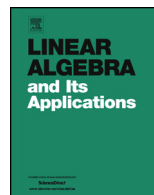




Contents lists available at ScienceDirect

Linear Algebra and its Applications

journal homepage: www.elsevier.com/locate/laa

Towards understanding CG and GMRES through examples

Erin Carson^{a,*}, Jörg Liesen^b, Zdeněk Strakoš^{a,2}^a Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic^b Institute of Mathematics, TU Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany

ARTICLE INFO

Article history:

Received 9 June 2023

Received in revised form 31 January 2024

Accepted 3 April 2024

Available online 8 April 2024

Submitted by V. Mehrmann

MSC:

15A60

65F10

65F35

Keywords:

Krylov subspace methods

Method of moments

CG method

GMRES method

Iterative methods

Convergence analysis

ABSTRACT

When the conjugate gradient (CG) method for solving linear algebraic systems was formulated about 70 years ago by Lanczos, Hestenes, and Stiefel, it was considered an iterative process possessing a mathematical finite termination property. With the deep insight of the original authors, CG was placed into a very rich mathematical context, including links with Gauss quadrature and continued fractions. The optimality property of CG was described via a normalized weighted polynomial least squares approximation to zero. This highly nonlinear problem explains the adaptation of CG iterates to the given data. Karush and Hayes immediately considered CG in infinite dimensional Hilbert spaces and investigated its superlinear convergence. Since then, the view of CG, as well as other Krylov subspace methods developed in the meantime, has changed. Today these methods are considered primarily as computational tools, and their behavior is typically characterized using linear upper bounds, or heuristics based on clustering of eigenvalues. Such simplifications limit the math-

* Corresponding author.

E-mail addresses: carson@karlin.mff.cuni.cz (E. Carson), liesen@math.tu-berlin.de (J. Liesen), strakos@karlin.mff.cuni.cz (Z. Strakoš).

¹ Supported by the Charles University Research Centre program No. UNCE/24/SCI/005, the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration, and by the European Union (ERC, inEXASCALE, 101075632). Views and opinions expressed are those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

² Supported by the Nečas Center for Mathematical Modeling.

<https://doi.org/10.1016/j.laa.2024.04.003>

0024-3795/© 2024 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY-NC license (<http://creativecommons.org/licenses/by-nc/4.0/>).

Rounding error analysis
Polynomial approximation problems

emathical understanding of Krylov subspace methods, and also negatively affect their practical application.

This paper offers a different perspective. Focusing on CG and the generalized minimal residual (GMRES) method, it presents mathematically important as well as practically relevant phenomena that uncover their behavior *through a discussion of computed examples*. These examples provide an easily accessible approach that enables understanding of the methods, while pointers to more detailed analyses in the literature are given. This approach allows readers to choose the level of depth and thoroughness appropriate for their intentions. Some of the points made in this paper illustrate well known facts. Others challenge mainstream views and explain existing misunderstandings. Several points refer to recent results leading to open problems. We consider CG and GMRES crucially important for the mathematical understanding, further development, and practical applications also of other Krylov subspace methods. The paper additionally addresses the motivation of preconditioning.

© 2024 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY-NC license (<http://creativecommons.org/licenses/by-nc/4.0/>).

1. Introduction

Taking the 1952 landmark paper of Hestenes and Stiefel [53] on the conjugate gradient method (CG) as their historical starting point, Krylov subspace methods for solving linear algebraic systems $Ax = b$ have been around for more than 70 years. The work of Hestenes and Stiefel, together with the papers from the early 1950s by Lanczos [62–64], which approach the same topic from a slightly different perspective, crowned the effort of many researchers aiming to find new ways of solving systems of linear algebraic equations using contemporary computers. These works were immediately followed by the description and investigation of the CG method in the infinite dimensional Hilbert space setting, for example by Karush [57] and Hayes [52].³

Over the last 70 years, tens of thousands of research articles on the derivation, analysis, or applications of Krylov subspace methods have been published by authors coming from the most diverse scientific backgrounds. Numerous different Krylov subspace methods have been developed. Their names typically give very brief descriptions of their algorithmic properties, and the methods then are referred to by acronyms that abbreviate these names. Examples include, besides CG, the biconjugate gradient (BiCG) method [23] (with the idea of biconjugation going back to Lanczos [62,63]), the stabilized biconjugate gradient (BiCG-Stab) method [109], the full orthogonalization method (FOM) [94],

³ Krylov subspaces are named after A. N. Krylov, who in 1931 published a paper [58] that described a method for solving a secular equation determining the frequency of small oscillations of mechanical systems. Algebraic formulations of his method were subsequently published by Luzin in 1931 [71] and Gantmacher in 1934 [27]. A detailed account of the life and work of A. N. Krylov, and of the early development of Krylov subspace methods can be found, e.g., in [67, Section 2.5.7].

the minimal residual (MINRES) method [85], the generalized minimal residual (GMRES) method [97], the quasi-minimal residual (QMR) method [26], and related methods like the induced dimension reduction (IDR) method [112]. The majority of publications on Krylov subspace methods focus on algorithmic techniques, i.e., on construction of methods or preconditioners. A minority focus on mathematical principles and analysis of the methods in a wider context. Comprehensive treatments of the area can be found in many monographs and survey articles that are devoted entirely, or at least to a large extent, to Krylov subspace methods; see, e.g., [44,67,73,75,96,110] and [3,25,36,68,77], respectively.

Instead of adding yet another technical and algorithmic overview or taxonomy of Krylov subspace methods, this paper approaches the methods through investigation of important and practically relevant phenomena that uncover their mathematical foundations. This leads to understanding their behavior, and it also allows the clarification of persisting misunderstandings and issues that still remain open. We use *computed examples* for this purpose. Many of these examples can be found scattered throughout the existing literature. But their presentation and organization in this paper represents an entirely different approach than in the monographs and survey papers mentioned above. We have organized everything around what we consider *the main points for the understanding of Krylov subspace methods*.

We believe that such an approach and the perspective offered by this paper can help students, researchers, and practitioners to gain additional insights into Krylov subspace methods, and that in this way insight can be gained more easily than through extensive technical expositions. The text gives many references to the literature containing more detailed descriptions and analysis. We have selected works that underline the presented points and provide technical and historical background for the given arguments. Readers can thus get involved in the subject in the way and to the depth they find appropriate.

We focus on the CG method and the GMRES method, which have evolved as the standard iterative methods for solving linear algebraic systems with symmetric positive definite and general (nonsymmetric) matrices, respectively. Their understanding is a prerequisite for understanding other Krylov subspace methods, as well as for recent and possible future developments that involve them. In each of the computed examples, we first describe the setup as transparently as possible. We then describe the phenomena observable in the computed figures, followed by an explanation that usually contains pointers to further research literature. Throughout the paper we purposely use simple data (matrices and right-hand sides), so that the examples can be easily reproduced. Clearly, difficulties that can be observed on small model problems usually do not disappear in problems coming from real-world applications. On the other hand, numerical efficiency demonstrated on small model problems may not materialize in large-scale computations. We have this in mind and aspire not to introduce distorted arguments.

Before starting with CG and GMRES, we would like to present a point that is often neglected and that is most important for the understanding of Krylov subspace meth-

ods. We consider a linear algebraic system $Ax = b$ and an initial vector x_0 . Krylov subspace methods are based on projecting onto (some variant of) the Krylov subspace $\mathcal{K}_k(A, r_0) = \text{span}\{r_0, Ar_0, \dots, A^{k-1}r_0\}$, $k = 1, 2, \dots$, where $r_0 = b - Ax_0$; see, e.g., [96]. Obviously, they are nonlinear in A because Krylov subspaces are formed using repeated multiplications with the matrix A . But this form of nonlinearity can be attributed to any iteration process repeatedly applying an iteration matrix. In particular, it can obviously be attributed to the Chebyshev method; see, e.g., [37,38]. A principally different and more substantial nonlinearity with respect to A as well as to r_0 comes from using *projections onto Krylov subspaces*, which is equivalent to enforcing some form of *optimality* of the approximate solution *that requires adaptation to the data* A and r_0 at each iteration step; see, e.g., [67] for a comprehensive treatment. This essential point is frequently overlooked with far reaching consequences. In some cases the nonlinear behavior of a Krylov subspace method is simply, and mathematically incorrectly, identified with widely known linear convergence bounds.⁴ In other cases, the nonlinear behavior is accepted but viewed primarily as an obstacle for the analysis. However, we claim the following:

Main point: The nontrivial nonlinearity is the main mathematical asset as well as the beauty of Krylov subspace methods, since it requires the methods to adapt to the hidden inner structure of the problem to be solved. This can lead to a significant speedup of the convergence in comparison with (linear) iterative methods that do not adapt to the problem.

Because of their nonlinearity, which is rooted in their optimality requiring adaptation to the data, Krylov subspace methods can show their advantages particularly in solving hard practical problems, where their behavior can not be understood using linear upper bounds. Neglecting the nonlinearity hampers further investigation of intricate and open problems, which however is critically needed for advancing the theory as well as the successful practical application of the methods. The main point stated above will therefore reappear in many of the computed examples below.

The paper is organized as follows. In Section 2 we consider the CG method, and in Section 3 the GMRES method. Both sections start with a brief description of the methods (mathematical properties and standard implementations), followed by the computed examples. Section 4 contains concluding remarks. The Appendix contains comments on the modern relevance of early works on Krylov subspace methods, and on the topic of preconditioning.

⁴ For symmetric positive definite matrices this means that CG is considered equal in performance to the Chebyshev method with a priori knowledge of the edges of the spectrum. Sometimes the distinction between CG and the Chebyshev method is indeed reduced only to the fact that CG does not need any such a priori information. But the distinction consists of their principally different optimality properties, which is not linked in the Chebyshev method to an adaptation to the data; see Section 2 below. This is clear from the early papers by Hestenes, Stiefel, and Lanczos (see, in particular, [64,63]) mentioned above, although the description of the Chebyshev method came later.

