bar{y}_{\hat{k}}   $
26	26	2.5e-16	7.4e+00	1.9e+02	20	20	1.7e-16	8.0e+00	7.3e+01
27	27	6.6e-16	4.2e+00	4.7e+02	20	20	2.0e-16	3.9e+00	5.9e+01
25	25	1.7e-16	3.3e+00	5.9e+01	20	20	2.7e-16	3.5e+00	4.0e+01
52	52	3.9e-15	4.6e+01	3.0e+03	20	20	1.1e-15	4.5e+01	4.7e+02
88	36	1.1e-16	4.6e+01	6.0e-04	25	5	1.5e-16	4.8e+01	1.5e-05
24	24	1.3e-16	2.0e+00	3.8e+01	20	20	2.6e-16	2.2e+00	2.8e+01
31	31	2.5e-16	8.8e+00	1.7e+02	20	20	1.9e-16	1.1e+01	8.4e+01
24	24	2.0e-16	3.5e+00	1.2e+02	20	20	2.0e-16	3.9e+00	6.9e+01
24	24	1.8e-16	2.7e+00	8.8e+01	20	20	6.2e-16	3.0e+00	5.8e+01
26	26	2.7e-16	3.2e+00	1.5e+02	20	20	3.2e-16	3.5e+00	3.6e+01
44	44	5.7e-16	1.9e+01	5.9e+02	20	20	4.0e-16	2.0e+01	1.6e+02

$$(SR = \frac{||b - A\bar{x}_{\hat{k}}||}{(||A|| ||\bar{x}_{\hat{k}}|| + ||b||)})$$



# MA57 sparse tests using mixed precision

Matrix Id	$n$	Iterative refinement		FGMRES				
		Total It	SR	Total It	Inner it	SR	$  A \bar{z}_{\hat{k}}  $	$   \bar{z}_{\hat{k}}      \bar{y}_{\hat{k}}   $
bcsstk20 $\kappa(A) = 5.10^9$	485	30	2.1e-15	2	2	1.4e-11	1.7e+00	4.6e+02
				4	2	3.4e-14	1.6e+00	3.8e-01
				6	2	7.2e-17	1.6e+00	5.6e-04
bcsstm27 $\kappa(A) = 5.10^9$	1224	22	1.6e-15	2	2	5.8e-11	1.7e+00	2.7e+01
				4	2	1.8e-11	6.3e-01	1.3e+00
				6	2	6.0e-13	2.0e+00	7.6e-02
				8	2	1.5e-13	1.7e+00	1.0e-02
				10	2	1.2e-14	1.7e+00	1.9e-03
				12	2	2.6e-15	1.8e+00	1.7e-04
s3rmq4m1 $\kappa(A) = 4.10^9$	5489	16	2.2e-15	2	2	3.5e-11	1.0e+00	8.6e+01
				4	2	2.1e-13	1.1e+00	3.2e-01
				6	2	4.5e-15	1.7e+00	6.4e-03
				8	2	1.1e-16	1.6e+00	1.3e-04
s3dkq4m2 $\kappa(A) = 7.10^{10}$	90449	53	1.1e-10	10	10	6.3e-17	1.2e+00	1.2e+03

Sparse matrices results ( $SR = \frac{||b - A\bar{x}_{\hat{k}}||}{(||A|| ||\bar{x}_{\hat{k}}|| + ||b||)}$ )



# Summary

■ **IR (PLAN A)** does not always work with mixed precision



# Summary

- **IR (PLAN A)** does not always work with mixed precision
- **GMRES** is also sensitive and not robust



# Summary

- **IR (PLAN A)** does not always work with mixed precision
- GMRES is also sensitive and not robust
- **FGMRES is robust and less sensitive**



# Summary

- **IR (PLAN A)** does not always work with mixed precision
- GMRES is also sensitive and not robust
- **FGMRES is robust and less sensitive**
- Gains from restarting



# Summary

- **IR (PLAN A)** does not always work with mixed precision
- GMRES is also sensitive and not robust
- **FGMRES is robust and less sensitive**
- Gains from restarting
- **PLAN B is working**