Mixed precision strategies for preconditioned GMRES

Speaker: Bastien Vieublé
Coauthors: Alfredo Buttari, Xin Liu, Théo Mary

April 2025

GAMM 2025

What is GMRES?

Throughout the presentation, we focus on the Generalized Minimal RESidual (GMRES) algorithm.

Algorithm: GMRES(A, b, x_0, τ)

Require: $A \in \mathbb{R}^{n \times n}$, $b, x_0 \in \mathbb{R}^n$, $\tau \in \mathbb{R}$

```
r_0 = b - A x_0
β = ‖r_0‖,  v_1 = r_0/β,  k = 1
repeat
    w_k = A v_k
    for i = 1, ..., k do
        h_{i,k} = v_i^T w_k
        w_k = w_k - h_{i,k} v_i
    end for
    h_{k+1,k} = ‖w_k‖, v_{k+1} = w_k / h_{k+1,k}
    V_k = [v_1, ..., v_k]
    H_k = {h_{i,j}}_{1≤i≤j+1; 1≤j≤k}
    y_k = argmin_y ‖β e_1 - H_k y‖
    k = k + 1
until ‖β e_1 - H_k y_k‖ ≤ τ
x_k = x_0 + V_k y_k
```

What is GMRES?

Throughout the presentation, we focus on the Generalized Minimal RESidual (GMRES) algorithm.

- ➤ GMRES = Krylov-based iterative solver for the solution of general square linear systems $Ax = b$.

Algorithm: GMRES(A, b, x_0, τ)

Require: $A \in \mathbb{R}^{n \times n}$, $b, x_0 \in \mathbb{R}^n$, $\tau \in \mathbb{R}$

$r_0 = b - Ax_0$
$\beta = \|r_0\|, \; v_1 = r_0/\beta, \; k = 1$
repeat
  $w_k = Av_k$
  for $i = 1, \ldots, k$ **do**
    $h_{i,k} = v_i^T w_k$
    $w_k = w_k - h_{i,k} v_i$
  end for
  $h_{k+1,k} = \|w_k\|, v_{k+1} = w_k / h_{k+1,k}$
  $V_k = [v_1, \ldots, v_k]$
  $H_k = \{h_{i,j}\}_{1 \leq i \leq j+1; 1 \leq j \leq k}$
  $y_k = \text{argmin}_y \|\beta e_1 - H_k y\|$
  $k = k + 1$
until $\|\beta e_1 - H_k y_k\| \leq \tau$
$x_k = x_0 + V_k y_k$

What is GMRES?

Throughout the presentation, we focus on the Generalized Minimal RESidual (GMRES) algorithm.

- ➤ GMRES = Krylov-based iterative solver for the solution of **general square linear systems $Ax = b$**.
- ➤ Computes iteratively an orthonormal **Krylov basis V_k** through an Arnoldi process.

Algorithm: GMRES(A, b, x_0, τ)

Require: $A \in \mathbb{R}^{n\times n}$, $b, x_0 \in \mathbb{R}^n$, $\tau \in \mathbb{R}$

```
r_0 = b − A x_0
β = ||r_0||,  v_1 = r_0/β,  k = 1
repeat
    w_k = A v_k
    for i = 1, ..., k do
        h_{i,k} = v_i^T w_k
        w_k = w_k − h_{i,k} v_i
    end for
    h_{k+1,k} = ||w_k||, v_{k+1} = w_k / h_{k+1,k}
    V_k = [v_1, ..., v_k]
    H_k = {h_{i,j}}_{1≤i≤j+1; 1≤j≤k}
    y_k = argmin_y ||β e_1 − H_k y||
    k = k + 1
until ||β e_1 − H_k y_k|| ≤ τ
x_k = x_0 + V_k y_k
```

What is GMRES?

Throughout the presentation, we focus on the Generalized Minimal RESidual (GMRES) algorithm.

- GMRES = Krylov-based iterative solver for the solution of **general square linear systems $Ax = b$**.
- Computes iteratively an orthonormal **Krylov basis V_k** through an Arnoldi process.
- Chooses the vector x_k in $span\{V_k\}$ that **minimizes $\|Ax_k - b\|$**.