Notation and conventions Throughout the paper we consider real linear algebraic systems for simplicity of notation. Most results can be easily extended to the complex case. We use N for the matrix size, i.e., $A \in \mathbb{R}^{N \times N}$, and k denotes the iteration number. Usually in our computed examples the right-hand side b is a normalized vector of ones, and the initial approximate solution is $x_0 = 0$. We use the term “mathematical” to refer to cases where computations are performed exactly, i.e., in infinite precision, and the term “computational” to refer to finite precision computations. In some experiments we compare the infinite and finite precision behavior of algorithms. Finite precision computations are performed using the standard double precision arithmetic in MATLAB. Unless otherwise specified, the mathematical (infinite precision) behavior is then simulated using the Advanpix Multiprecision Computing Toolbox for MATLAB.⁵

2. The CG method

The CG method is well defined for any linear algebraic system $Ax = b$ with a symmetric positive definite matrix $A \in \mathbb{R}^{N \times N}$ and right-hand side $b \in \mathbb{R}^N$. If $x_0 \in \mathbb{R}^N$ is an initial approximation, and $d = d(A, r_0)$ is the grade⁶ of the initial residual $r_0 = b - Ax_0$ with respect to A , then at every step $k = 1, 2, \dots, d$ the CG method constructs a uniquely determined approximation

$$x_k \in x_0 + \mathcal{K}_k(A, r_0) \quad \text{such that} \quad r_k \perp \mathcal{K}_k(A, r_0), \quad (1)$$

where $\mathcal{K}_k(A, r_0) := \text{span}\{r_0, Ar_0, \dots, A^{k-1}r_0\}$ is the k th Krylov subspace generated by A and r_0 . Mathematically the method terminates with $x_d = x$.

There are many equivalent formulations of the task that is solved by the CG method. For example, at step k the CG method determines the solution of the simplified Stieltjes moment problem (see [88]) or, equivalently, it determines the k -point Gauss quadrature of the Riemann-Stieltjes integral defined by A and r_0 ; see, e.g., [67, Section 3.5] and [72, Section 5.2] for overviews. These connections were pointed out by Hestenes and Stiefel in [53] and they are important for understanding the mathematical as well as the computational behavior of CG.

There are also many mathematically equivalent algorithms that realize the projection process (1). The most popular variant is the original formulation of Hestenes and Stiefel [53], shown in Algorithm 1. This algorithm recursively updates coupled 2-term recurrences for the approximate solution x_{k+1} and residual r_{k+1} , as well as the auxiliary “search direction” vector p_{k+1} . As it turns out, this variant is also preferable computationally; see, e.g., [91,50].

⁵ <https://www.advanpix.com>.

⁶ The grade of r_0 with respect to A is defined as the maximal dimension of the Krylov space generated by A and r_0 .

Algorithm 1 Conjugate Gradient (2-term recurrence variant).

Require: Symmetric positive definite matrix $A \in \mathbb{R}^{N \times N}$; right-hand side b ; initial approximation x_0 ; convergence tolerance τ ; maximum number of iterations n_{\max} .

- 1: $r_0 = b - Ax_0$
- 2: $p_0 = r_0$
- 3: **for** $k = 0, 1, 2, \dots, n_{\max}$ **do**
- 4: $\alpha_k = (r_k^T r_k) / (p_k^T A p_k)$
- 5: $x_{k+1} = x_k + \alpha_k p_k$
- 6: $r_{k+1} = r_k - \alpha_k A p_k$
- 7: Test for convergence using tolerance τ . If satisfied, then return x_{k+1} and stop.
- 8: $\beta_{k+1} = (r_{k+1}^T r_{k+1}) / (r_k^T r_k)$
- 9: $p_{k+1} = r_{k+1} + \beta_{k+1} p_k$
- 10: **end for**

Let $A = Q\Lambda Q^T$, with $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_N)$ and $0 < \lambda_1 \leq \dots \leq \lambda_N$, be an orthogonal diagonalization of A .⁷ We can represent the initial residual r_0 by its components in the individual eigenvectors of A , stored in the columns of Q , as $r_0 = Q[\eta_1, \dots, \eta_N]^T$. The approximation $x_k \in x_0 + \mathcal{K}_k(A, r_0)$ that is uniquely determined by the orthogonality condition in (1) satisfies the (equivalent) optimality property

$$\|x - x_k\|_A = \min_{p \in P_k(0)} \|p(A)(x - x_0)\|_A = \min_{p \in P_k(0)} \left(\sum_{i=1}^N \eta_i^2 \frac{p(\lambda_i)^2}{\lambda_i} \right)^{1/2}, \tag{2}$$

where $P_k(0)$ denotes the set of polynomials of degree at most k with value 1 at the origin; see, e.g., [67, Section 5.6]. Thus, in every step the CG method solves a certain weighted polynomial approximation problem on the discrete set $\{\lambda_1, \dots, \lambda_N\}$. Moreover, if $\theta_1^{(k)}, \dots, \theta_k^{(k)}$ are the k roots of the polynomial providing the minimum in (2), then we can easily get

$$\|x - x_k\|_A^2 = \sum_{i=1}^N \prod_{\ell=1}^k \left(1 - \frac{\lambda_i}{\theta_\ell^{(k)}} \right)^2 \frac{\eta_i^2}{\lambda_i}, \tag{3}$$

which establishes the relationship of the roots of the minimizing polynomial in (2), called also the Ritz values, with the eigenvalues of the matrix A .

Note that

$$\sum_{i=1}^N \frac{\eta_i^2}{\lambda_i} = r_0^T A^{-1} r_0 = (x - x_0)^T A (x - x_0) = \|x - x_0\|_A^2.$$

⁷ CG is mathematically invariant under orthogonal transformations of the basis in \mathbb{R}^N . In particular, one can study its mathematical behavior using the basis formed by the orthonormalized eigenvectors of A , i.e., using the diagonal matrix Λ instead of A . The results of Greenbaum [43] allow to view finite precision CG computations (apart from a small inaccuracy) as exact CG for a particular larger matrix having all its eigenvalues close to the eigenvalues of A . Combining this with experimental demonstrations in [47], and with further theoretical arguments in [67, Section 5.9.1] and [77], one can study also the computational behavior of CG using the diagonal matrix Λ .

Therefore, maximizing over the values $p(\lambda_i)$ in the minimization problem on the right-hand side of (2) and dividing by $\|x - x_0\|_A$ gives the upper bound

$$\frac{\|x - x_k\|_A}{\|x - x_0\|_A} \leq \min_{p \in P_k(0)} \max_{1 \leq i \leq N} |p(\lambda_i)| \tag{4}$$

It is important to note that the polynomial min-max approximation problem on the right-hand side of (4) only depends on A , but not on r_0 .

If $d(A)$ denotes the degree of the minimal polynomial of A , then $d(A) \geq d$. It was shown by Greenbaum [42], that for any given symmetric positive definite matrix $A \in \mathbb{R}^{N \times N}$ the bound (4) is sharp in the sense that for every step $k \leq d(A)$ there exists an initial residual r_0 so that equality holds. Thus, for the given matrix A the value of the polynomial min-max approximation problem is an attainable *worst-case bound* on the relative A -norm of the error in the CG method at every step $k \leq d(A)$. (The step $k = d(A)$ is trivial.) Moreover, for every $k = 1, \dots, d(A) - 1$, there exist $k + 1$ distinct eigenvalues $\hat{\lambda}_1, \dots, \hat{\lambda}_{k+1}$ of A , such that

$$\min_{p \in P_k(0)} \max_{1 \leq j \leq N} |p(\lambda_j)| = \left(\sum_{i=1}^{k+1} \prod_{\substack{j=1 \\ j \neq i}}^{k+1} \frac{\hat{\lambda}_j}{|\hat{\lambda}_j - \hat{\lambda}_i|} \right)^{-1}. \tag{5}$$

The value of the worst-case bound for CG in step k is thus expressed in terms of a subset of $k + 1$ particular eigenvalues of A . This subset is determined by the min-max polynomial approximation problem on the set $\{\lambda_1, \dots, \lambda_N\}$ for the polynomials of degree k that are normalized at the origin.⁸

Replacing the discrete set $\{\lambda_1, \dots, \lambda_N\}$ by the continuous interval $[\lambda_1, \lambda_N]$ and using Chebyshev polynomials on this interval yields (with a small additional simplification)

$$\min_{p \in P_k(0)} \max_{1 \leq i \leq N} |p(\lambda_i)| \leq 2 \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^k, \quad \kappa(A) = \frac{\lambda_N}{\lambda_1}. \tag{6}$$

This bound represents a substantial simplification. While (5) gives a sharp upper bound on the relative CG error A -norms for the given (fixed) spectrum and any initial residual, the right hand side in (6) gives a nearly sharp upper bound for *arbitrary* eigenvalues $\{\lambda_2, \dots, \lambda_{N-1}\}$ in the interval $[\lambda_1, \lambda_N]$ and any initial residual.

⁸ The whole paper [42] is worth reading. It illustrates the peculiarities of the polynomial approximation problem on a discrete set of points, and it derives the weights for which the normalized k th degree min-max polynomial on a subset $\{\hat{\lambda}_1, \dots, \hat{\lambda}_{k+1}\}$ of the set $\{\lambda_1, \dots, \lambda_N\}$ is equal to the normalized weighted least squares polynomial approximation to zero on $\{\hat{\lambda}_1, \dots, \hat{\lambda}_{k+1}\}$.

