

Opportunities for Mixed Precision in Preconditioned Iterative Methods

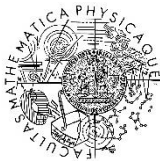
Erin Carson

Charles University

Preconditioning 2022

Chemnitz, DE

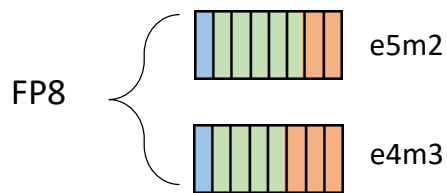
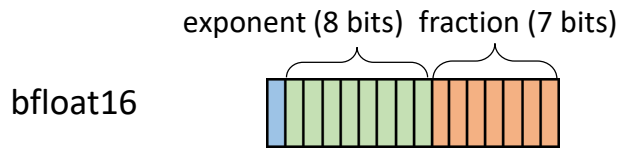
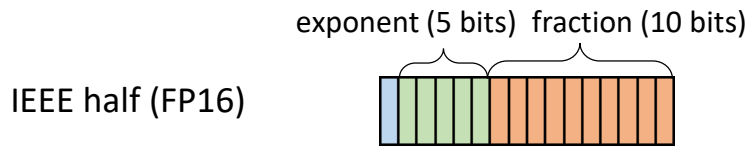
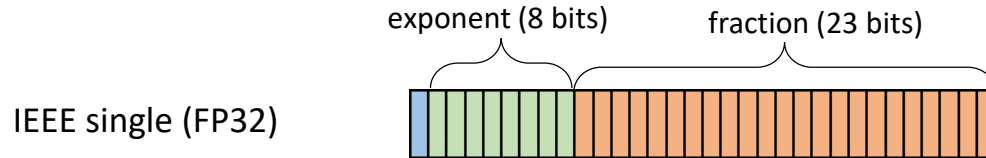
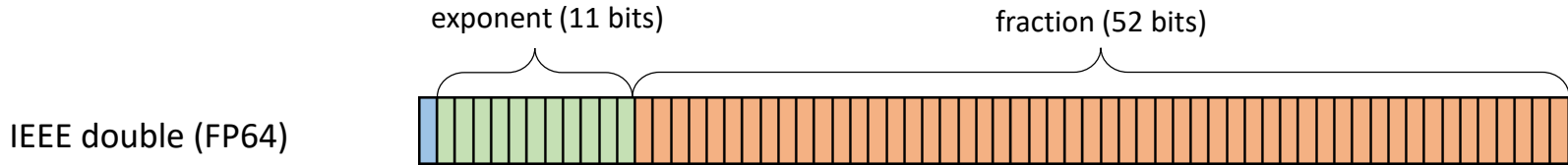
June 9, 2022



FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

Floating Point Formats

$$(-1)^{\text{sign}} \times 2^{(\text{exponent}-\text{offset})} \times 1.\text{fraction}$$



	size (bits)	range	u	perf. (NVIDIA H100)
FP64	64	$10^{\pm 308}$	1×10^{-16}	60 Tflops/s
FP32	32	$10^{\pm 38}$	6×10^{-8}	1 Pflop/s
FP16	16	$10^{\pm 5}$	5×10^{-4}	2 Pflops/s
bfloat16	16	$10^{\pm 38}$	4×10^{-3}	
FP8-e5m2	8	$10^{\pm 5}$	3×10^{-1}	4 Pflops/s
FP8-e4m3	8	$10^{\pm 2}$	1×10^{-1}	

Hardware Support for Multiprecision Computation

Use of low precision in machine learning has driven emergence of low-precision capabilities in hardware:

- Half precision (FP16) defined as storage format in 2008 IEEE standard
- [ARM NEON](#): SIMD architecture, instructions for 8x16-bit, 4x32-bit, 2x64-bit
- [AMD Radeon Instinct MI25 GPU](#), 2017:
 - single: 12.3 TFLOPS, half: 24.6 TFLOPS
- [NVIDIA Tesla P100](#), 2016: native ISA support for 16-bit FP arithmetic
- [NVIDIA Tesla V100](#), 2017: tensor cores for half precision;
 - 4x4 matrix multiply in one clock cycle
 - double: 7 TFLOPS, half+tensor: 112 TFLOPS (**16x!**)
- [Google's Tensor processing unit](#) (TPU)
- [NVIDIA A100](#), 2020: tensor cores with multiple supported precisions: FP16, FP64, Binary, INT4, INT8, bfloat16
- [NVIDIA H100](#), 2022: now with quarter-precision (FP8) tensor cores
- [Exascale supercomputers](#): Expected extensive support for reduced-precision arithmetic (Frontier: FP64, FP32, FP16, bfloat16, INT8, INT4)

Mixed precision in NLA

- **BLAS**: cuBLAS, MAGMA, [Agullo et al. 2009], [Abdelfattah et al., 2019], [Haidar et al., 2018]
- **Iterative refinement**:
 - Long history: [Wilkinson, 1963], [Moler, 1967], [Stewart, 1973], ...
 - More recently: [Langou et al., 2006], [C., Higham, 2017], [C., Higham, 2018], [C., Higham, Pranesh, 2020], [Amestoy et al., 2021]
- **Matrix factorizations**: [Haidar et al., 2017], [Haidar et al., 2018], [Haidar et al., 2020], [Abdelfattah et al., 2020]
- **Eigenvalue problems**: [Dongarra, 1982], [Dongarra, 1983], [Tisseur, 2001], [Davies et al., 2001], [Petschow et al., 2014], [Alvermann et al., 2019]
- **Sparse direct solvers**: [Buttari et al., 2008]
- **Orthogonalization**: [Yamazaki et al., 2015]
- **Multigrid**: [Tamstorf et al., 2020], [Richter et al., 2014], [Sumiyoshi et al., 2014], [Ljungkvist, Kronbichler, 2017, 2019]
- **(Preconditioned) Krylov subspace methods**: [Emans, van der Meer, 2012], [Yamagishi, Matsumura, 2016], [C., Gergelits, Yamazaki, 2021], [Clark, 2019], [Anzt et al., 2019], [Clark et al., 2010], [Gratton et al., 2020], [Arioli, Duff, 2009], [Hogg, Scott, 2010]

HPL-AI Benchmark

- Supercomputers traditionally ranked by performance on high-performance LINPACK (HPL) benchmark
 - Solves dense $Ax = b$ via Gaussian elimination with partial pivoting
- HPL-AI: Like HPL, solves dense $Ax = b$, results still to double precision accuracy
 - But achieves this via **mixed-precision** iterative refinement

HPL-AI Benchmark

June 2022

Rank	Site	Computer	Cores	HPL-AI (Eflop/s)	TOP500 Rank	HPL Rmax (Eflop/s)	Speedup
1	DOE/SC/ORNL, USA	Frontier	8,730,112	6.861	1	1.102	6.2
2	RIKEN, Japan	Fugaku	7,630,848	2.000	2	0.4420	4.5
3	DOE/SC/ORNL, USA	Summit	2,414,592	1.411	4	0.1486	9.5
4	NVIDIA, USA	Selene	555,520	0.630	8	0.0630	9.9
5	DOE/SC/LBNL, USA	Perlmutter	761,856	0.590	7	0.0709	8.3
6	FZJ, Germany	JUWELS BM	449,280	0.470	11	0.0440	10.0
7	University of Florida, USA	HiPerGator	138,880	0.170	34	0.0170	9.9
8	SberCloud, Russia	Christofari Neo	98,208	0.123	47	0.0120	10.3
9	DOE/SC/ANL, USA	Polaris	259,840	0.114	14	0.0238	4.8
10	ITC, Japan	Wisteria	368,640	0.100	20	0.0220	4.5

HPL-AI Benchmark

June 2022

Rank	Site	Computer	Cores	HPL-AI (Eflop/s)	TOP500 Rank	HPL Rmax (Eflop/s)	Speedup
1	DOE/SC/ORNL, USA	Frontier	8,730,112	6.861	1	1.102	6.2
2	RIKEN, Japan	Fugaku	7,630,848	2.000	2	0.4420	4.5
3	DOE/SC/ORNL, USA	Summit	2,414,592	1.411	4	0.1486	9.5
4	NVIDIA, USA	Selene	555,520	0.630	8	0.0630	9.9
5	DOE/SC/LBNL, USA	Perlmutter	761,856	0.590	7	0.0709	8.3
6	FZJ, Germany	JUWELS BM	449,280	0.470	11	0.0440	10.0
7	University of Florida, USA	HiPerGator	138,880	0.170	34	0.0170	9.9
8	SberCloud, Russia	Christofari Neo	98,208	0.123	47	0.0120	10.3
9	DOE/SC/ANL, USA	Polaris	259,840	0.114	14	0.0238	4.8
10	ITC, Japan	Wisteria	368,640	0.100	20	0.0220	4.5

HPL-AI Benchmark

June 2022

Rank	Site	Computer	Cores	HPL-AI (Eflop/s)	TOP500 Rank	HPL Rmax (Eflop/s)	Speedup
1	DOE/SC/ORNL, USA	Frontier	8,730,112	6.861	1	1.102	6.2
2	RIKEN, Japan	Fugaku	7,630,848	2.000	2	0.4420	4.5
3	DOE/SC/ORNL, USA	Summit	2,414,592	1.411	4	0.1486	9.5
4	NVIDIA, USA	Selene	555,520	0.630	8	0.0630	9.9
5	DOE/SC/LBNL, USA	Perlmutter	761,856	0.590	7	0.0709	8.3
6	FZJ, Germany	JUWELS BM	449,280	0.470	11	0.0440	10.0
7	University of Florida, USA	HiPerGator	138,880	0.170	34	0.0170	9.9
8	SberCloud, Russia	Christofari Neo	98,208	0.123	47	0.0120	10.3
9	DOE/SC/ANL, USA	Polaris	259,840	0.114	14	0.0238	4.8
10	ITC, Japan	Wisteria	368,640	0.100	20	0.0220	4.5

Iterative Refinement for $Ax = b$

Iterative refinement: well-established method for improving an approximate solution to $Ax = b$

A is $n \times n$ and nonsingular; u is unit roundoff

Solve $Ax_0 = b$ by LU factorization

for $i = 0: \maxit$

$$r_i = b - Ax_i$$

$$\text{Solve } Ad_i = r_i \quad \text{via } d_i = U^{-1}(L^{-1}r_i)$$

$$x_{i+1} = x_i + d_i$$

Iterative Refinement for $Ax = b$

Iterative refinement: well-established method for improving an approximate solution to $Ax = b$

A is $n \times n$ and nonsingular; u is unit roundoff

Solve $Ax_0 = b$ by LU factorization (in precision u)

for $i = 0$: maxit

$r_i = b - Ax_i$ (in precision u^2)

Solve $Ad_i = r_i$ via $d_i = U^{-1}(L^{-1}r_i)$ (in precision u)

$x_{i+1} = x_i + d_i$ (in precision u)

"Traditional" (high-precision residual computation)

[Wilkinson, 1948] (fixed point), [Moler, 1967] (floating point)