Algorithm: GMRES(A, b, x_0, τ)

Require: $A \in \mathbb{R}^{n\times n}$, $b, x_0 \in \mathbb{R}^n$, $\tau \in \mathbb{R}$

```
r_0 = b − A x_0
β = ||r_0||,  v_1 = r_0/β,  k = 1
repeat
    w_k = A v_k
    for i = 1, ..., k do
        h_{i,k} = v_i^T w_k
        w_k = w_k − h_{i,k} v_i
    end for
    h_{k+1,k} = ||w_k||, v_{k+1} = w_k / h_{k+1,k}
    V_k = [v_1, ..., v_k]
    H_k = {h_{i,j}}_{1≤i≤j+1; 1≤j≤k}
    y_k = argmin_y ||β e_1 − H_k y||
    k = k + 1
until ||β e_1 − H_k y_k|| ≤ τ
x_k = x_0 + V_k y_k
```

What is GMRES?

Throughout the presentation, we focus on the Generalized Minimal RESidual (GMRES) algorithm.

- ➤ GMRES = Krylov-based iterative solver for the solution of **general square linear systems $Ax = b$**.
- ➤ Computes iteratively an orthonormal **Krylov basis V_k** through an Arnoldi process.
- ➤ Chooses the vector x_k in $span\{V_k\}$ that **minimizes $\|Ax_k - b\|$**.
- ➤ **Reiterate** until x_k is a satisfactory approximant of x.

Algorithm: GMRES(A, b, x_0, τ)

Require: $A \in \mathbb{R}^{n \times n}$, $b, x_0 \in \mathbb{R}^n$, $\tau \in \mathbb{R}$

```
r_0 = b − A x_0
β = ||r_0||,  v_1 = r_0/β,  k = 1
repeat
    w_k = A v_k
    for i = 1, . . . , k do
        h_{i,k} = v_i^T w_k
        w_k = w_k − h_{i,k} v_i
    end for
    h_{k+1,k} = ||w_k||, v_{k+1} = w_k / h_{k+1,k}
    V_k = [v_1, . . . , v_k]
    H_k = {h_{i,j}}_{1≤i≤j+1;1≤j≤k}
    y_k = argmin_y ||β e_1 − H_k y||
    k = k + 1
until ||β e_1 − H_k y_k|| ≤ τ
x_k = x_0 + V_k y_k
```

What is mixed precision?

Commonly available floating point arithmetics:

	ID	Signif. bits	Exp. bits	Range	Unit roundoff u
fp128	Q	113	15	$10^{\pm 4932}$	1×10^{-34}
double-fp64	DD	107	11	$10^{\pm 308}$	6×10^{-33}
fp64	D	53	11	$10^{\pm 308}$	1×10^{-16}
fp32	S	24	8	$10^{\pm 38}$	6×10^{-8}
tfloat32	T	11	8	$10^{\pm 38}$	5×10^{-4}
fp16	H	11	5	$10^{\pm 5}$	5×10^{-4}
bfloat16	B	8	8	$10^{\pm 38}$	4×10^{-3}
fp8 (E4M3)	R	4	4	$10^{\pm 2}$	6.3×10^{-2}
fp8 (E5M2)	R*	3	5	$10^{\pm 5}$	1.3×10^{-1}

The low precision arithmetics are less accurate BUT are faster, consume less memory and energy.

What is preconditioning?

Principle: Transform the original linear system $Ax = b$ into an easier one to solve.

- Left:
$$M^{-1}Ax = M^{-1}b, \qquad \kappa(M^{-1}A) \ll \kappa(A).$$
- Right:
$$AM^{-1}u = b, \qquad x = M^{-1}u, \qquad \kappa(AM^{-1}) \ll \kappa(A).$$
- Flexible: A variant of right-preconditioning allowing the preconditioner $M^{(i)}$ to vary from an iteration to another. We will consider $M^{(i)} = M$ in this study.
- Split:
$$M_L^{-1}AM_R^{-1}u = M_L^{-1}b, \qquad x = M_R^{-1}u, \qquad \kappa(M_L^{-1}AM_R^{-1}) \ll \kappa(A).$$

What is preconditioning?