Combining (6) with (2) results in the frequently stated convergence bound

$$\frac{\|x - x_k\|_A}{\|x - x_0\|_A} \leq 2 \left(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1} \right)^k. \quad (7)$$

We will sometimes refer to (7) as the $\kappa(A)$ -bound. This bound implies that if the condition number $\kappa(A)$ is small, then a fast reduction of the A -norm of the error in the CG method can be expected. This bound does *not* imply, however, that a large condition number results in slow convergence of CG. In particular, preconditioners that provide smaller condition numbers than others do not necessarily lead to faster convergence. A convincing example in [30] uses standard PDE test problem and contributes towards opening a new line of research combining the PDE operator context with the algebraic matrix context arising from discretization; see further comments on this point below.

Also note that the $\kappa(A)$ -bound for CG is a *linear bound for a nonlinear process*. A comparison with the value of the polynomial min-max approximation problem in (5), which gives the worst-case CG value in step k for the given spectrum of the matrix A , shows that neglecting the eigenvalue distribution of A in the interval $[\lambda_1, \lambda_N]$ can mean a substantial loss of information. Similarly, a comparison with the actual minimization problem (2), which is solved by CG applied to the linear system $Ax = b$ with the initial approximation x_0 , shows that the size of the components η_j of r_0 in the invariant subspaces of A can be important; see also (3).

In the examples that follow, we will frequently make use of a certain class of diagonal matrices which is often used in the literature to illuminate the behavior of CG; see, e.g., [47]. For given $N \geq 3$, $0 < \lambda_1 < \lambda_N$, and $\rho > 0$ we define

$$A = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_{N-1}, \lambda_N) \quad \text{with} \quad \lambda_i = \lambda_1 + \left(\frac{i-1}{N-1} \right) (\lambda_N - \lambda_1) \rho^{N-i}, \quad (8)$$

for $i = 2, \dots, N-1$. The parameter ρ determines the eigenvalue distribution of A . When $\rho = 1$, the eigenvalues are equally spaced between λ_1 and λ_N . As ρ becomes smaller, the eigenvalues accumulate towards λ_1 . As mentioned above, we can use a diagonal matrix mathematically and computationally without any loss of generality.

2.1. Mathematical behavior of CG for different eigenvalue distributions

Main point: The CG optimality property (see (2)) depends on the positions of the individual eigenvalues. Therefore CG adapts without any a priori information not only to the spectral interval, but in a significant (and nonlinear) way also to the distribution of the inner eigenvalues. Acceleration of CG convergence is more pronounced for matrices with outlying eigenvalues, and is different when the outliers are small or large.

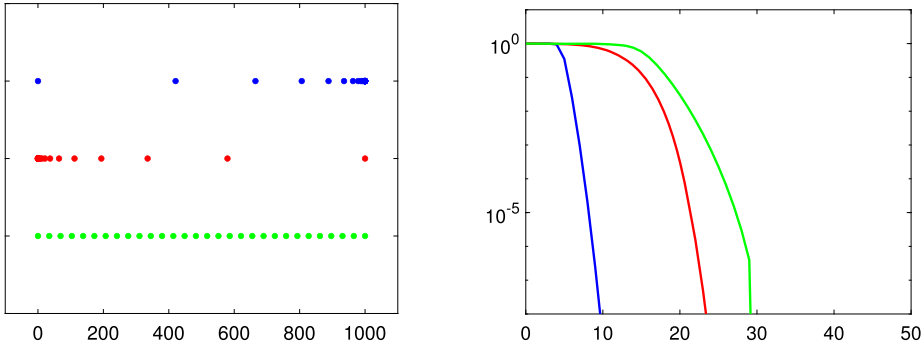


Fig. 1. Left: Three distributions of 30 eigenvalues in $[0.1, 10^3]$. Right: The relative error in the A -norm for exact CG applied to the corresponding linear algebraic systems. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

Setup: We consider the behavior of CG in exact arithmetic for matrices having three different eigenvalue distributions. All matrices are diagonal with $N = 30$, $\lambda_1 = 0.1$, and $\lambda_N = 10^3$. The first matrix is a slight modification of (8), with $\lambda_i = \lambda_N - \frac{(i-1)}{(N-1)}(\lambda_N - \lambda_1)\rho^{N-i}$ for $i = 2, \dots, N - 1$ and $\rho = 0.6$, so that the eigenvalues accumulate on the right side of the spectrum. The second matrix is (8) with $\rho = 0.6$, so that its eigenvalues accumulate to the left side of the spectrum, and the third matrix is (8) with $\rho = 1$, so that its eigenvalues are equally spaced. In all cases we use $b = [1, \dots, 1]^T / \sqrt{N}$, and $x_0 = 0$.

Observations: The eigenvalue distributions are shown in the left part of Fig. 1. The right part of Fig. 1 shows the mathematical behavior of CG. For the matrix with eigenvalues accumulated to the right (blue), CG converges fastest. For the matrix with eigenvalues accumulated to the left (red), CG converges significantly slower. For the matrix with equally spaced eigenvalues (green), CG converges the slowest. In Fig. 2 we show cumulative spectral density (CSD) plots using the stepwise functions with points of increase at Ritz values and the size of the vertical steps equal for each Ritz value (see [70, Appendix C]).⁹

Explanation: As can be seen from (2), for equal components of the initial residual in the invariant subspaces, the A -norm of the error minimizes the sum of the squared values of the CG polynomial divided by the associated eigenvalues. For an accumulation of the eigenvalues to the right, the CG polynomial approximates by its roots (Ritz values) the small outlying eigenvalues within a few iterations, which takes care for the associated part of the sum. For the rest of the eigenvalues that are large and close to each other, the values of the polynomial do not need to be so small, because their squares are

⁹ In order to illustrate the position of Ritz values (that can form tight clusters), we approximate the CSD defined by the spectrum of the matrix A using the CSD associated with the Ritz values. We purposefully do not use an approximation of the CSD defined by the spectrum of A via the Riemann-Stieltjes distribution functions associated with the Gauss quadrature (see, e.g., [67, Section 3.5] and [72, Section 3.2]), since this would not allow to observe forming clusters of Ritz values. This will be important in the next two subsections.

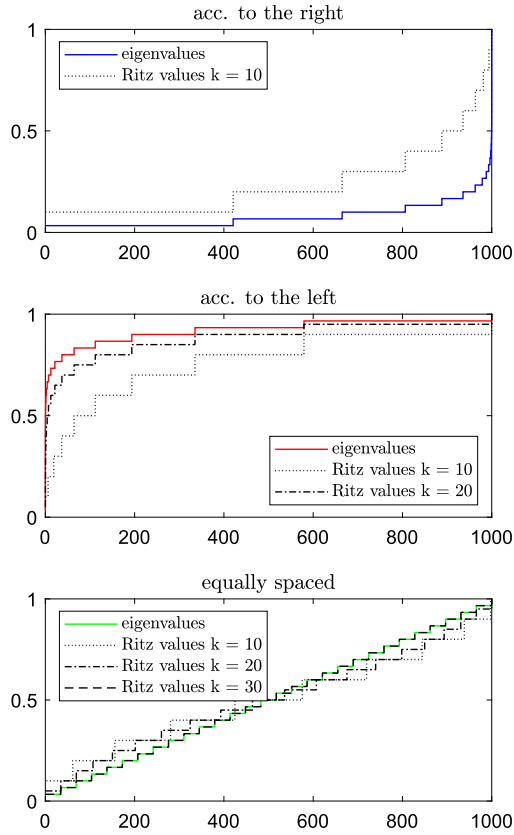


Fig. 2. Cumulative spectral density plots for Fig. 1.

divided by the large eigenvalues. Therefore fast convergence (as if the small outlying eigenvalues were nonexistent) will occur within a few iterations; see [67, Theorem 5.6.9] and the enlightening paper by van der Sluis and van der Vorst [108].¹⁰ Section 5.6.4 of [67], called “Outlying Eigenvalues and Superlinear Convergence”, recalls further closely related results by Lanczos, Rutishauser, Jennings, and others. The arguments above also explain why in this case the convergence rate becomes fast even when the CSD is not yet closely approximated.

For the eigenvalues accumulated to the left, the large outliers are also well approximated by the Ritz values within a few iterations. However, since for the bulk of the small eigenvalues the CG polynomial must place many roots close to the left end of the spectrum in order to make up for the division of its squared values by the small eigenvalues, the acceleration of convergence appears much later. Therefore also the CSD must

¹⁰ Although the paper [108] assumes exact arithmetic and focuses on spectra with small outlying eigenvalues, it also comments on the case of large outlying eigenvalues, and on the difference between the mathematical and computational CG behavior in such cases; see [108, p. 559] and the detailed discussion with further references in [67, pp. 279–280].

be closely approximated in order to significantly decrease the CG error. For the equally spaced eigenvalues the CSD seems visually well approximated. But a closer look reveals rather slow convergence of the Ritz values to the individual eigenvalues, which proceeds from both edges of the spectrum. For more on the convergence of Ritz values in this case see [14,47], [106, Lecture 36], and, in the asymptotic case, [59]. Further interesting points will occur when the same experiments will be performed in finite precision arithmetic; see Section 2.3 below.

2.2. Mathematical behavior of CG for matrices with clustered eigenvalues

Main point: A spectrum localized in ℓ tight clusters does not mean reaching mathematically a good CG approximation to the solution in ℓ steps. The position of clusters is essential.