Iterative Refinement for $Ax = b$

$$\kappa_{\infty}(A) = \|A^{-1}\|_{\infty} \|A\|_{\infty}$$

As long as $\kappa_{\infty}(A) \leq u^{-1}$,

- relative forward error is $O(u)$
- relative normwise and componentwise backward errors are $O(u)$

Solve $Ax_0 = b$ by LU factorization (in precision u)

for $i = 0$: maxit

$$r_i = b - Ax_i \quad (\text{in precision } u^2)$$

$$\text{Solve } Ad_i = r_i \quad \text{via } d_i = U^{-1}(L^{-1}r_i) \quad (\text{in precision } u)$$

$$x_{i+1} = x_i + d_i \quad (\text{in precision } u)$$

"Traditional" (high-precision residual computation)

[Wilkinson, 1948] (fixed point), [Moler, 1967] (floating point)

Iterative Refinement for $Ax = b$

Solve $Ax_0 = b$ by LU factorization (in precision u)

for $i = 0$: maxit

$$r_i = b - Ax_i \quad (\text{in precision } u)$$

$$\text{Solve } Ad_i = r_i \quad \text{via } d_i = U^{-1}(L^{-1}r_i) \quad (\text{in precision } u)$$

$$x_{i+1} = x_i + d_i \quad (\text{in precision } u)$$

"Fixed-Precision"

[Jankowski and Woźniakowski, 1977], [Skeel, 1980], [Higham, 1991]

Iterative Refinement for $Ax = b$

$$\text{cond}(A, x) = \| |A^{-1}| |A| |x| \|_{\infty} / \|x\|_{\infty}$$

As long as $\kappa_{\infty}(A) \leq u^{-1}$,

- relative forward error is $O(u) \mathbf{cond}(A, x)$
- relative normwise and componentwise backward errors are $O(u)$

Solve $Ax_0 = b$ by LU factorization (in precision u)

for $i = 0: \text{maxit}$

$$r_i = b - Ax_i \quad (\text{in precision } u)$$

$$\text{Solve } Ad_i = r_i \quad \text{via } d_i = U^{-1}(L^{-1}r_i) \quad (\text{in precision } u)$$

$$x_{i+1} = x_i + d_i \quad (\text{in precision } u)$$

"Fixed-Precision"

[Jankowski and Woźniakowski, 1977], [Skeel, 1980], [Higham, 1991]

Iterative Refinement for $Ax = b$

Solve $Ax_0 = b$ by LU factorization

(in precision $u^{1/2}$)

for $i = 0$: maxit

$$r_i = b - Ax_i$$

(in precision u)

$$\text{Solve } Ad_i = r_i \quad \text{via } d_i = U^{-1}(L^{-1}r_i)$$

(in precision u)

$$x_{i+1} = x_i + d_i$$

(in precision u)

"Low-precision factorization"

[Langou et al., 2006], [Arioli and Duff, 2009], [Hogg and Scott, 2010], [Abdelfattah et al., 2016]

Iterative Refinement for $Ax = b$

As long as $\kappa_\infty(A) \leq u^{-1/2}$,

- relative forward error is $O(u)\text{cond}(A, x)$
- relative normwise and componentwise backward errors are $O(u)$

Solve $Ax_0 = b$ by LU factorization (in precision $u^{1/2}$)

for $i = 0$: maxit

$$r_i = b - Ax_i \quad (\text{in precision } u)$$

$$\text{Solve } Ad_i = r_i \quad \text{via } d_i = U^{-1}(L^{-1}r_i) \quad (\text{in precision } u)$$

$$x_{i+1} = x_i + d_i \quad (\text{in precision } u)$$

"Low-precision factorization"

[Langou et al., 2006], [Arioli and Duff, 2009], [Hogg and Scott, 2010], [Abdelfattah et al., 2016]

Iterative Refinement for $Ax = b$

3-precision iterative refinement [C. and Higham, 2018]

u_f = factorization precision, u = working precision, u_r = residual precision

$$u_f \geq u \geq u_r$$

Solve $Ax_0 = b$ by LU factorization (in precision u_f)

for $i = 0$: maxit

$$r_i = b - Ax_i \quad (\text{in precision } u_r)$$

$$\text{Solve } Ad_i = r_i \quad (\text{in precision } u_s)$$

$$x_{i+1} = x_i + d_i \quad (\text{in precision } u)$$

u_s is the *effective precision* of the solve, with $u \leq u_s \leq u_f$

Forward Error for IR3

- Three precisions:
 - u_f : factorization precision
 - u : working precision
 - u_r : residual computation precision

$$\kappa_\infty(A) = \|A^{-1}\|_\infty \|A\|_\infty$$

$$\text{cond}(A) = \| |A^{-1}| |A| \|_\infty$$

$$\text{cond}(A, x) = \| |A^{-1}| |A| |x| \|_\infty / \|x\|_\infty$$

Forward Error for IR3

- Three precisions:

- u_f : factorization precision
- u : working precision
- u_r : residual computation precision

$$\kappa_\infty(A) = \|A^{-1}\|_\infty \|A\|_\infty$$

$$\text{cond}(A) = \| |A^{-1}| |A| \|_\infty$$

$$\text{cond}(A, x) = \| |A^{-1}| |A| |x| \|_\infty / \|x\|_\infty$$

Theorem [C. and Higham, SISC 40(2), 2018]

For IR in precisions $u_f \geq u \geq u_r$ and effective solve precision u_s , if

$$\phi_i \equiv 2u_s \min(\text{cond}(A), \kappa_\infty(A)\mu_i) + u_s \|E_i\|_\infty$$

is less than 1, then the forward error is reduced on the i th iteration by a factor $\approx \phi_i$ until an iterate \hat{x}_i is produced for which

$$\frac{\|x - \hat{x}_i\|_\infty}{\|x\|_\infty} \lesssim 4N u_r \text{cond}(A, x) + u,$$

where N is the maximum number of nonzeros per row in A .

Forward Error for IR3

- Three precisions:
 - u_f : factorization precision
 - u : working precision
 - u_r : residual computation precision

$$\kappa_\infty(A) = \|A^{-1}\|_\infty \|A\|_\infty$$

$$\text{cond}(A) = \| |A^{-1}| |A| \|_\infty$$

$$\text{cond}(A, x) = \| |A^{-1}| |A| |x| \|_\infty / \|x\|_\infty$$

Theorem [C. and Higham, SISC 40(2), 2018]

For IR in precisions $u_f \geq u \geq u_r$ and effective solve precision u_s , if

$$\phi_i \equiv 2u_s \min(\text{cond}(A), \kappa_\infty(A)\mu_i) + u_s \|E_i\|_\infty$$

is less than 1, then the forward error is reduced on the i th iteration by a factor $\approx \phi_i$ until an iterate \hat{x}_i is produced for which

$$\frac{\|x - \hat{x}_i\|_\infty}{\|x\|_\infty} \lesssim 4N u_r \text{cond}(A, x) + u,$$

where N is the maximum number of nonzeros per row in A .

→ Analogous traditional bounds: $\phi_i \equiv 3n u_f \kappa_\infty(A)$

Normwise Backward Error for IR3

Theorem [C. and Higham, SISC 40(2), 2018]

For IR in precisions $u_f \geq u \geq u_r$ and effective solve precision u_s , if

$$\phi_i \equiv (c_1 \kappa_\infty(A) + c_2) u_s$$

is less than 1, then the residual is reduced on the i th iteration by a factor $\approx \phi_i$ until an iterate \hat{x}_i is produced for which

$$\|b - A\hat{x}_i\|_\infty \lesssim Nu(\|b\|_\infty + \|A\|_\infty \|\hat{x}_i\|_\infty),$$

where N is the maximum number of nonzeros per row in A .

Effective Solve Precision

Allow for general solver:

Let u_s be the *effective precision* of the solve, with $u \leq u_s \leq u_f$

Effective Solve Precision

Allow for general solver:

Let u_s be the *effective precision* of the solve, with $u \leq u_s \leq u_f$

Assume computed solution \hat{d}_i to $Ad_i = \hat{r}_i$ satisfies:

1. $\hat{d}_i = (I + u_s E_i) d_i, \quad u_s \|E_i\|_\infty < 1$

→ normwise relative forward error is bounded by multiple of u_s and is less than 1

Effective Solve Precision

Allow for general solver:

Let u_s be the *effective precision* of the solve, with $u \leq u_s \leq u_f$

Assume computed solution \hat{d}_i to $Ad_i = \hat{r}_i$ satisfies:

1. $\hat{d}_i = (I + u_s E_i) d_i, \quad u_s \|E_i\|_\infty < 1$

→ normwise relative forward error is bounded by multiple of u_s and is less than 1

example: LU solve:

$$u_s \|E_i\|_\infty \leq 3n u_f \| |A^{-1}| |\hat{L}| |\hat{U}| \|_\infty$$

Effective Solve Precision

Allow for general solver:

Let u_s be the *effective precision* of the solve, with $u \leq u_s \leq u_f$

Assume computed solution \hat{d}_i to $Ad_i = \hat{r}_i$ satisfies:

1. $\hat{d}_i = (I + u_s E_i) d_i, \quad u_s \|E_i\|_\infty < 1$

→ normwise relative forward error is bounded by multiple of u_s and is less than 1

2. $\|\hat{r}_i - A\hat{d}_i\|_\infty \leq u_s (c_1 \|A\|_\infty \|\hat{d}_i\|_\infty + c_2 \|\hat{r}_i\|_\infty)$

→ normwise relative backward error is at most $\max(c_1, c_2) u_s$

example: LU solve:

$$u_s \|E_i\|_\infty \leq 3n u_f \| |A^{-1}| |\hat{L}| |\hat{U}| \|_\infty$$

Effective Solve Precision

Allow for general solver:

Let u_s be the *effective precision* of the solve, with $u \leq u_s \leq u_f$

Assume computed solution \hat{d}_i to $Ad_i = \hat{r}_i$ satisfies:

1. $\hat{d}_i = (I + u_s E_i) d_i, \quad u_s \|E_i\|_\infty < 1$

→ normwise relative forward error is bounded by multiple of u_s and is less than 1

2. $\|\hat{r}_i - A\hat{d}_i\|_\infty \leq u_s (c_1 \|A\|_\infty \|\hat{d}_i\|_\infty + c_2 \|\hat{r}_i\|_\infty)$

→ normwise relative backward error is at most $\max(c_1, c_2) u_s$

example: LU solve:

$$u_s \|E_i\|_\infty \leq 3n u_f \| |A^{-1}| |\hat{L}| |\hat{U}| \|_\infty$$

$$\max(c_1, c_2) u_s \leq \frac{3n u_f \| |\hat{L}| |\hat{U}| \|_\infty}{\|A\|_\infty}$$

Effective Solve Precision

Allow for general solver:

Let u_s be the *effective precision* of the solve, with $u \leq u_s \leq u_f$

example: LU solve:

Assume computed solution \hat{d}_i to $Ad_i = \hat{r}_i$ satisfies:

1. $\hat{d}_i = (I + u_s E_i) d_i, \quad u_s \|E_i\|_\infty < 1$

→ normwise relative forward error is bounded by multiple of u_s and is less than 1

$$u_s \|E_i\|_\infty \leq 3n u_f \| |A^{-1}| |\hat{L}| |\hat{U}| \|_\infty$$

2. $\|\hat{r}_i - A\hat{d}_i\|_\infty \leq u_s (c_1 \|A\|_\infty \|\hat{d}_i\|_\infty + c_2 \|\hat{r}_i\|_\infty)$

→ normwise relative backward error is at most $\max(c_1, c_2) u_s$

$$\max(c_1, c_2) u_s \leq \frac{3n u_f \| | \hat{L} | | \hat{U} | \|_\infty}{\|A\|_\infty}$$

3. $|\hat{r}_i - A\hat{d}_i| \leq u_s G_i |\hat{d}_i|$

→ componentwise relative backward error is bounded by a multiple of u_s

$E_i, c_1, c_2,$ and G_i depend on $A, \hat{r}_i, n,$ and u_s

Effective Solve Precision

Allow for general solver:

Let u_s be the *effective precision* of the solve, with $u \leq u_s \leq u_f$

example: LU solve:

Assume computed solution \hat{d}_i to $Ad_i = \hat{r}_i$ satisfies:

1. $\hat{d}_i = (I + u_s E_i) d_i, \quad u_s \|E_i\|_\infty < 1$

→ normwise relative forward error is bounded by multiple of u_s and is less than 1

$$u_s \|E_i\|_\infty \leq 3n u_f \| |A^{-1}| |\hat{L}| |\hat{U}| \|_\infty$$

2. $\|\hat{r}_i - A\hat{d}_i\|_\infty \leq u_s (c_1 \|A\|_\infty \|\hat{d}_i\|_\infty + c_2 \|\hat{r}_i\|_\infty)$

→ normwise relative backward error is at most $\max(c_1, c_2) u_s$

$$\max(c_1, c_2) u_s \leq \frac{3n u_f \| |\hat{L}| |\hat{U}| \|_\infty}{\|A\|_\infty}$$

3. $|\hat{r}_i - A\hat{d}_i| \leq u_s G_i |\hat{d}_i|$

→ componentwise relative backward error is bounded by a multiple of u_s

$$u_s \|G_i\|_\infty \leq 3n u_f \| |\hat{L}| |\hat{U}| \|_\infty$$

$E_i, c_1, c_2,$ and G_i depend on $A, \hat{r}_i, n,$ and u_s

Effective Solve Precision

Allow for general solver:

Let u_s be the *effective precision* of the solve, with $u \leq u_s \leq u_f$

Assume computed solution \hat{d}_i to $Ad_i = \hat{r}_i$ satisfies:

1. $\hat{d}_i = (I + u_s E_i) d_i$, $u_s \|E_i\|_\infty < 1$
→ normwise relative forward error is bounded by multiple of u_s and is less than 1

2. $\|\hat{r}_i - A\hat{d}_i\|_\infty \leq u_s (c_1 \|A\|_\infty \|\hat{d}_i\|_\infty + c_2 \|\hat{r}_i\|_\infty)$
→ normwise relative backward error is at most $\max(c_1, c_2) u_s$

3. $|\hat{r}_i - A\hat{d}_i| \leq u_s G_i |\hat{d}_i|$
→ componentwise relative backward error is bounded by a multiple of u_s

E_i, c_1, c_2 , and G_i depend on A , \hat{r}_i , n , and u_s

example: LU solve:

$$u_s = u_f$$

$$u_s \|E_i\|_\infty \leq 3n u_f \| |A^{-1}| |\hat{L}| |\hat{U}| \|_\infty$$

$$\max(c_1, c_2) u_s \leq \frac{3n u_f \| |\hat{L}| |\hat{U}| \|_\infty}{\|A\|_\infty}$$

$$u_s \|G_i\|_\infty \leq 3n u_f \| |\hat{L}| |\hat{U}| \|_\infty$$

IR3: Summary

Standard (LU-based) IR in three precisions ($u_s = u_f$)

Half $\approx 10^{-4}$, Single $\approx 10^{-8}$, Double $\approx 10^{-16}$, Quad $\approx 10^{-34}$

u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
				norm	comp	
H	S	S	10^4	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
H	D	D	10^4	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
S	S	S	10^8	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
S	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
S	D	D	10^8	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}

IR3: Summary

Standard (LU-based) IR in three precisions ($u_s = u_f$)

Half $\approx 10^{-4}$, Single $\approx 10^{-8}$, Double $\approx 10^{-16}$, Quad $\approx 10^{-34}$

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LP fact.	H	S	S	10^4	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
LP fact.	H	D	D	10^4	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
LP fact.	S	S	S	10^8	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
	S	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
	S	D	D	10^8	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}

IR3: Summary

Standard (LU-based) IR in three precisions ($u_s = u_f$)

Half $\approx 10^{-4}$, Single $\approx 10^{-8}$, Double $\approx 10^{-16}$, Quad $\approx 10^{-34}$

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LP fact.	H	S	S	10^4	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
LP fact.	H	D	D	10^4	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
Fixed	S	S	S	10^8	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
	S	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LP fact.	S	D	D	10^8	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}

IR3: Summary

Standard (LU-based) IR in three precisions ($u_s = u_f$)

Half $\approx 10^{-4}$, Single $\approx 10^{-8}$, Double $\approx 10^{-16}$, Quad $\approx 10^{-34}$

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LP fact.	H	S	S	10^4	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
LP fact.	H	D	D	10^4	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
Fixed	S	S	S	10^8	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
Trad.	S	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LP fact.	S	D	D	10^8	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}

IR3: Summary

Standard (LU-based) IR in three precisions ($u_s = u_f$)

Half $\approx 10^{-4}$, Single $\approx 10^{-8}$, Double $\approx 10^{-16}$, Quad $\approx 10^{-34}$

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LP fact.	H	S	S	10^4	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
New	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
LP fact.	H	D	D	10^4	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
New	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
Fixed	S	S	S	10^8	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
Trad.	S	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LP fact.	S	D	D	10^8	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
New	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}

IR3: Summary

Standard (LU-based) IR in three precisions ($u_s = u_f$)

Half $\approx 10^{-4}$, Single $\approx 10^{-8}$, Double $\approx 10^{-16}$, Quad $\approx 10^{-34}$

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LP fact.	H	S	S	10^4	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
New	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
LP fact.	H	D	D	10^4	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
New	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
Fixed	S	S	S	10^8	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
Trad.	S	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LP fact.	S	D	D	10^8	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
New	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}

\Rightarrow Benefit of IR3 vs. "LP fact.": no $\text{cond}(A, x)$ term in forward error

IR3: Summary

Standard (LU-based) IR in three precisions ($u_s = u_f$)

Half $\approx 10^{-4}$, Single $\approx 10^{-8}$, Double $\approx 10^{-16}$, Quad $\approx 10^{-34}$

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LP fact.	H	S	S	10^4	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
New	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
LP fact.	H	D	D	10^4	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
New	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
Fixed	S	S	S	10^8	10^{-8}	10^{-8}	$\text{cond}(A, x) \cdot 10^{-8}$
Trad.	S	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LP fact.	S	D	D	10^8	10^{-16}	10^{-16}	$\text{cond}(A, x) \cdot 10^{-16}$
New	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}

\Rightarrow Benefit of IR3 vs. traditional IR: As long as $\kappa_\infty(A) \leq 10^4$, can use lower precision factorization w/no loss of accuracy!

GMRES-Based Iterative Refinement

- Observation [Rump, 1990]: if \hat{L} and \hat{U} are computed LU factors of A in precision u_f , then

$$\kappa_\infty(\hat{U}^{-1}\hat{L}^{-1}A) \approx 1 + \kappa_\infty(A)u_f,$$

even if $\kappa_\infty(A) \gg u_f^{-1}$.

GMRES-Based Iterative Refinement

- Observation [Rump, 1990]: if \hat{L} and \hat{U} are computed LU factors of A in precision u_f , then

$$\kappa_\infty(\hat{U}^{-1}\hat{L}^{-1}A) \approx 1 + \kappa_\infty(A)u_f,$$

even if $\kappa_\infty(A) \gg u_f^{-1}$.

GMRES-IR [C. and Higham, SISC 39(6), 2017]

- To compute the updates d_i , apply GMRES to

$$\underbrace{\hat{U}^{-1}\hat{L}^{-1}A}_{\tilde{A}} d_i = \underbrace{\hat{U}^{-1}\hat{L}^{-1}r_i}_{\tilde{r}_i}$$

GMRES-Based Iterative Refinement

- Observation [Rump, 1990]: if \hat{L} and \hat{U} are computed LU factors of A in precision u_f , then

$$\kappa_\infty(\hat{U}^{-1}\hat{L}^{-1}A) \approx 1 + \kappa_\infty(A)u_f,$$

even if $\kappa_\infty(A) \gg u_f^{-1}$.

GMRES-IR [C. and Higham, SISC 39(6), 2017]

- To compute the updates d_i , apply GMRES to $\underbrace{\hat{U}^{-1}\hat{L}^{-1}A}_{\tilde{A}}d_i = \underbrace{\hat{U}^{-1}\hat{L}^{-1}r_i}_{\tilde{r}_i}$

Solve $Ax_0 = b$ by LU factorization

for $i = 0$: maxit

$$r_i = b - Ax_i$$

Solve $Ad_i = r_i$ via GMRES on $\tilde{A}d_i = \tilde{r}_i$

$$x_{i+1} = x_i + d_i$$

GMRES-Based Iterative Refinement

- Observation [Rump, 1990]: if \hat{L} and \hat{U} are computed LU factors of A in precision u_f , then

$$\kappa_\infty(\hat{U}^{-1}\hat{L}^{-1}A) \approx 1 + \kappa_\infty(A)u_f,$$

even if $\kappa_\infty(A) \gg u_f^{-1}$.