Principle: Transform the original linear system $Ax = b$ into an easier one to solve.

- Left:
$$M^{-1}Ax = M^{-1}b, \qquad \kappa(M^{-1}A) \ll \kappa(A).$$
- Right:
$$AM^{-1}u = b, \qquad x = M^{-1}u, \qquad \kappa(AM^{-1}) \ll \kappa(A).$$
- Flexible: A variant of right-preconditioning allowing the preconditioner $M^{(i)}$ to vary from an iteration to another. We will consider $M^{(i)} = M$ in this study.
- Split:
$$M_L^{-1}AM_R^{-1}u = M_L^{-1}b, \qquad x = M_R^{-1}u, \qquad \kappa(M_L^{-1}AM_R^{-1}) \ll \kappa(A).$$

"*The stability of split-preconditioned FGMRES in four precisions*" by **Erin Carson and Ieva Daužickaitė**, 2024, ETNA.

Mixed precision layout for preconditioned GMRES

Algorithm: Left(A,M^{-1}, b, x_0, τ)

$r_0 = b - Ax_0$
$s_0 = M^{-1}r_0$
$\beta = \|s_0\|,\ v_1 = s_0/\beta,\ k = 1$
repeat
  $z_k = Av_k$
  $w_k = M^{-1}z_k$
  for $i = 1, \ldots, k$ **do**
    $h_{i,k} = v_i^T w_k$
    $w_k = w_k - h_{i,k}v_i$
  end for
  $h_{k+1,k} = \|w_k\|, v_{k+1} = w_k/h_{k+1,k}$
  $V_k = [v_1, \ldots, v_k]$
  $H_k = \{h_{i,j}\}_{1<i<j+1;1<j<k}$
  $y_k = \textbf{argmin}_y \|\beta e_1 - H_k y\|$
  $k = k + 1$
until $\|\beta e_1 - H_k y_k\| \leq \tau$
$x_k = x_0 + V_k y_k$

Algorithm: Right(A,M^{-1}, b, x_0, τ)

$r_0 = b - Ax_0$
$\beta = \|r_0\|,\ v_1 = r_0/\beta,\ k = 1$
repeat
  $z_k = M^{-1}v_k$
  $w_k = Az_k$
  for $i = 1, \ldots, k$ **do**
    $h_{i,k} = v_i^T w_k$
    $w_k = w_k - h_{i,k}v_i$
  end for
  $h_{k+1,k} = \|w_k\|, v_{k+1} = w_k/h_{k+1,k}$
  $V_k = [v_1, \ldots, v_k]$
  $H_k = \{h_{i,j}\}_{1<i<j+1;1<j<k}$
  $y_k = \textbf{argmin}_y \|\beta e_1 - H_k y\|$
  $k = k + 1$
until $\|\beta e_1 - H_k y_k\| \leq \tau$
$d_k = V_k y_k$
$x_k = x_0 + M^{-1}d_k$

Mixed precision layout for preconditioned GMRES

Algorithm: Left(A,M^{-1}, b, x_0, τ)

$r_0 = b - Ax_0$ — u_a
$s_0 = M^{-1} r_0$ — u_m
$\beta = \|s_0\|,\ v_1 = s_0/\beta,\ k = 1$ — u_g
repeat
  $z_k = Av_k$ — u_a
  $w_k = M^{-1} z_k$ — u_m
  for $i = 1, \ldots, k$ **do**
    $h_{i,k} = v_i^T w_k$ — u_g
    $w_k = w_k - h_{i,k} v_i$ — u_g
  end for
  $h_{k+1,k} = \|w_k\|, v_{k+1} = w_k / h_{k+1,k}$ — u_g
  $V_k = [v_1, \ldots, v_k]$
  $H_k = \{h_{i,j}\}_{1 \leq i \leq j+1; 1 \leq j \leq k}$
  $y_k = \text{argmin}_y \|\beta e_1 - H_k y\|$ — u_g
  $k = k + 1$
until $\|\beta e_1 - H_k y_k\| \leq \tau$