Setup: We generate three auxiliary diagonal matrices via (8) with different parameters ρ to control the eigenvalue distributions. All matrices have $N = 10$, $\lambda_1 = 0.1$, and $\lambda_N = 10^3$. The first matrix uses $\rho = 0.6$ and a slight modification of (8) so that eigenvalues accumulate to the right. The second matrix uses $\rho = 0.6$ with eigenvalues accumulated to the left. The third matrix uses $\rho = 1.0$, which gives equally spaced eigenvalues. For each auxiliary matrix, we then construct a new matrix of size $N = 100$, which is used in the experiment, by replacing each of the eigenvalues (diagonal entries) by a tight cluster of 10 eigenvalues with spacing 10^{-12} . Thus our matrices have 10 clusters, each with 10 eigenvalues, with cluster diameter $O(10^{-11})$. In each case we use $b = [1, \dots, 1]^T / \sqrt{N}$ and $x_0 = 0$.

Observations: In Fig. 3 we plot the convergence of exact CG for the three problems. Accompanying CSD plots are given in Fig. 4. The matrix has in each case 10 tight clusters of eigenvalues. When the clusters of eigenvalues are accumulated to the right (blue) and when they are the equally spaced (green), the relative error in the A -norm reaches in 10 iterations the level below 10^{-10} . When the clusters are accumulated to the left (red), the relative error in the A -norm makes no progress in 10 iterations. Note that this behavior contradicts the widespread general claims about clustering of eigenvalues and CG convergence, which ignore the positions of clusters.¹¹

Explanation: In the first 10 iterations the CG polynomials for all cases place a single Ritz value in each cluster. For the clusters accumulated to the right as well as equally spaced this is sufficient for approximating the minimal polynomial of the matrix (which is of degree 100) in the sense of (3). For the clusters accumulated to the left, placing one Ritz value in each cluster is not enough to significantly decrease the error, and the minimal polynomial of the matrix is in the same sense not well approximated, despite

¹¹ We use on purpose a very small and simple example. It is easy to find examples for which the same observations are substantially more pronounced.

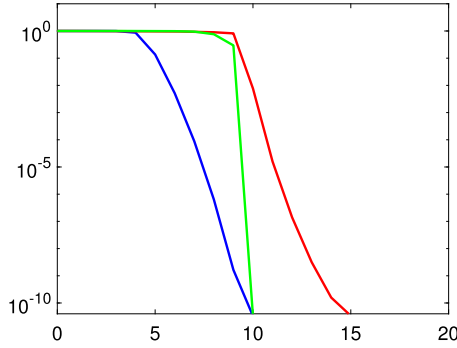


Fig. 3. The relative error in the A -norm for exact CG run on three problems with matrices having different distributions of 10 eigenvalue clusters, where each cluster contains 10 eigenvalues.

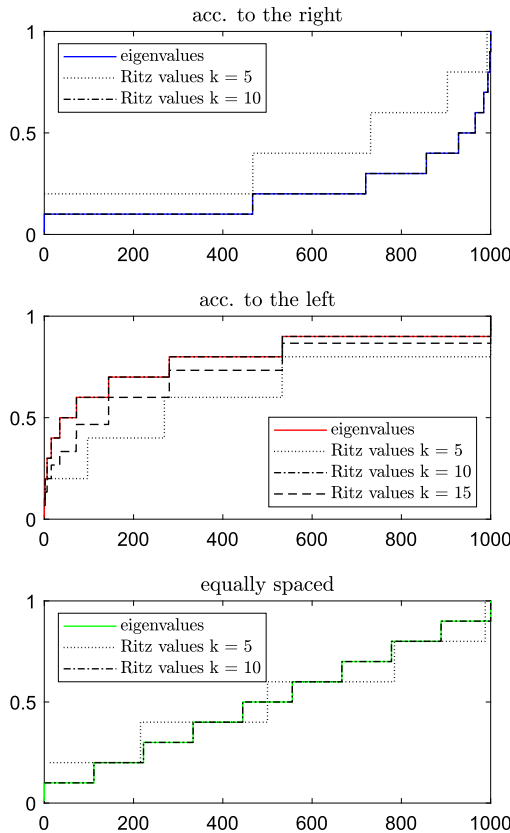


Fig. 4. Cumulative spectral density plots for Fig. 3.

the seemingly analogous position of Ritz values; see the CSD plots, where for $k = 10$ the solid lines and the associated dash-dotted lines graphically coincide. To achieve the desired decrease of the error for the case of clusters accumulated to the left, CG must place additional Ritz values in the rightmost clusters, which delays convergence. At

iteration 15, the five rightmost clusters contain two Ritz values each, and the dashed line representing the CSD for $k = 15$ departs from the CSD representing the matrix spectrum. If the computation proceeds, then this departure would become more and more significant because more and more Ritz values will be placed in the rightmost clusters.

This mechanism has been demonstrated in [47] and it was further thoroughly explained in [67, Section 5.6]. A very detailed account of the relationship between preconditioning and the clustering argument can be found in [6, Section 3(c)]. In order to avoid misunderstandings, we again emphasize that this subsection has dealt with the mathematical behavior of CG. The effects of rounding errors on the convergence are examined next. As we will see, clusters of eigenvalues will then play a specific fundamental role.

2.3. Sensitivity of CG to rounding errors

Main point: Particular eigenvalue distributions, specifically in cases of large outlying eigenvalues, cause CG convergence to be more susceptible to delay caused by finite precision errors. Convergence behavior for finite precision CG can be equated (up to an unimportant difference) with exact CG on a larger problem, whose eigenvalues are replaced by tight clusters.

Setup: We use the same three diagonal matrices and the same right-hand sides as in Section 2.1, but now we run CG in finite (double) precision. We plot the resulting convergence curves on the left in Fig. 5, and the CSDs at certain iterations for each problem in Fig. 6.

Then, for each diagonal matrix, we create a larger matrix by replacing each eigenvalue with a tight cluster of 4 eigenvalues. The spacing between eigenvalues in a cluster is 10^{-13} . We run exact CG for these problems and plot the resulting convergence curves on the right in Fig. 5.

Observations: A comparison of the green and blue curves in Figs. 1 and 5 (left) shows that the convergence of CG for the matrices with eigenvalues accumulated to the right of the spectrum (blue) and for equally distributed eigenvalues (green) is essentially not affected by finite precision errors. On the other hand, for the matrix with eigenvalues accumulated to the left and only a few large outlying eigenvalues, the finite precision CG suffers from a significant delay of convergence (red curves). Moreover, a comparison of the two plots in Fig. 5 shows that the behavior of finite precision CG (left) is remarkably similar to the behavior of exact CG where the eigenvalues are replaced by tight clusters (right).

Explanation: There are two phenomena working against each other here. Whereas large outlying eigenvalues are desirable in exact arithmetic, they cause the problem to be more sensitive to rounding errors, which can result in convergence delay in finite precision computations. This phenomenon was investigated in [100], which was inspired by the earlier discussion by Jennings [55], who related the convergence of CG to a

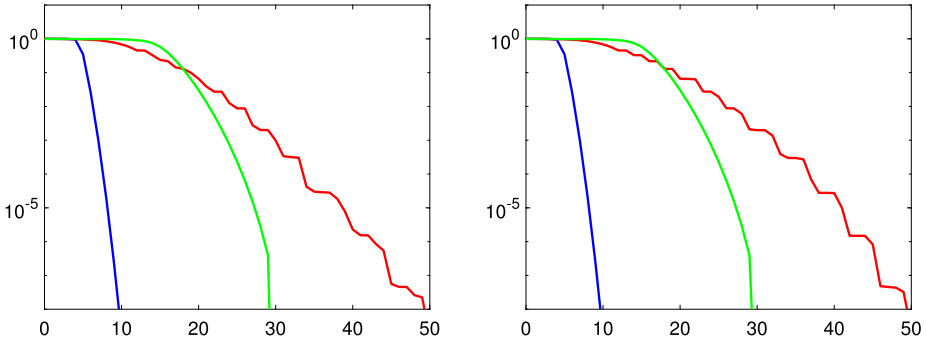


Fig. 5. Left: The relative error in the A -norm for CG in finite precision run on three problems with matrices having different eigenvalues distributions, corresponding to those in Fig. 1. Right: The relative error in the A -norm for exact CG run on three problems with matrices with the same eigenvalue distributions as in the left plot, but with each eigenvalue replaced by a tight cluster of 4 eigenvalues.

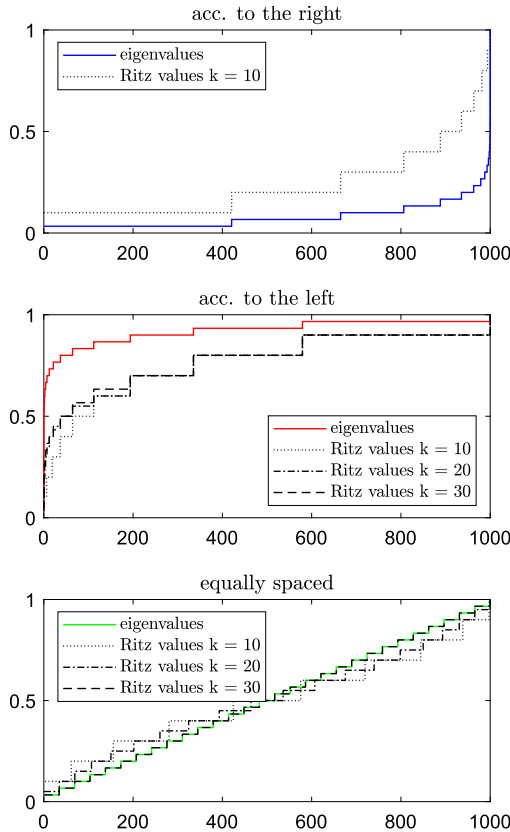


Fig. 6. Cumulative spectral density plots for the left part of Fig. 5, i.e., the Ritz values are determined from the finite precision CG run. Compare with Fig. 2.

polynomial curve fitting problem. It can be nicely viewed via the CSD plots in Fig. 6. While for the eigenvalues accumulated to the right and for equally distributed eigenvalues

there is no observable difference between the exact and finite precision CG computations (compare the top and the bottom plots in Figs. 5 and 2), the CSDs associated with the eigenvalues accumulated to the left are remarkably different. There is no chance that for this problem the CSD determined by the eigenvalues can be closely approximated by the CSD generated by the Ritz values resulting from the finite precision CG computation. The plot in Fig. 5 shows that with increasing iteration number more and more Ritz values have to be placed close to the rightmost outlying eigenvalues.