GMRES-IR [C. and Higham, SISC 39(6), 2017]

- To compute the updates d_i , apply GMRES to $\underbrace{\hat{U}^{-1}\hat{L}^{-1}A}_{\tilde{A}}d_i = \underbrace{\hat{U}^{-1}\hat{L}^{-1}r_i}_{\tilde{r}_i}$

Solve $Ax_0 = b$ by LU factorization

for $i = 0$: maxit

$$r_i = b - Ax_i$$

Solve $Ad_i = r_i$ via GMRES on $\tilde{A}d_i = \tilde{r}_i$

$$x_{i+1} = x_i + d_i$$


$$u_s = u$$

GMRES-IR: Summary

GMRES-IR: Solve for d_i via GMRES on $U^{-1}L^{-1}Ad_i = U^{-1}L^{-1}r_i$

GMRES-based IR in three precisions ($u_s = u$)

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LU-IR	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
GMRES-IR	H	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LU-IR	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	S	D	Q	10^{16}	10^{-16}	10^{-16}	10^{-16}
LU-IR	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	H	D	Q	10^{12}	10^{-16}	10^{-16}	10^{-16}


⇒ With GMRES-IR, lower precision factorization will work for higher $\kappa_\infty(A)$

GMRES-IR: Summary

GMRES-IR: Solve for d_i via GMRES on $U^{-1}L^{-1}Ad_i = U^{-1}L^{-1}r_i$

GMRES-based IR in three precisions ($u_s = u$)

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LU-IR	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
GMRES-IR	H	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LU-IR	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	S	D	Q	10^{16}	10^{-16}	10^{-16}	10^{-16}
LU-IR	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	H	D	Q	10^{12}	10^{-16}	10^{-16}	10^{-16}


 $\kappa_\infty(A) \leq u^{-1/2} u_f^{-1}$


\Rightarrow With GMRES-IR, lower precision factorization will work for higher $\kappa_\infty(A)$

GMRES-IR: Summary

GMRES-IR: Solve for d_i via GMRES on $U^{-1}L^{-1}Ad_i = U^{-1}L^{-1}r_i$

GMRES-based IR in three precisions ($u_s = u$)

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LU-IR	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
GMRES-IR	H	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LU-IR	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	S	D	Q	10^{16}	10^{-16}	10^{-16}	10^{-16}
LU-IR	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	H	D	Q	10^{12}	10^{-16}	10^{-16}	10^{-16}


 $\kappa_\infty(A) \leq u^{-1/2} u_f^{-1}$


\Rightarrow As long as $\kappa_\infty(A) \leq 10^{12}$, can use half precision factorization and still obtain double precision accuracy!

GMRES-IR: Summary

GMRES-IR: Solve for d_i via GMRES on $U^{-1}L^{-1}Ad_i = U^{-1}L^{-1}r_i$

GMRES-based IR in three precisions ($u_s = u$)

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LU-IR	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
GMRES-IR	H	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LU-IR	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	S	D	Q	10^{16}	10^{-16}	10^{-16}	10^{-16}
LU-IR	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	H	D	Q	10^{12}	10^{-16}	10^{-16}	10^{-16}


 $\kappa_\infty(A) \leq u^{-1/2} u_f^{-1}$

\Rightarrow As long as $\kappa_\infty(A) \leq 10^{12}$, can use half precision factorization and still obtain double precision accuracy!


Recent work: 5-precision GMRES-IR [Amestoy, et al., 2021]

GMRES-IR: Summary

GMRES-IR: Solve for d_i via GMRES on $U^{-1}L^{-1}Ad_i = U^{-1}L^{-1}r_i$


GMRES-based IR in three precisions ($u_s = u$)

	u_f	u	u_r	$\max \kappa_\infty(A)$	Backward error		Forward error
					norm	comp	
LU-IR	H	S	D	10^4	10^{-8}	10^{-8}	10^{-8}
GMRES-IR	H	S	D	10^8	10^{-8}	10^{-8}	10^{-8}
LU-IR	S	D	Q	10^8	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	S	D	Q	10^{16}	10^{-16}	10^{-16}	10^{-16}
LU-IR	H	D	Q	10^4	10^{-16}	10^{-16}	10^{-16}
GMRES-IR	H	D	Q	10^{12}	10^{-16}	10^{-16}	10^{-16}


 $\kappa_\infty(A) \leq u^{-1/2} u_f^{-1}$

\Rightarrow As long as $\kappa_\infty(A) \leq 10^{12}$, can use half precision factorization and still obtain double precision accuracy!

Recent work: 5-precision GMRES-IR [Amestoy, et al., 2021]


 $\kappa_\infty(A) \leq u^{-1/3} u_f^{-2/3}$

GMRES-IR with Inexact Preconditioners

- Existing analyses of GMRES-IR assume we use full LU factors
- In practice, often want to use approximate preconditioners (ILU, SPAI, etc.)

GMRES-IR with Inexact Preconditioners

- Existing analyses of GMRES-IR assume we use full LU factors
- In practice, often want to use approximate preconditioners (ILU, SPAI, etc.)
- [Amestoy et al., 2022]
 - Analysis of **block low-rank (BLR) LU** within GMRES-IR
 - Analysis of use of **static pivoting** in LU within GMRES-IR

GMRES-IR with Inexact Preconditioners

- Existing analyses of GMRES-IR assume we use full LU factors
- In practice, often want to use approximate preconditioners (ILU, SPAI, etc.)
- [Amestoy et al., 2022]
 - Analysis of **block low-rank (BLR) LU** within GMRES-IR
 - Analysis of use of **static pivoting** in LU within GMRES-IR
- [C., Khan, 2022]
 - Analysis of **sparse approximate inverse (SPAI) preconditioners** within GMRES-IR

SPAI Preconditioners

Goal: Construct sparse matrix $M \approx A^{-1}$ (for survey see [Benzi, 2002])

Approach of [Grote, Huckle, 1997]: Construct columns m_k of M dynamically

Given matrix A , initial sparsity structure J , and tolerance ε

For each column k :

 Compute QR factorization of submatrix of A defined by J

 Use QR factorization to solve $\min_{m_k} \|e_k - Am_k\|_2$

 If $\|r_k\|_2 = \|e_k - Am_k\|_2 \leq \varepsilon$

 break;

 Else

 add select nonzeros to J , repeat.

SPAI Preconditioners

Goal: Construct sparse matrix $M \approx A^{-1}$ (for survey see [Benzi, 2002])

Approach of [Grote, Huckle, 1997]: Construct columns m_k of M dynamically

Given matrix A , initial sparsity structure J , and tolerance ε

For each column k :

 Compute QR factorization of submatrix of A defined by J

 Use QR factorization to solve $\min_{m_k} \|e_k - Am_k\|_2$

 If $\|r_k\|_2 = \|e_k - Am_k\|_2 \leq \varepsilon$

 break;

 Else

 add select nonzeros to J , repeat.

Benefits: Highly parallelizable

But **construction can still be costly**, esp. for large-scale problems

[Gao, Chen, He, 2021], [Chao, 2001], [Benzi, Tuma, 1999], [He, Yin, Gao, 2020]

SPAI Preconditioners in Low Precision

What is the effect of using low precision in SPAI construction?

Notes and assumptions:

- We will assume that the SPAI construction is performed in some precision u_f
- We will denote quantities computed in finite precision with hats
- In our application, we want a left preconditioner, so we will run the algorithm on A^T and set $M \leftarrow M^T$.
- We will assume that the QR factorization of the submatrix of A^T is computed fully using HouseholderQR/TSQR

SPAI Preconditioners in Low Precision

Two interesting questions:

1. Assuming we **impose no maximum sparsity pattern** on \widehat{M} , under what constraint on u_f can we guarantee that $\|\hat{r}_k\|_2 \leq \varepsilon$, with $\hat{r}_k = fl_{u_f}(e_k - A^T \hat{m}_k^T)$ for the computed \hat{m}_k^T ?

SPAI Preconditioners in Low Precision

Two interesting questions:

1. Assuming we **impose no maximum sparsity pattern** on \widehat{M} , under what constraint on u_f can we guarantee that $\|\hat{r}_k\|_2 \leq \varepsilon$, with $\hat{r}_k = fl_{u_f}(e_k - A^T \widehat{m}_k^T)$ for the computed \widehat{m}_k^T ?
2. Assume that when M is computed in exact arithmetic, we quit as soon as $\|r_k\| \leq \varepsilon$. For \widehat{M} computed in precision u_f with **the same sparsity pattern as M** , what is $\|e_k - A^T \widehat{m}_k^T\|_2$?

SPAI Preconditioning in Low Precision

Using standard rounding error analysis and perturbation results for LS problems, we have

$$\|\hat{r}_k\|_2 \leq n^3 u_f \left(\|e_k\| + |A^T| \|\hat{m}_k^T\| \right)_2.$$

So in order to guarantee we eventually reach a solution with $\|\hat{r}_k\|_2 \leq \varepsilon$, we need

$$n^3 u_f \left(\|e_k\| + |A^T| \|\hat{m}_k^T\| \right)_2 \leq \varepsilon.$$

SPAI Preconditioning in Low Precision

Using standard rounding error analysis and perturbation results for LS problems, we have

$$\|\hat{r}_k\|_2 \leq n^3 u_f \left(\|e_k\| + |A^T| \|\hat{m}_k^T\| \right)_2.$$

So in order to guarantee we eventually reach a solution with $\|\hat{r}_k\|_2 \leq \varepsilon$, we need

$$n^3 u_f \left(\|e_k\| + |A^T| \|\hat{m}_k^T\| \right)_2 \leq \varepsilon.$$

→ problem must not be so ill-conditioned WRT u_f that we incur an error greater than ε just computing the residual

SPAI Preconditioning in Low Precision

Can turn this into the looser but more descriptive a priori bound:

$$\text{cond}_2(A^T) \lesssim \varepsilon u_f^{-1},$$

where $\text{cond}_2(A^T) = \|A^{-T}\|A^T\|_2$.

SPAI Preconditioning in Low Precision

Can turn this into the looser but more descriptive a priori bound:

$$\text{cond}_2(A^T) \lesssim \varepsilon u_f^{-1},$$

where $\text{cond}_2(A^T) = \|A^{-T}\|A^T\|_2$.

Another view: with a given matrix A and a given precision u_f , one must set ε such that

$$\varepsilon \geq u_f \text{cond}_2(A^T).$$

Confirms intuition: The more approximate the inverse, the lower the precision we can us.

SPAI Preconditioning in Low Precision

Can turn this into the looser but more descriptive a priori bound:

$$\text{cond}_2(A^T) \lesssim \varepsilon u_f^{-1},$$

where $\text{cond}_2(A^T) = \|A^{-T}\|A^T\|_2$.