$x_k = x_0 + V_k y_k$ — u_g

Algorithm: Right(A,M^{-1}, b, x_0, τ)

$r_0 = b - Ax_0$ — u_a

$\beta = \|r_0\|,\ v_1 = r_0/\beta,\ k = 1$ — u_g
repeat
  $z_k = M^{-1} v_k$ — u_m
  $w_k = Az_k$ — u_a
  for $i = 1, \ldots, k$ **do**
    $h_{i,k} = v_i^T w_k$ — u_g
    $w_k = w_k - h_{i,k} v_i$ — u_g
  end for
  $h_{k+1,k} = \|w_k\|, v_{k+1} = w_k / h_{k+1,k}$ — u_g
  $V_k = [v_1, \ldots, v_k]$
  $H_k = \{h_{i,j}\}_{1 \leq i \leq j+1; 1 \leq j \leq k}$
  $y_k = \text{argmin}_y \|\beta e_1 - H_k y\|$ — u_g
  $k = k + 1$
until $\|\beta e_1 - H_k y_k\| \leq \tau$
$d_k = V_k y_k$ — u_g
$x_k = x_0 + M^{-1} d_k$ — u_m

Mixed precision layout for preconditioned GMRES

Algorithm: Left(A,M^{-1}, b, x_0, τ)

$r_0 = b - Ax_0$	u_a
$s_0 = M^{-1} r_0$	u_m
$\beta = \|s_0\|,\ v_1 = s_0/\beta,\ k = 1$	u_g
repeat	
$\quad z_k = Av_k$	u_a
$\quad w_k = M^{-1} z_k$	u_m
$\quad$ **for** $i = 1, \ldots, k$ **do**	
$\qquad h_{i,k} = v_i^T w_k$	u_g
$\qquad w_k = w_k - h_{i,k} v_i$	u_g
$\quad$ **end for**	
$\quad h_{k+1,k} = \|w_k\|, v_{k+1} = w_k/h_{k+1,k}$	u_g
$\quad V_k = [v_1, \ldots, v_k]$	
$\quad H_k = \{h_{i,j}\}_{1 \le i \le j+1; 1 \le j \le k}$	
$\quad y_k = \mathbf{argmin}_y \|\beta e_1 - H_k y\|$	u_g
$\quad k = k + 1$	
until $\|\beta e_1 - H_k y_k\| \le \tau$	
$x_k = x_0 + V_k y_k$	u_g

Algorithm: Flexible(A,M^{-1}, b, x_0, τ)

$r_0 = b - Ax_0$	u_a
$\beta = \|r_0\|,\ v_1 = r_0/\beta,\ k = 1$	u_g
repeat	
$\quad z_k = M^{-1} v_k$	u_m
$\quad w_k = Az_k$	u_a
$\quad$ **for** $i = 1, \ldots, k$ **do**	
$\qquad h_{i,k} = v_i^T w_k$	u_g
$\qquad w_k = w_k - h_{i,k} v_i$	u_g
$\quad$ **end for**	
$\quad h_{k+1,k} = \|w_k\|, v_{k+1} = w_k/h_{k+1,k}$	u_g
$\quad V_k = [v_1, \ldots, v_k],\ Z_k = [z_1, \ldots, z_k]$	
$\quad H_k = \{h_{i,j}\}_{1 \le i \le j+1; 1 \le j \le k}$	
$\quad y_k = \mathbf{argmin}_y \|\beta e_1 - H_k y\|$	u_g
$\quad k = k + 1$	
until $\|\beta e_1 - H_k y_k\| \le \tau$	
$x_k = x_0 + Z_k d_k$	u_m

State-of-the-art

How to read: u_a, u_m, and u_g refer to the precision or the unit roundoff. If we write $u_a \ll u_g$, it means u_a is an higher precision than u_g.

State-of-the-art

How to read: u_a, u_m, and u_g refer to the precision or the unit roundoff. If we write $u_a \ll u_g$, it means u_a is an higher precision than u_g.