A theoretical explanation is provided by the work of Greenbaum [43] that was further thoroughly illustrated in [47] and extended by Notay [82]. Additional Ritz values close to the large outlying eigenvalues have to appear in order to eliminate excessively large gradients of the CG approximation polynomials, which would otherwise occur in their neighborhood; see [67, Section 5.9] and [77, Section 5]. For the accumulation of the eigenvalues to the right and for equally distributed eigenvalues the gradient of the CG approximation polynomial near all eigenvalues is sufficiently bounded without a need for placing additional Ritz values in their neighborhoods. This explains the numerical stability of CG for these problems; see also [6, Sections 3(b)(i)]. It is also worth recalling the arguments in Section 2.2 above that deal with the mathematical behavior for problems with tight clusters of eigenvalues.

2.4. Preconditioned CG and the condition number

Main point: The nonlinear adaptivity of CG to the location of the individual eigenvalues indicates that a smaller condition number does not necessarily lead to faster convergence (contrary to widespread misinterpretations of the $\kappa(A)$ -bound in the literature). Therefore it is not recommended to use the minimization of the condition number as the only criterion for the choice of preconditioners. Alternatives to the condition number exist, but they require deep knowledge of CG and of the problem to be solved.

Setup: We perform two experiments. First, we define the matrix A to be of the form (8) with $N = 40$, $\lambda_1 = 10^{-3}$, $\lambda_N = 100$, and $\rho = 0.1$. Thus, the eigenvalues accumulate at the lower end of the spectrum of A , and we have

$$\kappa(A) = 10^5.$$

We consider the diagonal preconditioner P so that $P^{-1}A$ is a diagonal matrix with eigenvalues equally spaced between $\lambda_1 = 10$ and $\lambda_N = 100$, and hence

$$\kappa(P^{-1}A) = 10.$$

We apply exact CG with $x_0 = 0$ to $Ax = b$ and $P^{-1}Ax = P^{-1}b$, where $b = [1, \dots, 1]^T / \sqrt{N}$.

Second, following [78, Section 5.3], we consider the boundary value problem

$$-\nabla \cdot (k(x)\nabla u) = 0 \text{ in } \Omega = (-1, 1) \times (-1, 1), \quad u = u_D \text{ on } \partial\Omega, \tag{9}$$

where the domain Ω is divided into four subdomains $\Omega_1, \Omega_2, \Omega_3, \Omega_4$ corresponding to the four axis quadrants numbered counterclockwise. Let $k(x)$ be piecewise constant on the individual subdomains with $k_1 = k_3 \approx 161.45$ and $k_2 = k_4 = 1$. The Dirichlet boundary conditions are described in [78, Section 5.3]. We use the linear finite element discretization with the standard uniform triangulation and $N = 3969$ degrees of freedom. We apply CG in finite (double) precision with $x_0 = 0$ to the unpreconditioned system, and to preconditioned systems with the algebraic incomplete Cholesky factorization preconditioning (ICHOL) of the matrix A with no fill-in, with ICHOL and the drop-off tolerance 10^{-2} , and with the Laplace operator preconditioning. The corresponding matrices have the following condition numbers:

$$\begin{aligned} \kappa(A) &\approx 6750 && \text{(unpreconditioned),} \\ \kappa(P^{-1}A) &\approx 431 && \text{(ICHOL with no fill-in),} \\ \kappa(P^{-1}A) &\approx 16 && \text{(ICHOL with drop-off tolerance } 10^{-2}\text{),} \\ \kappa(P^{-1}A) &\approx 160 && \text{(Laplace operator preconditioning).} \end{aligned}$$

The same setting was used for the motivating example in [30], where one can find a more detailed description.

Observations: First example: In the left part of Fig. 7 we plot the relative error A -norms for exact CG on $Ax = b$ (blue) and $P^{-1}A = P^{-1}b$ (red). The number of iterative steps for reaching the accuracy level 10^{-8} is for the preconditioned system three times larger than for the unpreconditioned system, although $\kappa(P^{-1}A)$ is four orders of magnitude smaller than $\kappa(A)$. (Note that the preconditioning works well when the desired accuracy level is only on the order 10^{-2} .)

Second example: In the right part of Fig. 7 we plot the relative error A -norms for finite precision CG applied to the unpreconditioned and the three preconditioned systems. A comparison of the results for the unpreconditioned system and the two ICHOL preconditioners seems to confirm the often repeated claim that “reducing the condition number *implies* faster convergence of CG”. However, the fastest convergence occurs for the Laplace preconditioning, although in this case the condition number is an order of magnitude larger than for the ICHOL preconditioner with drop-off tolerance 10^{-2} . It is important to note also the dramatic acceleration of CG with the Laplace preconditioning after the fifth iteration.

Explanation: Both examples show that a smaller condition number does not necessarily imply faster convergence of CG. In particular, in the second example the decrease of the condition number using ICHOL with a larger drop-off tolerance is just a side effect and not the driving force of faster convergence. In this case the improvement is due to

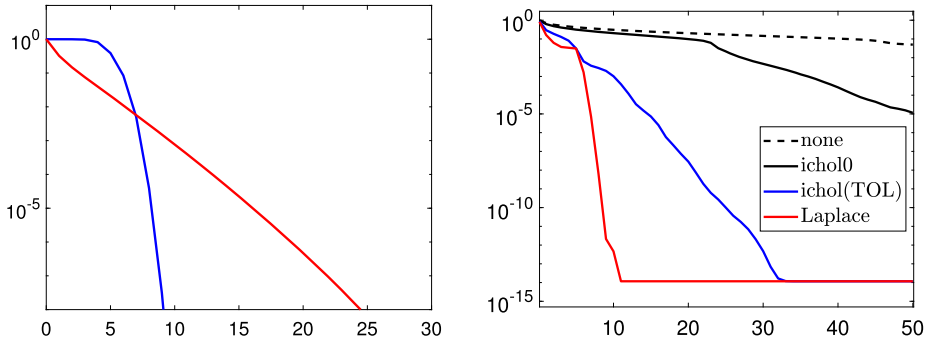


Fig. 7. Left: The relative error in the A -norm for unpreconditioned exact CG (blue) and preconditioned exact CG (red) for a linear algebraic system with A of the form (8). Right: The relative error in the A -norm for finite precision computations with unpreconditioned CG (dashed), ICHOL preconditioning with no fill-in (black), ICHOL preconditioning with drop-off tolerance 10^{-2} (blue), and Laplace preconditioning (red), applied to the discretization of (9).

a better approximation of the true Cholesky factors of A . Clearly, the frequently repeated claim that the goal of preconditioning should be to reduce the condition number of the coefficient matrix (thus reducing the $\kappa(A)$ -bound), and that doing so *guarantees* an improvement of the CG convergence is false.¹² However, one should always take into account the context of the problem to be solved, in particular the desired accuracy of the computed approximation.

A detailed explanation of the second example is out of the scope of this paper, but can be found, including the effects of rounding errors, in [30]. It shows why for the Laplace operator preconditioning the CG convergence *must exactly after five initial iterations* accelerate so rapidly that after a few subsequent iterations an approximate solution is found with an accuracy close to the machine precision level. It also provides an accurate estimate for the speed of the accelerated convergence. For the ICHOL preconditioning it explains why an analogous acceleration can not take place, and also provides an accurate approximation of convergence rate.

As mentioned in the Introduction, it may seem that the design of preconditioners in practical computations using CG has to rely on the condition number because there is no viable alternative. The recent works [30,31,61,89,65,60] show that for self-adjoint second order elliptic PDE boundary value problems with the operator of the form $-\nabla(k(x)\nabla u)$ such an alternative exists. Analysis of the motivating problem in [30] suggests a possible path for further research. The convergence behavior can be anticipated *a priori* based on a low-cost approximation of all eigenvalues of the preconditioned matrix before any preconditioned CG computation starts. We therefore believe that the approach in [30], together with the other given references, question the status quo.

¹² In this context we point out the monograph of Hackbusch [51] which covers, among many other topics, CG in a truly insightful way. Its Section 9.4.3 “Convergence Analysis” addresses also the difference between the Chebyshev method with the bound (7), and CG with the application of the same bound.

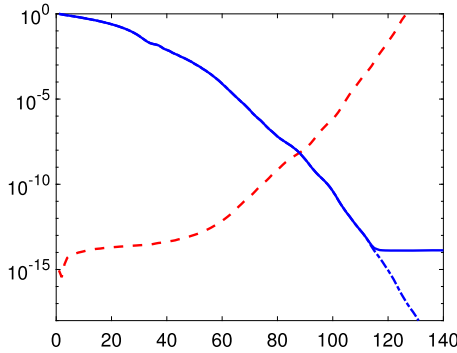


Fig. 8. The relative error in the A -norm for exact CG (dash-dotted blue) as well as for CG in finite precision (solid blue), and the loss of orthogonality among Lanczos basis vectors for CG in finite precision (dashed red), for a 2D Poisson model problem.

The latest paper in this line of development [80] suggests further arguments for combining infinite dimensional operator reasoning with algebraic considerations about solving algebraic systems resulting from discretizations. It also formulates open problems that should be addressed when considering eigenvalues of preconditioned matrices in relation to the spectra of the associated infinite dimensional operators on Hilbert spaces. Section A.2 below indicates that it is promising to investigate more complex approaches as an alternative to decreasing the condition number in preconditioning practical hard problems.