Another view: with a given matrix A and a given precision u_f , one must set ε such that

$$\varepsilon \geq u_f \text{cond}_2(A^T).$$

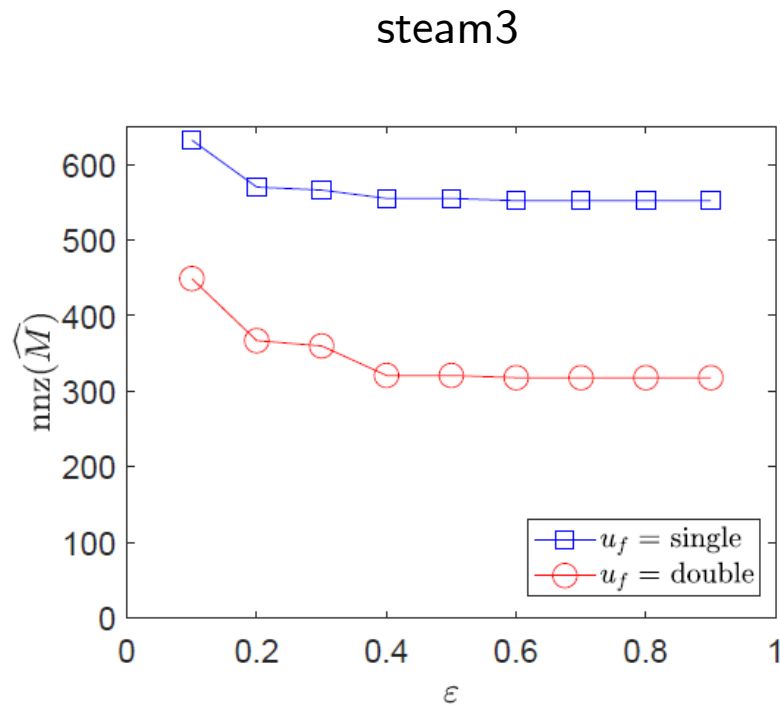
Confirms intuition: **The more approximate the inverse, the lower the precision we can use.**

Resulting bounds for \hat{M} :

$$\|I - A^T \hat{M}^T\|_F \leq 2\sqrt{n}\varepsilon, \quad \|I - \hat{M}A\|_\infty \leq 2n\varepsilon$$

Size of SPAI Preconditioner in Low Precision

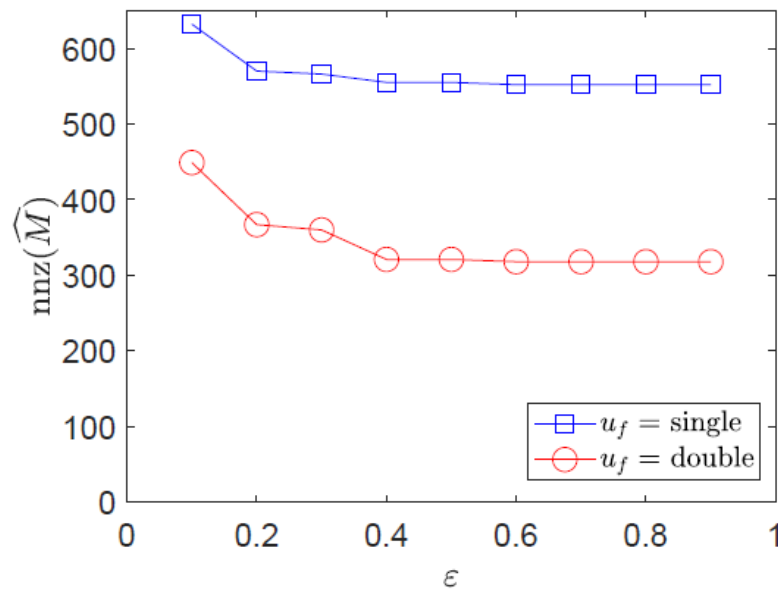
How does precision used affect the number of nonzeros in \widehat{M} ?



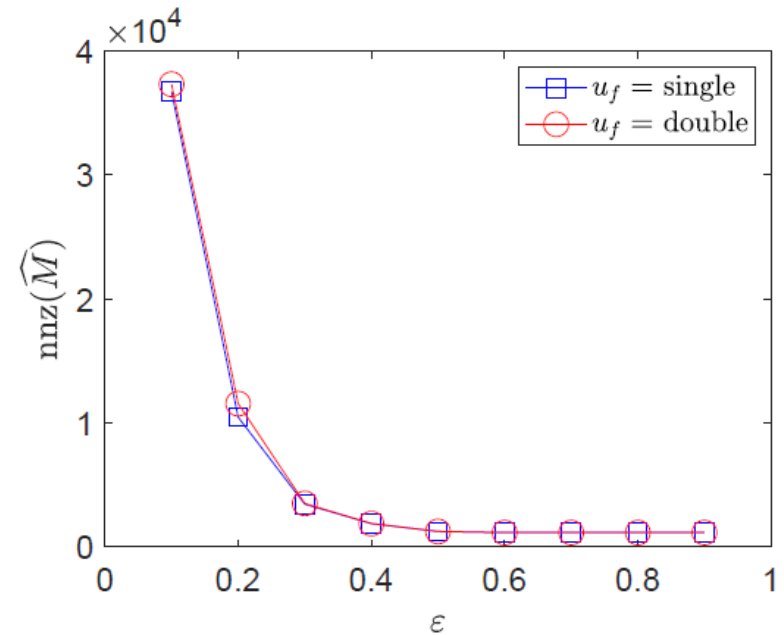
Size of SPAI Preconditioner in Low Precision

How does precision used affect the number of nonzeros in \widehat{M} ?

steam3



saylr1



Second Question

Assume that when M is computed in exact arithmetic, we quit as soon as $\|r_k\| \leq \varepsilon$. For \hat{M} computed in precision u_f with *the same sparsity pattern as M* , what is $\|e_k - A^T \hat{m}_k^T\|_2$?

Second Question

Assume that when M is computed in exact arithmetic, we quit as soon as $\|r_k\| \leq \varepsilon$. For \widehat{M} computed in precision u_f with *the same sparsity pattern as M* , what is $\|e_k - A^T \widehat{m}_k^T\|_2$?

In this case, we obtain the bound

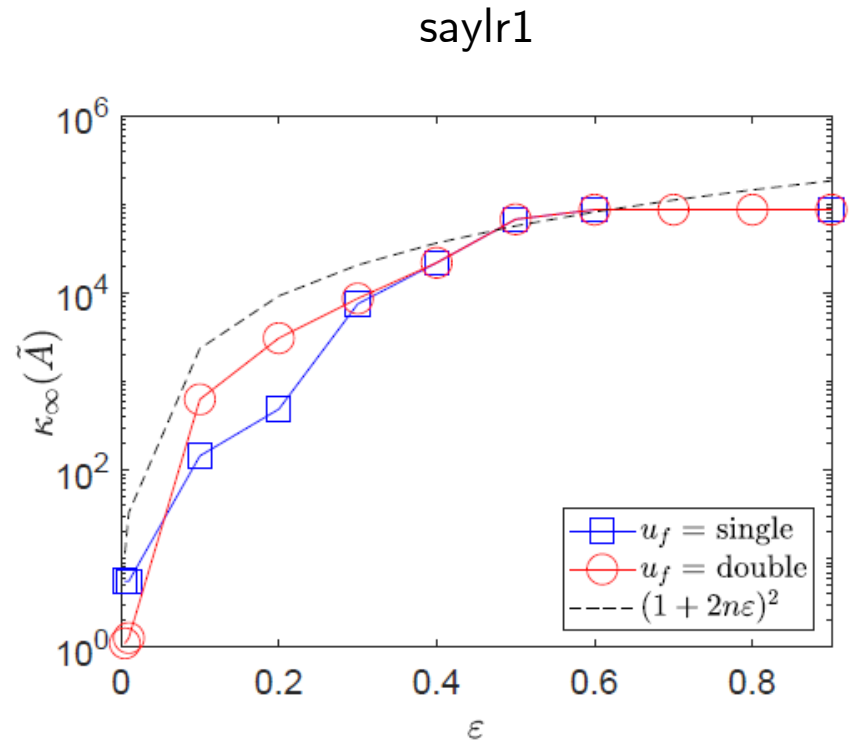
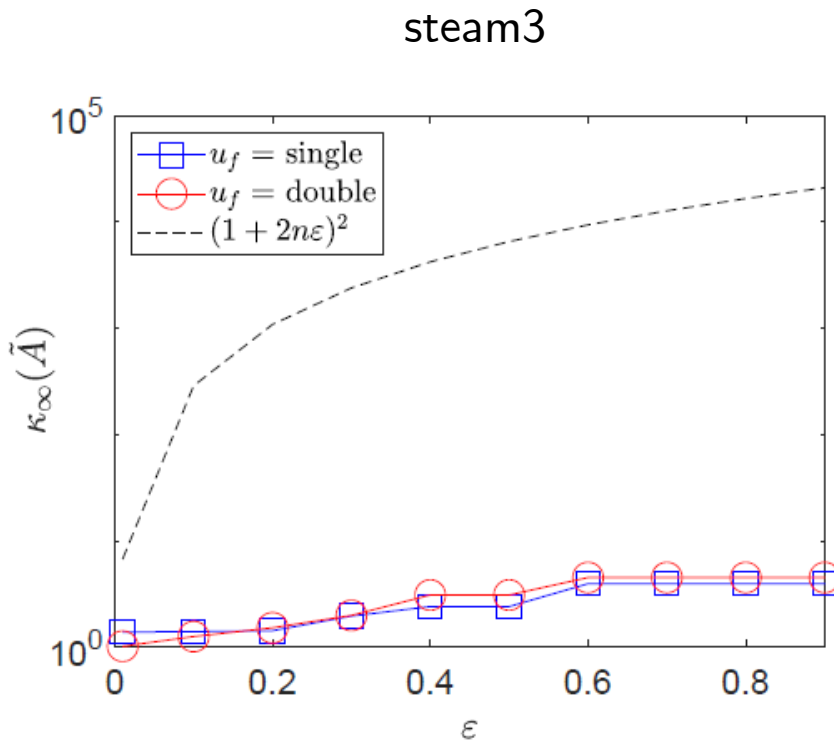
$$\|I - \widehat{M}A\|_\infty \leq n \left(\varepsilon + n^{7/2} u_f \kappa_\infty(A) \right).$$

→ If $\kappa_\infty(A) \gg \varepsilon u_f^{-1}$, then computed \widehat{M} with same sparsity structure as M can be of much lower quality.

Low Precision SPAI within GMRES-IR

Using \hat{M} computed in precision u_f , for the preconditioned system $\tilde{A} = \hat{M}A$,

$$\kappa_{\infty}(\tilde{A}) \lesssim (1 + 2n\varepsilon)^2.$$



Low Precision SPAI within GMRES-IR

To guarantee that both SPAI construction will complete and the GMRES-based iterative refinement scheme will converge, we must have roughly

$$nu_f \text{cond}_2(A^T) \lesssim n\varepsilon \lesssim u^{-1/2}.$$

Low Precision SPAI within GMRES-IR

To guarantee that both SPAI construction will complete and the GMRES-based iterative refinement scheme will converge, we must have roughly

$$nu_f \text{cond}_2(A^T) \lesssim n\varepsilon \lesssim u^{-1/2}.$$

\hat{M} can be
constructed

Low Precision SPAI within GMRES-IR

To guarantee that both SPAI construction will complete and the GMRES-based iterative refinement scheme will converge, we must have roughly

$$nu_f \text{cond}_2(A^T) \lesssim n\varepsilon \lesssim u^{-1/2}.$$

\hat{M} can be
constructed

\hat{M} is a good enough
preconditioner

Low Precision SPAI within GMRES-IR

To guarantee that both SPAI construction will complete and the GMRES-based iterative refinement scheme will converge, we must have roughly

$$nu_f \text{cond}_2(A^T) \lesssim n\varepsilon \lesssim u^{-1/2}.$$

\hat{M} can be
constructed

\hat{M} is a good enough
preconditioner

If ε satisfies these constraints, then the **constraints on condition number** for forward and backward errors to converge are the **same as for GMRES-IR with full LU factorization**.