- $u_a = u_m \ll u_g$: Applying A and M^{-1} in high precision to improve accuracy and robustness. Existing studies are dedicated to left-preconditioned GMRES with LU and QR-based preconditioners, and SPAI preconditioners.
  - "*A New Analysis of Iterative Refinement and Its Application to Accurate Solution of Ill-Conditioned Sparse Linear Systems*" by **E. Carson and N. J. Higham**, 2017, SIAM SISC.
  - "*Three-Precision GMRES-Based Iterative Refinement for Least Squares Problems*" by **E. Carson, N. J. Higham, and S. Pranesh**, 2020, SIAM SISC.
  - "*Five-Precision GMRES-Based Iterative Refinement*" by **P. Amestoy, A. Buttari, N. J. Higham, J.-Y. LExcellent, T. Mary, and B. Vieublé**, 2024, SIAM SIMAX.

- $u_a = u_g \ll u_m$: Applying M^{-1} in a lower precision to improve performance. Existing studies are dedicated to flexible-preconditioned GMRES.
  - "*Using FGMRES to obtain backward stability in mixed precision*" by **M. Arioli and I. S. Duff**, 2008, ETNA.
  - "*The stability of split-preconditioned FGMRES in four precisions*" by **E. Carson and I. Daužickaitė**, 2024, ETNA.

Goal

We want to answer the question:

What are all the numerically meaningful ways to set u_g, u_m, and u_a?

Numerically meaningful means there is a **tradeoff** between employing computationally efficient **low precision**, the **accuracy** of the computed solution, and **number of iterations**.

Error analysis with generic preconditioners

Two main important numerical properties of GMRES: **convergence rate** and **attainable accuracies**.

We cannot derive strong theoretical result on the convergence rate.

"Any nonincreasing convergence curve is possible for GMRES" by **A. Greenbaum, V. Pták and Z. Strakoš**, 1996, SIAM SIMAX.

$\Rightarrow$ We focus on the attainable accuracy/error.

Error analysis with generic preconditioners

Two main important numerical properties of GMRES: **convergence rate** and **attainable accuracies**.

We cannot derive strong theoretical result on the convergence rate.
"Any nonincreasing convergence curve is possible for GMRES" by **A. Greenbaum, V. Pták and Z. Strakoš**, 1996, SIAM SIMAX.

$\Rightarrow$ We focus on the attainable accuracy/error.

Two main ingredients for bounds on the attainable error:

- A generic preconditioner model

$$\mathrm{fl}(M^{-1}v_j) = (M^{-1} + \Delta M^{(j)})v_j, \qquad \|\Delta M^{(j)}\|_F \leq c(n,k)\mathbf{u_m}\eta\|M^{-1}\|_F,$$

- The modular framework for the error analysis of GMRES.
  "A modular framework for the backward error analysis of GMRES" by **A. Buttari, N. J. Higham, T. Mary, and B. Vieublé**, 2024, preprint.

Simplified bounds on the forward error

Let's call $\widehat{x}$ the computed solution and x the exact solution, we define the forward error as

$$\frac{\|\widehat{x} - x\|_2}{\|x\|_2}.$$

- Left:

$$\boldsymbol{u_g}\kappa\left(M^{-1}A\right) + \boldsymbol{u_m}\rho + \boldsymbol{u_a}\kappa\left(A\right),$$

where $\rho \leq \kappa(M^{-1}A)\kappa(M)\|Av_j\|_2/\|A\|_F$.

- Right:

$$\boldsymbol{u_g}\kappa\left(AM^{-1}\right)\kappa\left(M\right) + \boldsymbol{u_m}\kappa\left(M\right) + \boldsymbol{u_a}\kappa\left(A\right).$$

- Flexible:

$$\boldsymbol{u_g}\kappa\left(AM^{-1}\right)\kappa\left(M\right) + \boldsymbol{u_a}\kappa\left(A\right).$$

List of the different strategies

	Left	Right	Flexible
$u_a = u_g = u_m$	exists	exists	exists
$u_a = u_m \ll u_g$	exists	new	new
$u_a = u_g \ll u_m$	—	new	exists
$u_a \ll u_g = u_m$	new	new	new
$u_a \ll u_g \ll u_m$	—	new	new
$u_a \ll u_m \ll u_g$	new	new	new

We choose $u_a \leq \min(u_g, u_m)$ to reduce the overall amount of combinations considered.

