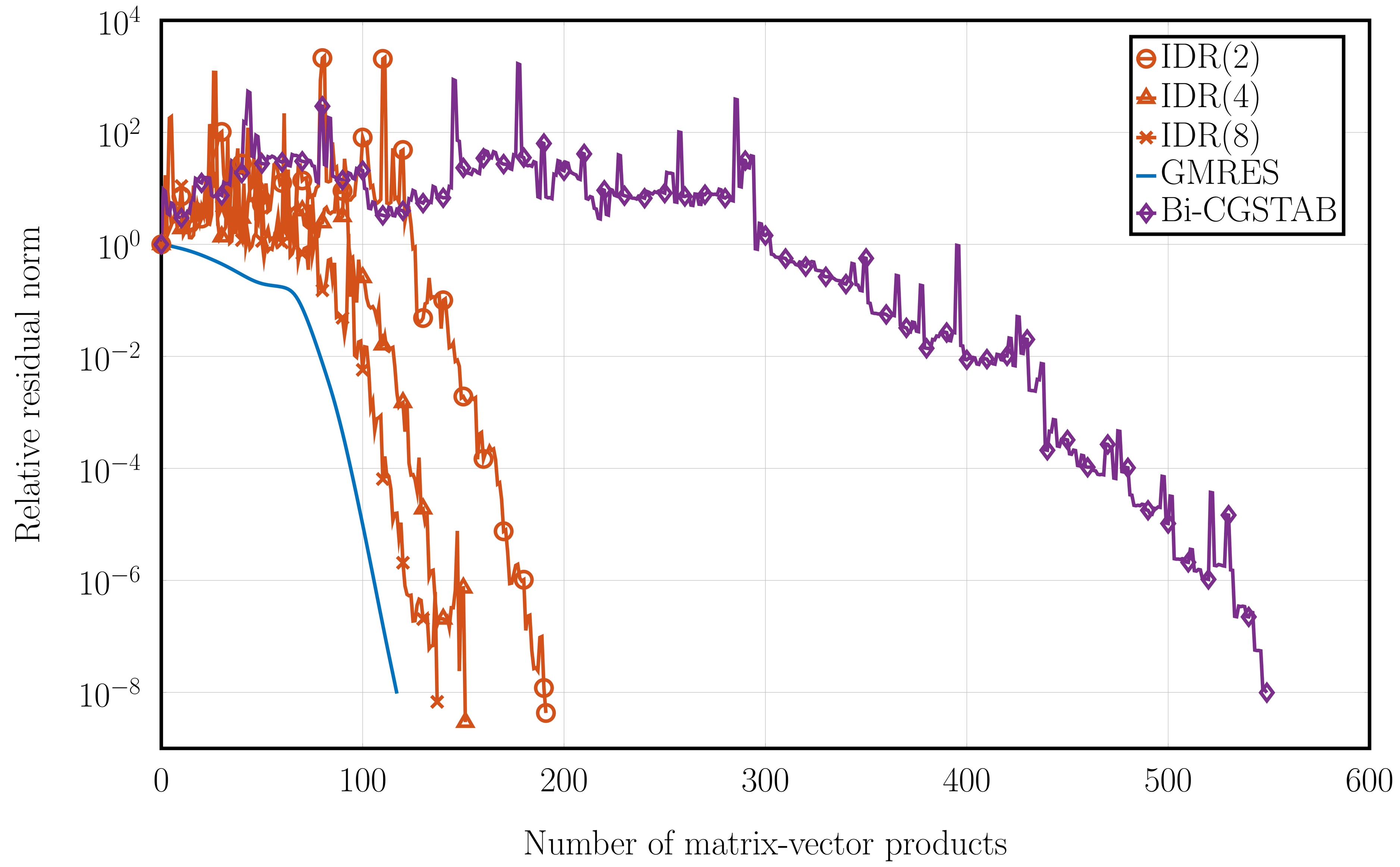
A Set of Fortran 90 and Python Routines for Solving Linear Equations with IDR(s)

Reinaldo Astudillo, Manuel Baumann, and Martin B. van Gijzen

Some IDR facts

- IDR(s) is a Krylov subspace method designed for solving large-scale linear systems of the form
$$A\mathbf{x} = \mathbf{b}, \quad A \in \mathbb{C}^{N\times N}.$$
- Based on $(s+2)$-term recursion: limited memory method.
- No restrictive assumptions on system matrix A.
- Finite termination after $N + N/s$ matrix-vector products.



P. Sonneveld and M. B. van Gijzen. *IDR(s): A Family of Simple and Fast Algorithms for Solving Large Nonsymmetric Systems of Linear Equations*, SIAM J. Sci. Comput., 31(2), 1035–1062, 2008

IDR(s) for linear matrix equations

$$\sum_{j=1}^{k} A_j X B_j^{\mathsf{T}} = C, \quad \text{and} \quad \begin{cases} A\mathbf{x} &= \mathbf{b} \\ A^{\mathsf{T}}\hat{\mathbf{x}} &= \hat{\mathbf{b}} \end{cases}$$

R. Astudillo and M. B. van Gijzen. *Induced Dimension Reduction method for solving linear matrix equations*, Delft University of Technology, TR-05, 2015

IDR(s) for shifted linear systems

$$(A - \sigma_j I)\mathbf{x}_j = \mathbf{b}, \quad j = 1, 2, \ldots$$

M. Baumann and M. B. van Gijzen. *Nested Krylov methods for shifted linear systems*, SIAM J. Sci. Comput. [in press]

New code design

Recently developed IDR version:

- flexible user interface via `types`
- available for download in many languages

Fortran 90

Matlab

Julia

Python

TU Delft
Delft University of Technology