Low Precision SPAI within GMRES-IR

To guarantee that both SPAI construction will complete and the GMRES-based iterative refinement scheme will converge, we must have roughly

$$nu_f \text{cond}_2(A^T) \lesssim n\varepsilon \lesssim u^{-1/2}.$$

\hat{M} can be
constructed

\hat{M} is a good enough
preconditioner

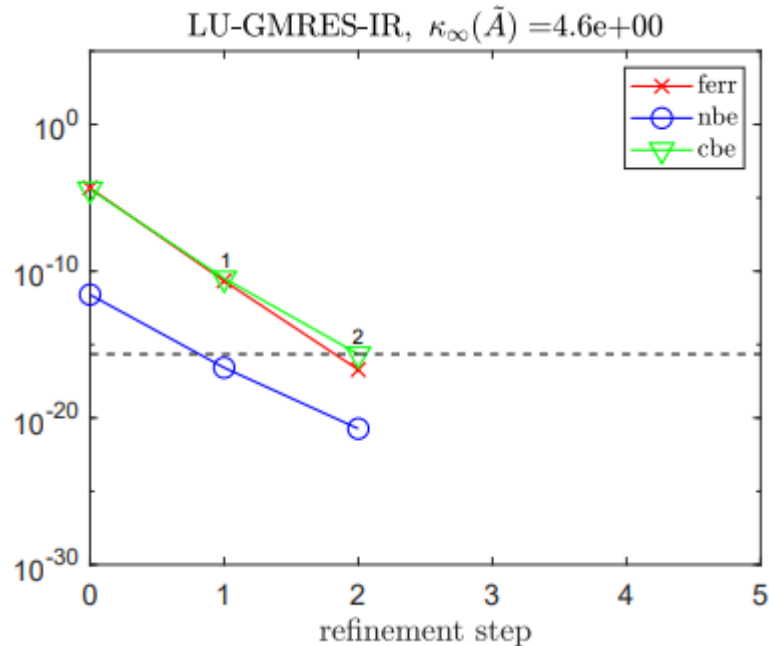
If ε satisfies these constraints, then the **constraints on condition number** for forward and backward errors to converge are the **same as for GMRES-IR with full LU factorization**.

Compared to GMRES-IR with full LU factorization, in general expect **slower convergence, but much sparser preconditioner**.

SPAI-GMRES-IR Example

Matrix: steam1, $n = 240$, $\text{nnz} = 2,248$, $\kappa_\infty(A) = 3 \cdot 10^7$, $\text{cond}(A^T) = 3 \cdot 10^3$

$(u_f, u, u_r) = (\text{single}, \text{double}, \text{quad})$

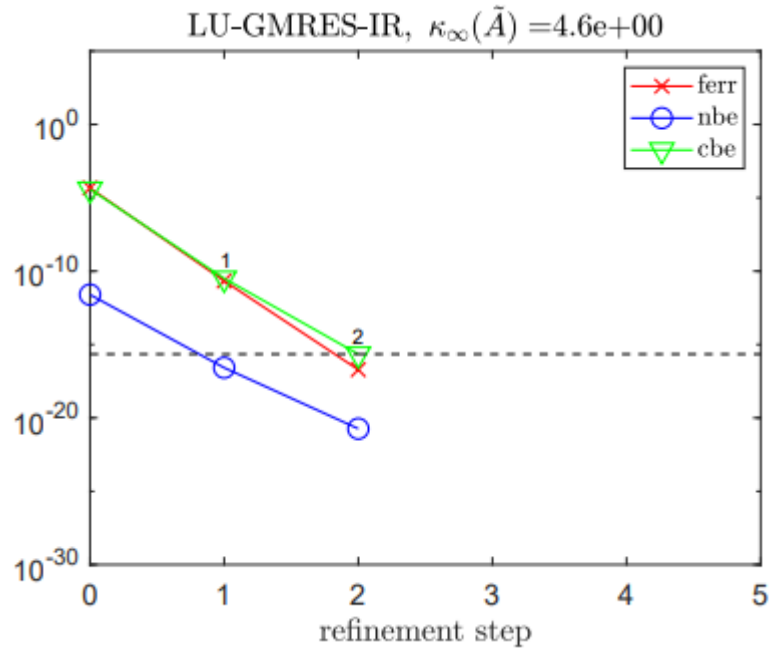


$\text{nnz}(L + U) = 21,657$

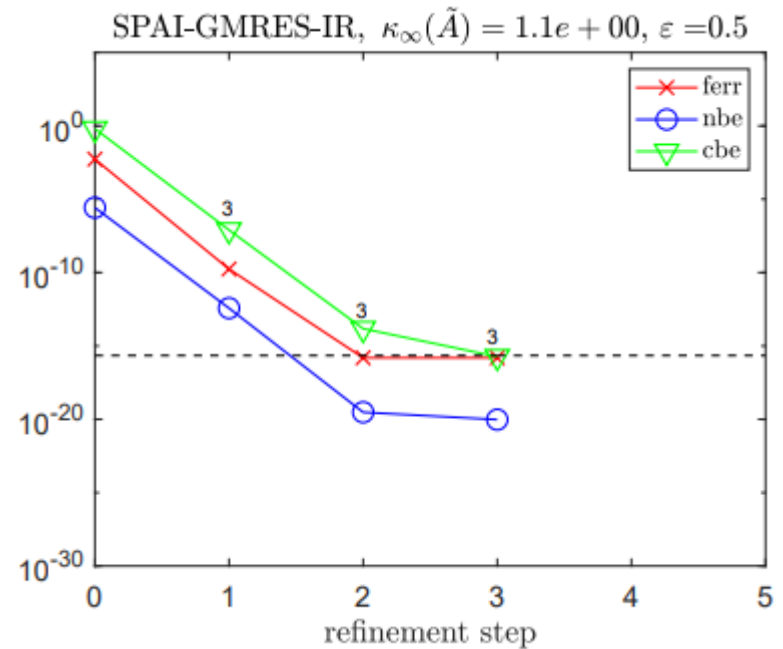
SPAI-GMRES-IR Example

Matrix: steam1, $n = 240$, $\text{nnz} = 2,248$, $\kappa_\infty(A) = 3 \cdot 10^7$, $\text{cond}(A^T) = 3 \cdot 10^3$

$(u_f, u, u_r) = (\text{single}, \text{double}, \text{quad})$



$\text{nnz}(L + U) = 21,657$



$\text{nnz}(M) = 2,248$

Is there a point in using precision higher than that dictated by $u_f \text{cond}_2(A^T) \leq \varepsilon$?

Matrix: bfwa782, $n = 782$, $\text{nnz} = 7514$, $\kappa_\infty(A) = 7 \cdot 10^3$, $\text{cond}(A^T) = 1 \cdot 10^3$

$(u_f, u, u_r) = (\mathbf{half}, \text{single}, \text{double})$

Preconditioner	$\kappa_\infty(\tilde{A})$	Precond. nnz	GMRES-IR steps/iteration
SPAI ($\varepsilon = 0.2$)	$2.1e + 02$	28053	67 (31, 36)
SPAI ($\varepsilon = 0.5$)	$9.7e + 02$	7528	153 (71, 82)
Full LU	$2.9e + 00$	347828	7 (3,4)
None	$6.8e + 03$	0	379 (172, 207)

Is there a point in using precision higher than that dictated by $u_f \text{cond}_2(A^T) \leq \varepsilon$?

Matrix: bfw782, $n = 782$, $\text{nnz} = 7514$, $\kappa_\infty(A) = 7 \cdot 10^3$, $\text{cond}(A^T) = 1 \cdot 10^3$

$$(u_f, u, u_r) = (\mathbf{half}, \text{single}, \text{double})$$

Preconditioner	$\kappa_\infty(\tilde{A})$	Precond. nnz	GMRES-IR steps/iteration
SPAI ($\varepsilon = 0.2$)	$2.1e + 02$	28053	67 (31, 36)
SPAI ($\varepsilon = 0.5$)	$9.7e + 02$	7528	153 (71, 82)
Full LU	$2.9e + 00$	347828	7 (3,4)
None	$6.8e + 03$	0	379 (172, 207)

$$(u_f, u, u_r) = (\mathbf{single}, \text{single}, \text{double})$$

Preconditioner	$\kappa_\infty(\tilde{A})$	Precond. nnz	GMRES-IR steps/iteration
SPAI ($\varepsilon = 0.2$)	$2.2e + 02$	26801	69 (32, 37)
SPAI ($\varepsilon = 0.5$)	$9.7e + 02$	7529	153 (71, 82)
Full LU	$1.0e + 00$	347828	1 (1)
None	$6.8e + 03$	0	379 (172, 207)

Randomized Limited Memory Preconditioners

Let $A \in \mathbb{R}^{n \times n}$ be a symmetric positive semidefinite matrix. Want to solve

$$(A + \mu I)x = b$$

where $\mu \geq 0$ is set so that $A + \mu I$ is positive definite. Assume A has rapidly decreasing eigenvalues or cluster of large eigenvalues.

Randomized Limited Memory Preconditioners

Let $A \in \mathbb{R}^{n \times n}$ be a symmetric positive semidefinite matrix. Want to solve

$$(A + \mu I)x = b$$

where $\mu \geq 0$ is set so that $A + \mu I$ is positive definite. Assume A has rapidly decreasing eigenvalues or cluster of large eigenvalues.

Want to solve using PCG using **spectral limited memory preconditioner** [Gratton, Sartenaer, Tshimanga, 2011], [Tshimanga et al., 2008]:

$$P = I - UU^T + \frac{1}{\alpha + \mu} U(\Theta + \mu I)U^T$$
$$P^{-1} = I - UU^T + (\alpha + \mu)U(\Theta + \mu I)^{-1}U^T$$

where columns of $U \in \mathbb{R}^{n \times k}$ are k approximate eigenvectors of A and $U^T U = I$, Θ is diagonal with approximations to eigenvalues of A , and $\alpha \geq 0$.

Used in data assimilation [Laloyaux et al., 2018], [Mogensen, Alonso Balmaseda, Weaver, 2012], [Moore et al., 2011], [Daužickaitė, Lawless, Scott, van Leeuwen, 2021]

Randomized Nyström Approximation

Want to compute a rank- k approximation $A \approx U\Theta U^T$ via the randomized Nyström method.

Nyström approximation:

$$A_N = (AQ)(Q^T A Q)^+ (AQ)^T$$

where Q is an $n \times k$ sampling matrix (random projection).

Randomized Nyström Approximation

In the case that A is very large, [matrix-matrix products with \$A\$](#) are the bottleneck.

This motivates the [single-pass version](#) of the Nyström method.

Stabilized Single-Pass Nyström method [Tropp et al., 2017]

Given sym. PSD matrix A , target rank k

$G = \text{randn}(n, k)$

$[Q, \sim] = \text{qr}(G, 0)$

$Y = \mathbf{A}Q$

Compute shift ν ; $Y_\nu = Y + \nu Q$

$B = Q^T Y_\nu$

$C = \text{chol}((B + B^T)/2)$

Solve $F = Y_\nu / C$

$[U, \Sigma, \sim] = \text{svd}(F, 0)$

$\Theta = \max(0, \Sigma^2 - \nu I)$

Randomized Nyström Approximation

In the case that A is very large, **matrix-matrix products with A** are the bottleneck.

This motivates the **single-pass version** of the Nyström method.