List of the different strategies

	Left	Right	Flexible
$u_a = u_g = u_m$	exists	exists	exists
$u_a = u_m \ll u_g$	exists	new	new
$u_a = u_g \ll u_m$	—	new	exists
$u_a \ll u_g = u_m$	new	new	new
$u_a \ll u_g \ll u_m$	—	new	new
$u_a \ll u_m \ll u_g$	new	new	new

We choose $u_a \leq \min(u_g, u_m)$ to reduce the overall amount of combinations considered.

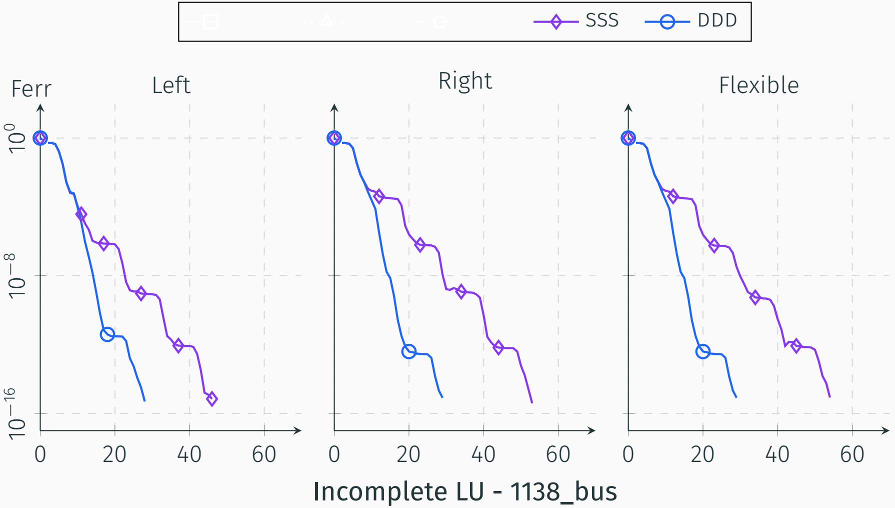
Experimental settings

➤ We run and compare the mixed precision strategies on various SuiteSparse matrices with various practical preconditioners.

➤ We employ restart (equivalent to iterative refinement) to improve all the solutions to the same prescribed accuracy

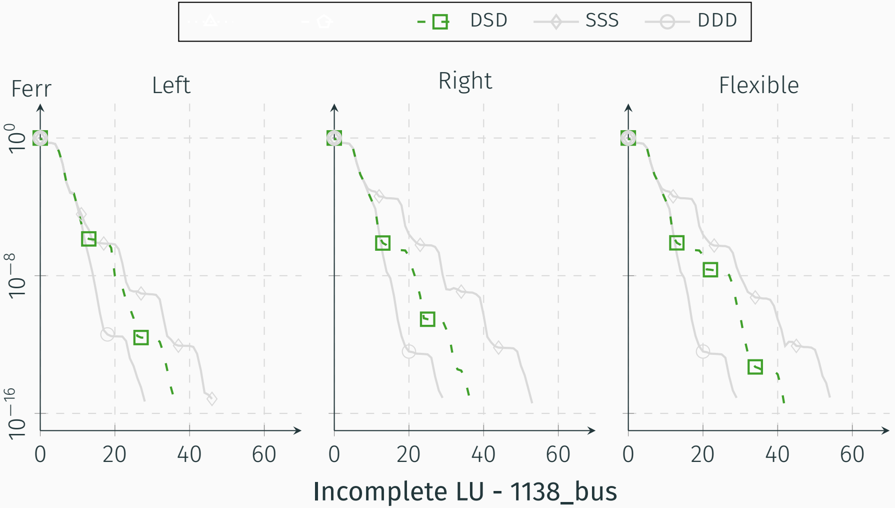
$$\frac{\|x - \widehat{x}\|_2}{\|x\|_2} \leq 10^{-15}.$$