2.5. Simple model problems and the practical performance of CG

Main point: Model problems that are used out of context are not indicative of the behavior of CG in solving practical problems. They can also complicate understanding of important CG features.

Setup: We consider the 2D Poisson problem $-\Delta u = f$ in $\Omega = (0, 1) \times (0, 1)$, $u = 0$ on $\partial\Omega$, and f is constant. This boundary value problem is discretized using the five-point finite differences and the 50×50 grid, giving $N = 2500$ degrees of freedom. The matrix A is simply generated using the MATLAB command `gallery('poisson', 50)`, and we use $b = [1, \dots, 1]^T / \sqrt{N}$. We apply exact CG and CG in finite (double) precision, both with $x_0 = 0$.

Observations: In Fig. 8 we plot the relative A -norm of the error for both exact and finite precision CG. For finite precision CG we plot also the loss of orthogonality among the Lanczos basis vectors, measured by $\|I - V_k^T V_k\|_F$. Its growth seems to mirror the convergence of the relative error in the A -norm, which is almost the same for both exact and finite precision CG until the latter reaches its maximal attainable accuracy.

Explanation: For the given model problem, the eigenvalues are almost uniformly distributed. The loss of orthogonality among the Lanczos basis vectors is gradual. Since

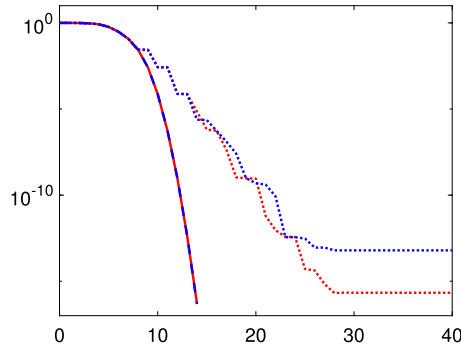


Fig. 9. The relative error in the A -norm for the two variants of CG in exact arithmetic (solid red and dashed blue) and in finite precision (2-term: dotted red; 3-term: dotted blue).

there is no loss of rank in the computed basis until the finite precision CG reaches its final accuracy, there is no delay of the CG convergence; see [67, Section 5.9.4] for a more detailed explanation.¹³ Such behavior can not be extrapolated to CG behavior in solving practical problems.

2.6. Computational behavior of different CG algorithms

Main point: Rounding errors cause convergence delay and affect the attainable accuracy. The magnitude of these effects, and in the case of attainable accuracy, the mechanism, depends on the particular algorithm/implementation of CG.

Setup: We use a diagonal matrix A as defined in (8) with $N = 48$, $\lambda_1 = 0.1$, $\lambda_N = 10^3$, and $\rho = 0.25$. We also use $b = [1, \dots, 1]^T / \sqrt{N}$ and $x_0 = 0$. We test two mathematically equivalent algorithmic variants of CG: the variant of Hestenes and Stiefel [53] which uses three 2-term recurrences (see Algorithm 1), and a different variant which uses two 3-term recurrences.

Observations: The results comparing the relative A -norm of the error for exact and finite (double) precision computations are shown in Fig. 9. In exact arithmetic the A -norm error curves of the 2-term and 3-term variants are obviously identical. In finite precision the convergence is delayed. The delay is slightly worse in the 3-term variant, and the final accuracy level attained by this variant is over two orders of magnitude worse than the level attained by the 2-term variant.

¹³ Paige proved that the loss of orthogonality can occur only in the directions of the converged Ritz vectors; see [83], and [77, pp. 504-508] for an explanation of misinterpretations of this breakthrough result as “convergence implies loss of orthogonality.” Since there is no loss of orthogonality until CG reaches its final accuracy, no Ritz pair can approximate a matrix eigenpair with accuracy proportional to the machine precision level.

Explanation: In the 2-term recurrences the loss of accuracy is caused by a simple accumulation of local rounding errors, but in the 3-term recurrences these local rounding errors can be *substantially amplified*. This behavior is analyzed together with its dependence on the initial residual in [50].

Despite maintaining mathematical equivalence to Algorithm 1, any algorithm that reorders computations or introduces auxiliary quantities can have a different computational behavior. Examples of CG algorithms designed for high-performance parallel environments include *s*-step (communication-avoiding) CG and pipelined CG, both of which are subject to potential amplification of local rounding errors and thus more substantial delays of convergence and worse maximal attainable accuracy than Algorithm 1; see, e.g., [7,8,12].

2.7. Residual versus error, and stopping criteria for CG

Main point: Unlike the *A*-norm (or energy norm) of the error, the residual 2-norm does not have any physical meaning. Moreover, if the matrix condition number is large, then the residual 2-norm is not a reliable indicator of the error. Theoretically justified estimators for the *A*-norm and the 2-norm of the error are available at a negligible computational cost.

Setup: We follow Meurant [74] for constructing two linear algebraic systems of size $N = 20$ such that the trajectories of the residual 2-norm and the *A*-norm of the error in CG are prescribed. For the first, the residual norms $\|r_k\|_2$ oscillate between 1 and 2, and the errors are $\|e_0\|_A = \|x - x_0\|_A = 1$, $\|e_k\|_A = 0.4 \cdot \|e_{k-1}\|_A$ for $k = 1, \dots, N - 1$. For the second, the residual norms are $\|r_0\|_2 = 1$, $\|r_k\|_2 = 0.4 \cdot \|r_{k-1}\|_2$ for $k = 1, \dots, N - 1$, and the errors are $\|e_0\|_A = 1$, $\|e_k\|_A = 0.999 \cdot \|e_{k-1}\|_A$ for $k = 1, \dots, N - 1$.

To construct the systems we set $\nu_k = 1/\|r_k\|_2$, $k = 1, \dots, N - 1$, and $\sigma_k = \|e_k\|_A^2 / (\|r_k\|_2 \|r_0\|_2)$, $k = 0, \dots, N - 1$, and we then set

$$L = \begin{bmatrix} \sigma_0 & & & \\ \sigma_1 & \sigma_1 \nu_1 & & \\ \vdots & \vdots & \ddots & \\ \sigma_{N-1} & \sigma_{N-1} \nu_1 & \cdots & \sigma_{N-1} \nu_{N-1} \end{bmatrix}.$$

We apply exact CG to $A = (L + \widehat{L}^T)^{-1}$ and $b = e_1$, where \widehat{L} is the strictly lower triangular part of L and e_1 is the first column of the identity matrix.

Observations: The relative residual 2-norm and error *A*-norms are shown in Fig. 10. For the first system (left plot) the residual norms stagnate, apart from oscillating between 1 and 2, whereas the error norms decrease linearly. For the second system (right plot) the error norms almost stagnate (they must be strictly decreasing), whereas the residual norms are decreasing relatively quickly.

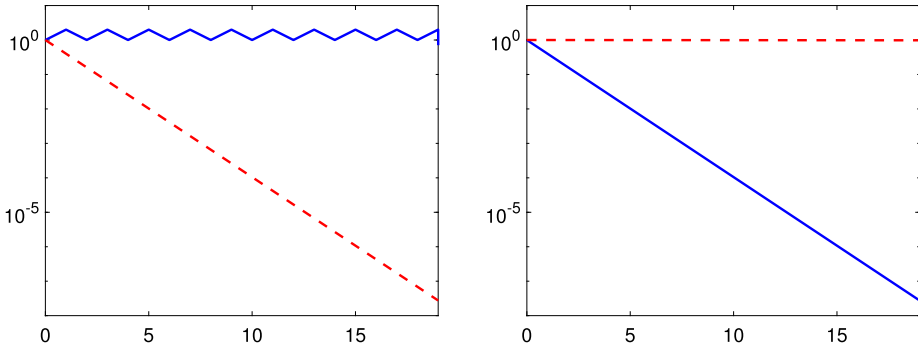


Fig. 10. Comparison of the relative residual 2-norms (solid blue) and the relative A -norms of the error (dashed red) of exact CG for two linear algebraic systems.

Explanation: Hestenes and Stiefel comment in their original paper [53, Section 18] that for *any* prescribed sequence of residual 2-norms, there exists a symmetric positive definite matrix A and right-hand side b such that CG exhibits the prescribed convergence behavior. Thus, in general circumstances, one cannot equate a small residual norm with a small error. The converse also does not hold: a large residual does not imply a large error. Note that because the convergence of CG is linked with the distribution of the eigenvalues of A , it is not possible to also simultaneously prescribe the eigenvalues of A . This is in contrast to the GMRES method; see Example 3.1. If $\kappa(A)$ is small, then the difference between the residual 2-norm and the A -norm (or the 2-norm) of the error is not substantial. This can happen in practical applications. But it can also happen that $\kappa(A)$ is large and there is no practically applicable preconditioning available that makes $\kappa(P^{-1}A)$ small. Therefore the question on what should be used for stopping criteria in CG is relevant.

The relative residual norm is inexpensive to compute, and we already have the recursively computed residual available in each iteration of CG.¹⁴ However, the norm of the residual has no physical meaning, while the A -norm of the error, which is minimized in each step by CG, represents in many applications the (discretization of) the energy norm. In practice, we of course do not know the true solution x , and thus we can not compute the error e_k . This was commented on already by Hestenes and Stiefel, who gave formulas for estimating the error norm; see [53, Section 4]. Since then, much research has focused on developing reliable error norm estimation and associated stopping criteria for CG; see, e.g., [33,35,76,102,103]. Very useful estimates of both the A -norm and the 2-norm of the error are available at a negligible computational cost, and they are theoretically guaranteed to hold also in cases with severe effects of rounding errors. The associated software realizations are simple and freely available, and hence

¹⁴ When CG is used as an inner solver embedded in an outer loop as, for example, in nonlinear optimization, this can be a viable option because the inner-outer heuristic stopping criteria are consistent.

there are strong arguments for considering such error estimates in practical computations.