Stabilized Single-Pass Nyström method [Tropp et al., 2017]

Given sym. PSD matrix A , target rank k

$G = \text{randn}(n, k)$

$[Q, \sim] = \text{qr}(G, 0)$

$Y = AQ$

Compute shift ν ; $Y_\nu = Y + \nu Q$

$B = Q^T Y_\nu$

$C = \text{chol}((B + B^T)/2)$

Solve $F = Y_\nu / C$

$[U, \Sigma, \sim] = \text{svd}(F, 0)$

$\Theta = \max(0, \Sigma^2 - \nu I)$

Can we **further reduce the cost** of the matrix-matrix product with A by using **low precision**?

Error Bounds

$$\|A - \hat{A}_N\|_2 = \|A - A_N + A_N - \hat{A}_N\|_2 \leq \underbrace{\|A - A_N\|_2}_{\substack{\text{exact} \\ \text{approximation} \\ \text{error}}} + \underbrace{\|A_N - \hat{A}_N\|_2}_{\substack{\text{finite precision} \\ \text{error}}}$$

Error Bounds

$$\|A - \hat{A}_N\|_2 = \|A - A_N + A_N - \hat{A}_N\|_2 \leq \underbrace{\|A - A_N\|_2}_{\text{exact approximation error}} + \underbrace{\|A_N - \hat{A}_N\|_2}_{\text{finite precision error}}$$

Deterministic bound [Gittens, Mahoney, 2016]:

$$\|A - A_N\|_2 \leq \lambda_{k+1} + \left\| \Sigma_2^{1/2} U_2^T Q (U_1 Q)^+ \right\|_2^2$$

with $A = [U_1 \ U_2] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} [U_1 \ U_2]^T$.

Error Bounds

$$\|A - \hat{A}_N\|_2 = \|A - A_N + A_N - \hat{A}_N\|_2 \leq \underbrace{\|A - A_N\|_2}_{\text{exact approximation error}} + \underbrace{\|A_N - \hat{A}_N\|_2}_{\text{finite precision error}}$$

Deterministic bound [Gittens, Mahoney, 2016]:

$$\|A - A_N\|_2 \leq \lambda_{k+1} + \left\| \Sigma_2^{1/2} U_2^T Q (U_1 Q)^+ \right\|_2^2$$

with $A = [U_1 \ U_2] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} [U_1 \ U_2]^T$.

Expected value bound [Frangella, Tropp, Udell, 2021]:

$$\mathbb{E} \|A - A_N\|_2 \leq \min_{2 \leq p \leq k-2} \left(\left(1 + \frac{2(k-p)}{p-1} \right) \lambda_{k-p+1} + \frac{2e^2 k}{p^2 - 1} \sum_{j=k-p+1}^n \lambda_j \right)$$

where $\lambda_i \geq \lambda_{i+1}$ are the eigenvalues of A .

Finite Precision Error Bound

Finite precision error: $A_N - \hat{A}_N$

Assumptions:

- A is stored in precision u_p and matrix-matrix product AQ is computed in precision u_p
- All other quantities stored and computed in precision $u \ll u_p$

Finite Precision Error Bound

Finite precision error: $A_N - \hat{A}_N$

Assumptions:

- A is stored in precision u_p and matrix-matrix product AQ is computed in precision u_p
- All other quantities stored and computed in precision $u \ll u_p$

[C., Daužickaitė, 2022]:

$$\|A_N - \hat{A}_N\|_2 \leq O(u_p)n^{5/2}\|A\|_2$$

Finite Precision Error Bound

Finite precision error: $A_N - \hat{A}_N$

Assumptions:

- A is stored in precision u_p and matrix-matrix product AQ is computed in precision u_p
- All other quantities stored and computed in precision $u \ll u_p$

[C., Daužickaitė, 2022]:

$$\|A_N - \hat{A}_N\|_2 \leq O(u_p)n^{5/2}\|A\|_2$$

Interpretation: $\|A_N - \hat{A}_N\|_2 \gtrsim \|A - A_N\|_2$ when

$$\frac{\lambda_{k+1}}{\lambda_1} \lesssim \sqrt{n}u_p$$

Finite Precision Error Bound

Finite precision error: $A_N - \hat{A}_N$

Assumptions:

- A is stored in precision u_p and matrix-matrix product AQ is computed in precision u_p
- All other quantities stored and computed in precision $u \ll u_p$

[C., Daužickaitė, 2022]:

$$\|A_N - \hat{A}_N\|_2 \leq O(u_p)n^{5/2}\|A\|_2$$

Interpretation: $\|A_N - \hat{A}_N\|_2 \gtrsim \|A - A_N\|_2$ when

$$\frac{\lambda_{k+1}}{\lambda_1} \lesssim \sqrt{nu_p}$$

The more approximate the low-rank representation, the lower the precision we can use!

Condition Number Bounds

Let $E = A - A_N$, $\mathcal{E} = A_N - \hat{A}_N$, and assume $(A + \mu I)$ is SPD.

Let

$$\hat{P}^{-1} = I - \hat{U}\hat{U}^T + (\hat{\lambda}_k + \mu)\hat{U}(\hat{\Theta} + \mu I)^{-1}\hat{U}^T$$

be the LMP preconditioner constructed using the mixed precision Nyström approximation $\hat{A}_N = \hat{U}\hat{\Theta}\hat{U}^T$.

Condition Number Bounds

Let $E = A - A_N$, $\mathcal{E} = A_N - \hat{A}_N$, and assume $(A + \mu I)$ is SPD.

Let

$$\hat{P}^{-1} = I - \hat{U}\hat{U}^T + (\hat{\lambda}_k + \mu)\hat{U}(\hat{\Theta} + \mu I)^{-1}\hat{U}^T$$

be the LMP preconditioner constructed using the mixed precision Nyström approximation $\hat{A}_N = \hat{U}\hat{\Theta}\hat{U}^T$.

Then

$$\max \left\{ 1, \frac{\hat{\lambda}_k + \mu - \|\mathcal{E}\|_2}{\mu + \lambda_{\min}(A)} \right\} \leq \kappa(\hat{P}^{-1/2}(A + \mu I)\hat{P}^{-1/2}) \leq 1 + \frac{\hat{\lambda}_k + \|E\|_2 + 2\|\mathcal{E}\|_2}{\mu - \|\mathcal{E}\|_2}$$

where the upper bound holds if $\mu > \|\mathcal{E}\|_2$.

Regardless of this constraint, if A is positive definite, then

$$\kappa(\hat{P}^{-1/2}(A + \mu I)\hat{P}^{-1/2}) \leq (\hat{\lambda}_k + \mu + \|E\|_2 + \|\mathcal{E}\|_2) \left(\frac{1}{\hat{\lambda}_k + \mu} + \frac{\|\mathcal{E}\|_2 + 1}{\lambda_{\min}(A) + \mu} \right).$$

Condition Number Bounds

Let $E = A - A_N$, $\mathcal{E} = A_N - \hat{A}_N$, and assume $(A + \mu I)$ is SPD.

Let

$$\hat{P}^{-1} = I - \hat{U}\hat{U}^T + (\hat{\lambda}_k + \mu)\hat{U}(\hat{\Theta} + \mu I)^{-1}\hat{U}^T$$

be the LMP preconditioner constructed using the mixed precision Nyström approximation $\hat{A}_N = \hat{U}\hat{\Theta}\hat{U}^T$.

Then

If $\mathcal{E} = 0$, reduces to bounds of [Frangella, Tropp, Udell, 2021] for exact case.

$$\max \left\{ 1, \frac{\hat{\lambda}_k + \mu - \|\mathcal{E}\|_2}{\mu + \lambda_{\min}(A)} \right\} \leq \kappa(\hat{P}^{-1/2}(A + \mu I)\hat{P}^{-1/2}) \leq 1 + \frac{\hat{\lambda}_k + \|E\|_2 + 2\|\mathcal{E}\|_2}{\mu - \|\mathcal{E}\|_2}$$

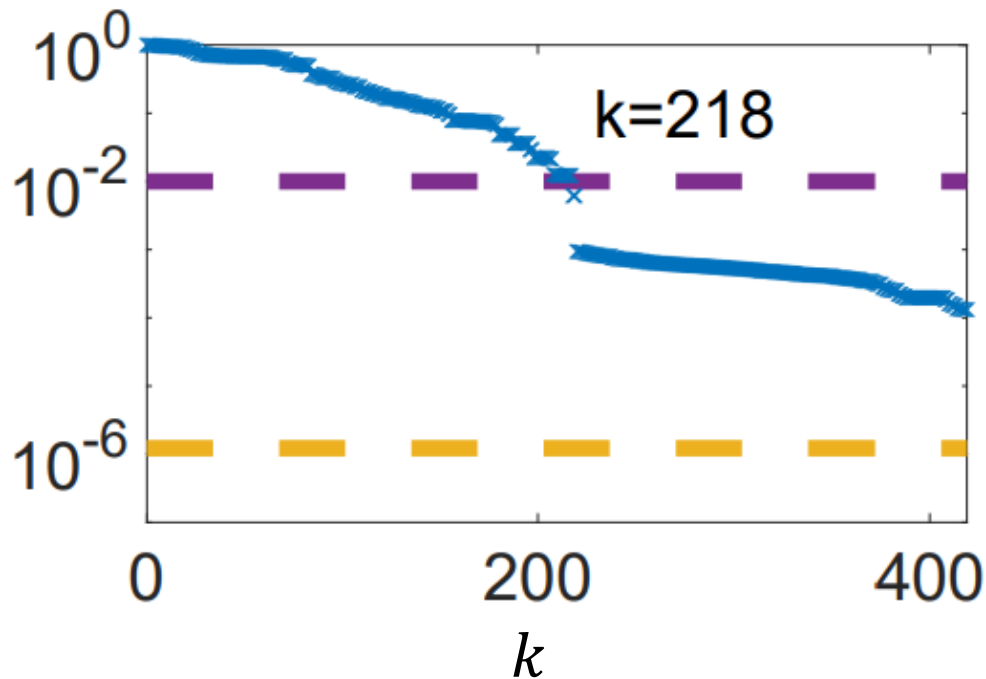
where the upper bound holds if $\mu > \|\mathcal{E}\|_2$.