$u_a = u_g = u_m$



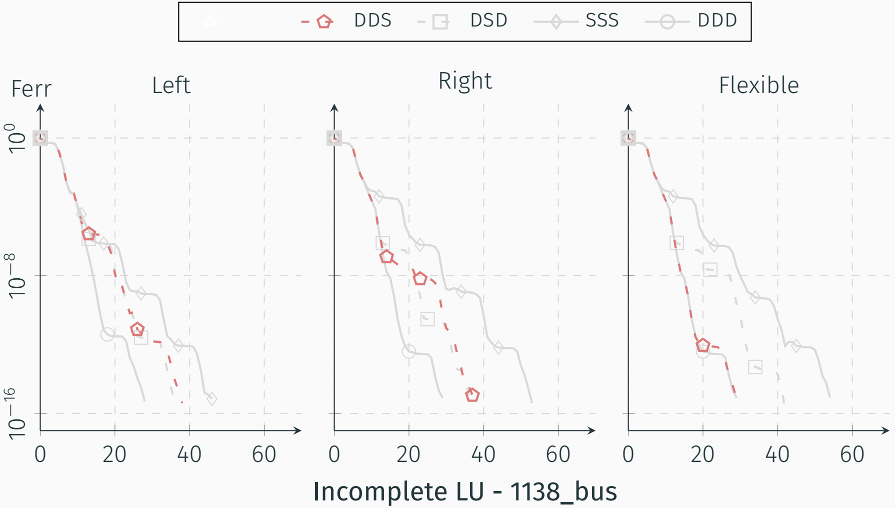
In the legend, a combination of precisions is defined by a triplet (u_a, u_g, u_m). E.g., DSD means $u_a = \text{D}$, $u_g = \text{S}$, $u_m = \text{D}$.

$u_a = u_m \ll u_g$



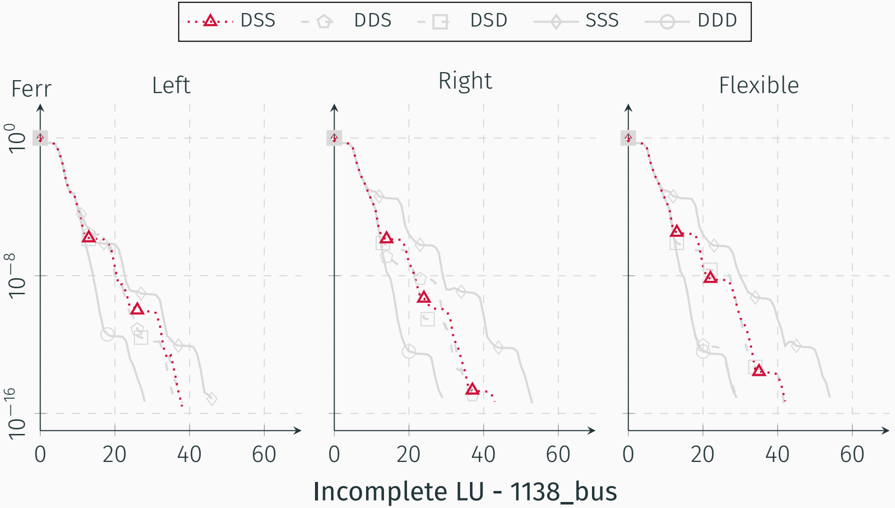
In the legend, a combination of precisions is defined by a triplet (u_a, u_g, u_m). E.g., DSD means $u_a = \mathrm{D}$, $u_g = \mathrm{S}$, $u_m = \mathrm{D}$.