2.8. The trajectory of finite precision CG computations

Main point: The approximate solutions produced by finite precision CG can be closely mapped to those produced by exact CG via a mapping defined by examining the rank deficiency of the Krylov subspace basis. It seems that the trajectory of the approximate solutions produced by finite precision CG remain in a narrow “tunnel” around those produced by exact CG.

Setup: We generate a diagonal matrix A as defined in (8) with $N = 35$, $\lambda_1 = 0.1$, $\lambda_N = 10^2$, and $\rho = 0.65$. We use $b = [1, \dots, 1]^T / \sqrt{N}$ and $x_0 = 0$. We run exact CG and CG in finite (double) double precision arithmetic. Following [67, Section 5.9.1] and [28], the finite precision CG iterates are then shifted as follows: Consider the sequence

$$\ell(k) = \max \{i \mid \text{rank}_t(\mathcal{K}_i(A, r_0)) = k\}, \quad k = 1, 2, \dots, \tag{10}$$

for the (inexact) Krylov subspace $\mathcal{K}_i(A, r_0)$ computed in double precision. To compute $\text{rank}_t(\mathcal{K}_i(A, r_0))$, we use the built-in MATLAB function `rank` for determining the numerical rank with the threshold $t = 10^{-1}$; compare with [67, Section 5.9.1], in particular, Figure 5.17.¹⁵ For exact CG iterates x_k and finite precision CG iterates $\bar{x}_{\ell(k)}$, we measure the ratios

$$\frac{\|x - \bar{x}_{\ell(k)}\|_A}{\|x - x_k\|_A} \tag{11}$$

and

$$\left| 1 - \frac{\|x - \bar{x}_{\ell(k)}\|_A}{\|x - x_k\|_A} \right|. \tag{12}$$

Observations: In the left part of Fig. 11 we plot the relative error A -norms of the exact CG iterates x_k (solid blue) and the finite precision CG iterates \bar{x}_k (blue circles). In the right part of Fig. 11 the solid blue curve remains the same as in the left part (apart from the change of the horizontal scale), while the blue circles now show the relative error A -norms for the shifted finite precision CG iterates $\bar{x}_{\ell(k)}$. We also plot the ratios (11) (dotted red) and (12) (dashed red). We see that using the mapping (10), the finite precision CG iterates match well with the exact CG iterates. The ratio (11) stays close

¹⁵ As discussed in [67], in this way we count the number of basis vectors that are “strongly linearly independent”. Different value of the threshold will illustrate the same phenomena with a slightly worse quantitative match.

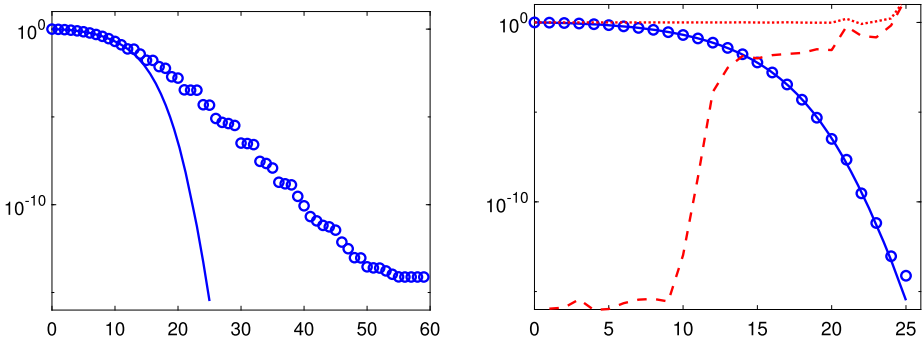


Fig. 11. Left: The relative error in the A -norm for exact CG (solid blue) and finite precision CG (blue circles). Right: The relative error in the A -norm for exact CG (solid blue) and *shifted* finite precision CG (blue circles), the ratio (11) (dotted red), and the ratio (12) (dashed red).

to one throughout the computation. The ratio (12) starts close to machine precision and grows as the iteration proceeds, but stays below one until the finite precision CG gets close to the final accuracy level.

Explanation: The fact that the ratio (11) remains close to one means that the convergence trajectories for exact CG and the shifted finite precision CG are almost identical. This means that finite precision CG computations follow closely the trajectory of exact CG computations, but with progress delayed according to the approximate rank deficiency of the computed (inaccurate) Krylov subspaces. The ratio (12) tells us how far (11) is from one. Rewriting (12), we can say that if

$$\left| \frac{\|x - x_k\|_A - \|x - \bar{x}_{\ell(k)}\|_A}{\|x - x_k\|_A} \right| \ll 1, \tag{13}$$

then the trajectory of finite precision CG iterates is within a narrow “tunnel” around the trajectory of exact CG iterates, where the diameter of the “tunnel” is given by the left-hand side of (13). Although (13) holds for this particular example, it has not been shown to hold for finite precision computations in general. We strongly suspect that such a result holds; initial investigations have been carried out in, e.g., [28] and [29]. The analysis is complicated by the fact that in finite precision computations we can not easily compare the exact Krylov subspaces with the computed subspaces, since the latter ones depart from Krylov subspaces due to the effects of rounding errors.

3. The GMRES method

The GMRES method [97] is well defined for any linear algebraic system $Ax = b$ with a nonsingular matrix $A \in \mathbb{R}^{N \times N}$ and right-hand side $b \in \mathbb{R}^N$. If $d = d(A, r_0)$ is the grade of $r_0 = b - Ax_0$ with respect to A , then at every step $k = 1, 2, \dots, d$ the GMRES method constructs a uniquely determined approximation $x_k \in x_0 + \mathcal{K}_k(A, r_0)$ such that $r_k \perp A\mathcal{K}_k(A, r_0)$, or equivalently

$$\|r_k\|_2 = \min_{z \in x_0 + \mathcal{K}_k(A, r_0)} \|b - Az\|_2 = \min_{p \in P_k(0)} \|p(A)r_0\|_2. \tag{14}$$

GMRES is implemented using the Arnoldi algorithm [2], in practical applications usually in the modified Gram-Schmidt (MGS) variant, which generates an orthonormal basis V_k of $\mathcal{K}_k(A, r_0)$ and a matrix decomposition of the form $AV_k = V_{k+1}H_{k+1,k}$, where $H_{k+1,k} \in \mathbb{R}^{k+1,k}$ is an unreduced upper Hessenberg matrix. Then $x_k = x_0 + V_k t_k$ is determined by

$$t_k = \arg \min_{t \in \mathbb{R}^k} \|b - A(x_0 + V_k t)\|_2 = \arg \min_{t \in \mathbb{R}^k} \|\|r_0\|e_1 - H_{k+1,k}t\|_2. \tag{15}$$

In practical implementations of GMRES, the least squares problem on the right is solved by computing the QR decomposition of the matrix $H_{k+1,k}$. Because of the upper Hessenberg structure, this decomposition can be obtained using Givens rotations, which can be updated in every step. This process also yields an update of the value $\|r_k\|_2$ without explicitly computing x_k . Thus, in practical implementations of GMRES, the least squares problem is solved only when the updated value of the residual norm is below a given tolerance. While all of this is very efficient, the Arnoldi algorithm for a general nonsymmetric matrix requires full recurrences (unlike the short recurrences in CG), and hence the computational cost in terms of work and storage requirements per iteration of GMRES grows significantly. As a consequence, full recurrence GMRES is for large problems typically computationally unfeasible. A common strategy, which already appeared in [97, Algorithm 4], is to restart the algorithm after a certain number of steps. We do not consider restarted GMRES here, but point out that its behavior is not fully understood and sometimes counterintuitive; see, e.g., [19].

A pseudocode implementation of GMRES is shown in Algorithm 2, and more details about the implementation of GMRES can be found, e.g., in [97] and [67, Section 2.5.5]. A detailed analysis why the computation of orthogonal Krylov subspace bases for general nonsymmetric matrices in general requires full instead of short recurrences is given in [67, Chapter 4].

Algorithm 2 GMRES method (pseudocode).

Require: Nonsingular matrix $A \in \mathbb{R}^{N \times N}$, right-hand side b , initial approximation x_0 ; convergence tolerance τ ; maximum number of iterations n_{\max} .

- 1: $r_0 = b - Ax_0$
- 2: **for** $k = 1, 2, \dots, n_{\max}$ **do**
- 3: Compute step k of the Arnoldi algorithm to obtain $AV_k = V_{k+1}H_{k+1,k}$.
- 4: Update the QR factorization of $H_{k+1,k}$ and compute the updated $\|r_k\|_2$.
- 5: If $\|r_k\|_2 \leq \tau$, then compute t_k in (15), return $x_k = x_0 + V_k t_k$, and stop.
- 6: **end for**

If A is diagonalizable, $A = X\Lambda X^{-1}$ with $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_N)$, then

$$\|r_k\|_2 = \min_{p \in P_k(0)} \|p(A)r_0\|_2 \leq \|r_0\|_2 \min_{p \in P_k(0)} \|p(A)\|_2 \tag{16}$$

$$\leq \kappa(X) \|r_0\|_2 \min_{p \in P_k(0)} \max_{1 \leq i \leq N} |p(\lambda_i)|, \tag{17}$$

where $\kappa(X) = \|X\|_2 \|X^{-1}\|_2$. Greenbaum and Trefethen have called the minimization problem on the right hand side of (16) the *ideal GMRES* approximation problem, because taking the upper bound “disentangles the matrix essence of the process from the distracting effects of the initial vector” [49, p. 361]. For certain matrices A and iteration steps k , however, the value of the ideal GMRES approximation problem is much larger than even the worst-case GMRES residual norm, i.e., it can happen that

$$\max_{\substack{v \in \mathbb{R}^N \\ \|v\|_2=1}} \min_{p \in P_k(0)} \|p(A)v\|_2 \ll \min_{p \in P_k(0)} \max_{\substack{v \in \mathbb{R}^N \\ \|v\|_2=1}} \|p(A)v\|_2 = \min_{p \in P_k(0)} \|p(A)\|_2;$$

see [20,104] for the first examples, and [21] for additional examples and a detailed study of the mathematical properties of worst-case GMRES. In such cases every further bound that is derived using ideal GMRES, in particular the one given in (17), will be a significant overestimate of the actual GMRES residual norm.