Regardless of this constraint, if A is positive definite, then

$$\kappa(\hat{P}^{-1/2}(A + \mu I)\hat{P}^{-1/2}) \leq (\hat{\lambda}_k + \mu + \|E\|_2 + \|\mathcal{E}\|_2) \left(\frac{1}{\hat{\lambda}_k + \mu} + \frac{\|\mathcal{E}\|_2 + 1}{\lambda_{\min}(A) + \mu} \right).$$

Numerical Experiment

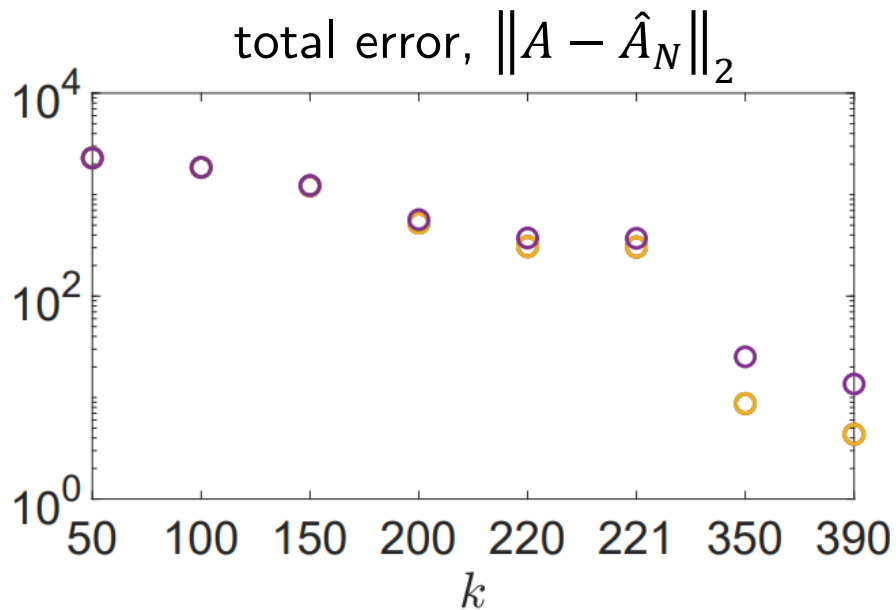
Matrix: bcsstm07, $n = 420$



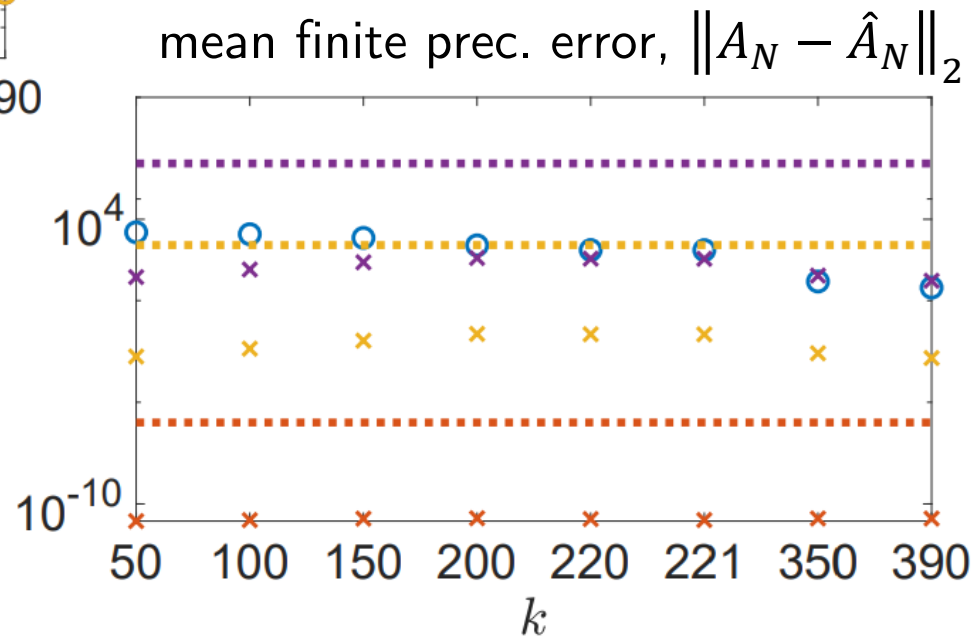
-  λ_{k+1}/λ_1
-  $\sqrt{n}u_p, u_p = \text{half}$
-  $\sqrt{n}u_p, u_p = \text{single}$

Numerical Experiment

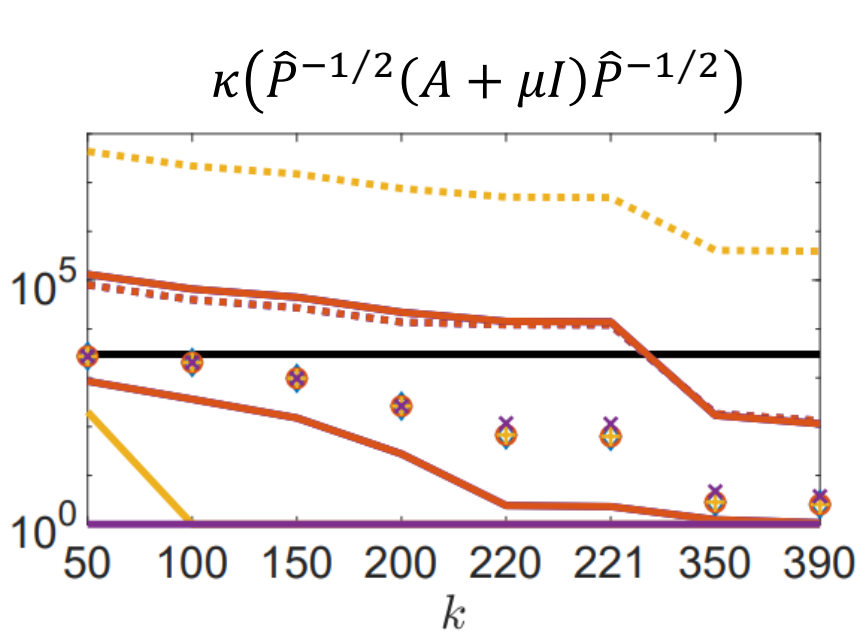
Matrix: bcsstm07, $n = 420$



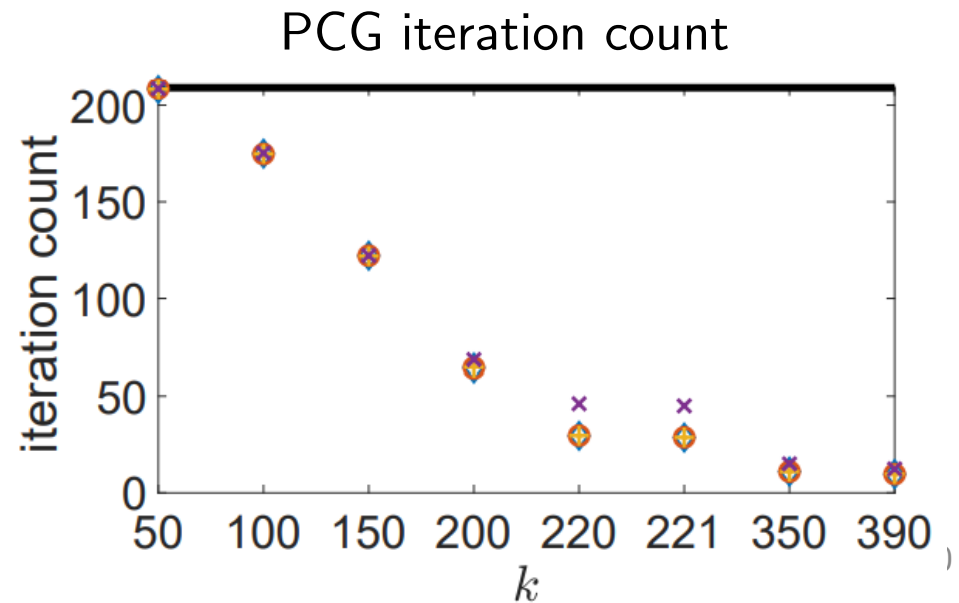
- exact
- mixed, $u_p = \text{half}$
- mixed, $u_p = \text{single}$
- mixed, $u_p = \text{double}$



Numerical Experiment



- unpreconditioned
- exact
- mixed, $u_p = \text{half}$
- mixed, $u_p = \text{single}$
- mixed, $u_p = \text{double}$



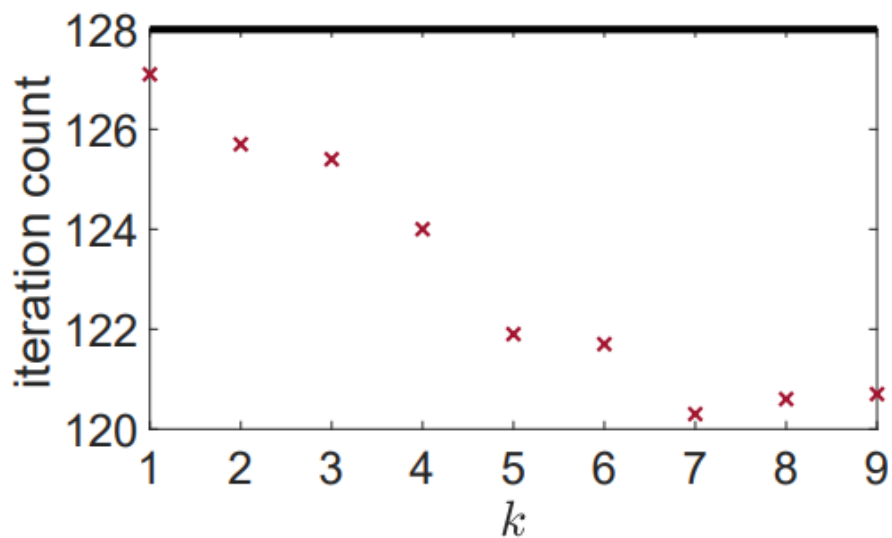
Summary and Takeaway

- We now have a multi-precision ecosystem
- Huge opportunities for using mixed precision in matrix computations
- But also big challenges!

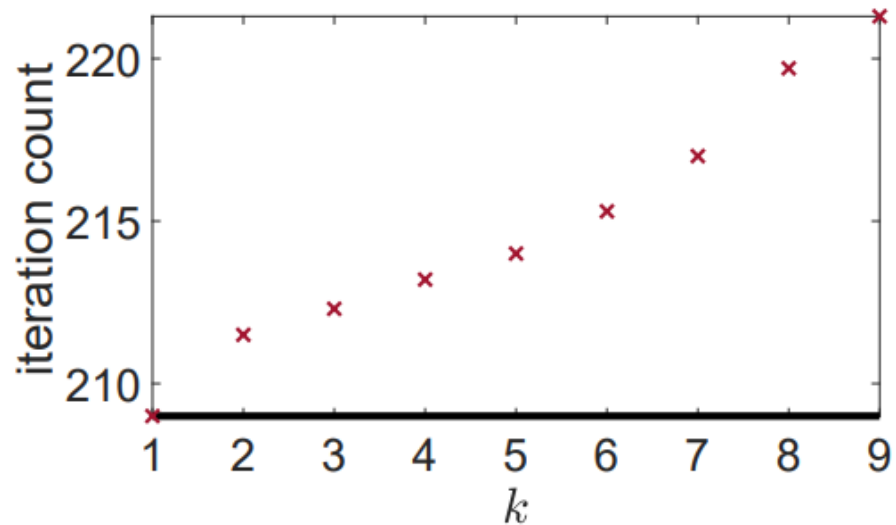
Thank You!

carson@karlin.mff.cuni.cz
www.karlin.mff.cuni.cz/~carson/

Quarter precision?



(e) Journals, iteration count



(f) bcsstm07, iteration count