$u_a = u_g \ll u_m$



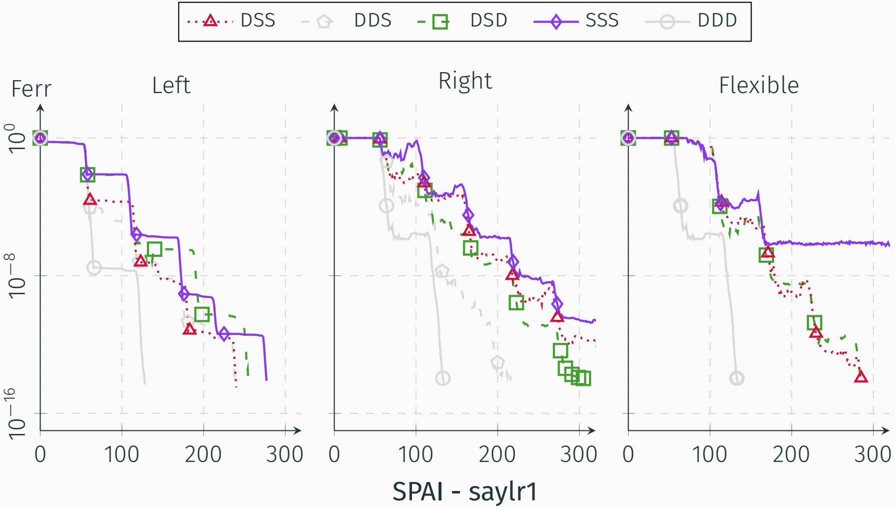
In the legend, a combination of precisions is defined by a triplet (u_a, u_g, u_m). E.g., DSD means $u_a = \text{D}$, $u_g = \text{S}$, $u_m = \text{D}$.

$u_a \ll u_g = u_m$



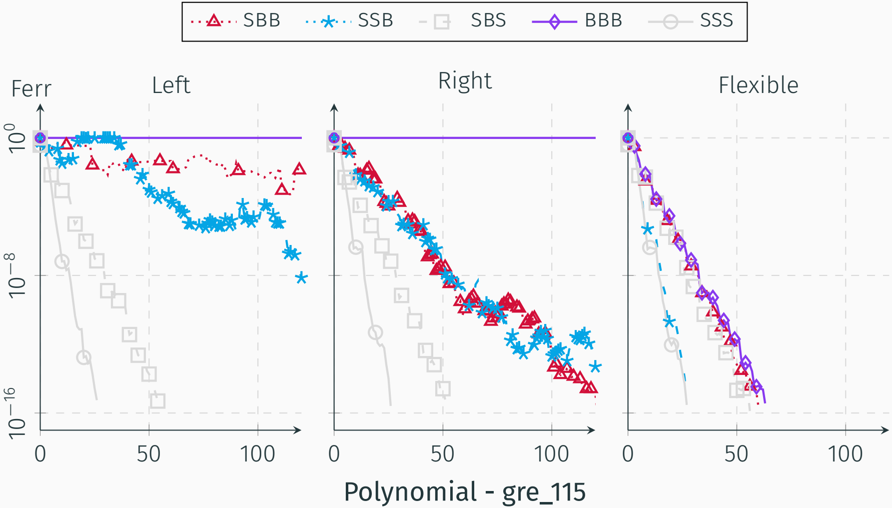
In the legend, a combination of precisions is defined by a triplet (u_a, u_g, u_m). E.g., DSD means $u_a = \text{D}$, $u_g = \text{S}$, $u_m = \text{D}$.

Low precision u_g



In the legend, a combination of precisions is defined by a triplet (u_a, u_g, u_m). E.g., DSD means u_a = D, u_g = S, u_m = D.

Low precision u_m



In the legend, a combination of precisions is defined by a triplet (u_a, u_g, u_m). E.g., DSD means $u_a = \text{D}$, $u_g = \text{S}$, $u_m = \text{D}$.

Conclusion

Takeaways

- ➤ We derived the most descriptive **bounds on the attainable forward error** for left-, right-, and flexible-preconditioned GMRES.
- ➤ We **identified possible mixed precision strategies** to apply the preconditioners in GMRES. They present different tradeoffs between performance and accuracy/robustness.
- ➤ We highlighted that in mixed precision the **difference between left-, right-, and flexible-preconditioning is critical**.

Future work: High performance implementations of some of these mixed precision strategies to solve large linear systems from industrial applications.

"Mixed precision strategies for preconditioned GMRES: a comprehensive analysis" by **A. Buttari, X. Liu, T. Mary, and B. Vieublé**, 2025, incoming.