If A is normal, then we can choose an eigenvector matrix with $\kappa(X) = 1$, and it can be shown that the bound (17) is sharp in the sense that for each step k there exists an initial residual r_0 (depending on A and k) so that equality holds; see the original proofs in [45,56] and [69]. Thus, for a normal matrix A the location of its eigenvalues determines the worst-case behavior of GMRES and, in this worst-case sense, gives an indication of the possible actual behavior.

If A is not normal, then $\kappa(X)$ can be very large, and the value of the upper bound (17) may be far from the actual GMRES residual norm. Moreover, by separating the initial residual from the minimization problem in the upper bound (16), we have lost all information about the relation between the particular given A and r_0 . This relationship between the particular A and b , which is often inherited from the underlying problem to be solved, can be essential for the convergence behavior of GMRES, particularly for nonnormal matrices. We point out that the dependence of the convergence of CG on the initial residual, which is expressed in the exact formula (2), is in general much less pronounced than for GMRES. It is possible to derive closed formulas analogous to (2) also for the GMRES residual norms, even for nondiagonalizable matrices. But because of the generally nonorthogonal eigenvectors (or principal vectors) in the GMRES context, such formulas are more involved and harder to interpret than (2). More details can be found, e.g., in [15] and [75, Sections 5.5–5.6].

The eigenvalues of a general nonsingular matrix A may be anywhere in the complex plane, and hence estimating the value of the polynomial min-max approximation problem in (17) can be very challenging. A quantitative bound can be given in the simple case that the eigenvalues of A are contained in a disk centered at $c \in \mathbb{C}$ and with radius $\rho > 0$, where $\rho < |c|$ is necessary so that zero is outside the disk. Taking the polynomial $(1 - z/c)^k \in P_k(0)$ then shows that

$$\min_{p \in P_k(0)} \max_{1 \leq i \leq N} |p(\lambda_i)| \leq \left(\frac{\rho}{|c|}\right)^k; \tag{18}$$

see, e.g., [96, Section 6.11]. Thus, we can expect that GMRES converges quickly when $\kappa(X)$ is small, and the eigenvalues of A are contained in a small disk that is far away from the origin in the complex plane. A survey of approaches for estimating the value of the min-max approximation problem beyond this special case is given in [67, Sections 5.7.2–5.7.3].

It needs to be stressed that the sharpness of the bound (17) for normal matrices does not imply that GMRES converges faster for normal matrices, or that the (departure from) normality has an easy analyzable effect on the convergence of GMRES. In fact, it can be shown that GMRES may exhibit a complete stagnation even for unitary and hence normal matrices; see [48]. On the other hand, for a nonnormal matrix the location of the eigenvalues alone, and hence the value of the min-max approximation problem in (17), may not give relevant information about convergence behavior of GMRES. If A is not diagonalizable, then its spectral decomposition does not exist, and an analogue of (17) based on the Jordan canonical form is of very limited use. As shown in [46,48], any non-increasing convergence curve is possible for GMRES for a (in general, nonnormal) matrix A having any prescribed set of eigenvalues. The work in [1] gives a parametrization of the set of *all matrices and right-hand-sides* such that GMRES provides a given convergence curve while the matrix has a prescribed spectrum; see [67, Section 5.7.4] for a summary.

3.1. *Any nonincreasing GMRES convergence curve is possible for any eigenvalues*

Main point: Eigenvalues alone are in general not sufficient for describing the GMRES convergence behavior.

Setup: We follow [46, Section 2] for constructing a linear algebraic system $Ax = b$, where $A \in \mathbb{R}^{N \times N}$ has a prescribed set of eigenvalues $\lambda_1, \dots, \lambda_N \in \mathbb{C}$, so that the residual norms of GMRES applied to this system with $x_0 = 0$ are given by a prescribed nonincreasing sequence $f_0 \geq f_1 \geq \dots \geq f_{N-1} > f_N = 0$.

Define $g_j = \sqrt{(f_{j-1})^2 - (f_j)^2}$ for $j = 1, \dots, N$, let $V \in \mathbb{R}^{N \times N}$ be any orthogonal matrix, and let $b = V[g_1, \dots, g_N]^T$. We then construct the polynomial

$$(z - \lambda_1)(z - \lambda_2) \cdots (z - \lambda_N) = z^n - \sum_{j=0}^{N-1} \alpha_j z^j,$$

and its companion matrix

$$A^B = \begin{bmatrix} 0 & \cdots & 0 & \alpha_0 \\ 1 & & 0 & \alpha_1 \\ & \ddots & \vdots & \vdots \\ & & 1 & \alpha_{N-1} \end{bmatrix}.$$

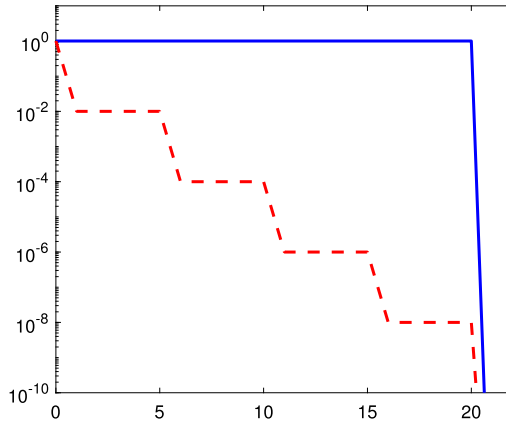


Fig. 12. Residual norms of GMRES applied to the two linear algebraic systems constructed with prescribed eigenvalues and convergence curves (solid blue for $\lambda_1 = \dots = \lambda_N = 1$ and dashed red for $\lambda_j = j$).

With $B = [b, v_1, \dots, v_{N-1}]$, where v_j denotes the j th column of V , we set $A = BA^B B^{-1}$.

We use $N = 21$ and consider two scenarios: In the first we prescribe the eigenvalues $\lambda_1 = \dots = \lambda_N = 1$, and the convergence curve $f_1 = \dots = f_{N-1} = 1 > f_N = 0$. In the second we prescribe the eigenvalues $\lambda_j = j$ for $j = 1, \dots, N$, and a convergence curve that starts at $f_1 = 1$, and then decreases every 4 steps through 10^{-2} , 10^{-4} , and 10^{-6} , to 10^{-8} . In both cases we take $V = I$, and apply MATLAB’s `gmres` function to $Ax = b$ with x_0 . Both matrices in this example are highly ill conditioned and highly nonnormal. Computations with MATLAB’s `cond` and `eig` functions yield

$$\begin{aligned} \kappa(A) &\approx 5.4 \times 10^{11} \quad \text{and} \quad \kappa(X) \approx 7.6 \times 10^{10} && (A \text{ with } \lambda_1 = \dots = \lambda_N = 1), \\ \kappa(A) &\approx 3.5 \times 10^{14} \quad \text{and} \quad \kappa(X) \approx 1.5 \times 10^{21} && (A \text{ with } \lambda_j = j). \end{aligned}$$

Observations: The computed GMRES residual for the two linear algebraic systems is shown by the solid blue and dashed red curves in Fig. 12. As expected, they follow the prescribed convergence curves.

Explanation: The convergence curves in Fig. 12 illustrate the proven theorems from [1, 46,48], and hence should not be surprising from a purely mathematical point of view. On the other hand, the curves illustrate the potentially intriguing behavior of GMRES for nonnormal matrices.

The much slower convergence of GMRES (solid blue curve) occurs for the matrix having a single eigenvalue of algebraic multiplicity N . This clearly demonstrates that for general nonnormal matrices the goal of preconditioners for GMRES needs to be more than just “clustering of the eigenvalues”.¹⁶ We also note that the slower convergence occurs for the somewhat better conditioned matrix.

¹⁶ Recall that this argument requires a thorough reconsideration also for CG; see Sections 2.2 and 2.4.

We would like to mention a point that to our knowledge has not been sufficiently investigated yet: The paper [1] contains a complete parameterization of the set of *all pairs* (A, b) for which GMRES with $x_0 = 0$ generates the prescribed convergence curve, while each of the matrices A has the prescribed fixed eigenvalues. Thus, the parameterization contains not only artificially constructed examples, but also matrices and right hand sides that arise in practical applications. Given the GMRES convergence curve for a particular (practical) linear algebraic system $\hat{A}x = \hat{b}$, we therefore know all pairs (A, b) for which GMRES follows (with $x_0 = 0$) the practically observed behavior, while each of the matrices A has the same eigenvalues as \hat{A} . The set of the pairs (A, b) is worth investigating in particular when \hat{A} is far from normal, but nevertheless its eigenvalues actually describe the GMRES convergence behavior.

3.2. GMRES convergence for normal matrices

Main point: For a normal matrix the eigenvalues, and hence the bound (17), give a reasonable descriptive information of the convergence behavior of GMRES. If (17) is estimated from above by (18), then the resulting linear bound cannot capture a possible acceleration of convergence, which can occur due to adaptation of GMRES to the data.

Setup: We consider normal (in fact, diagonal) matrices with eigenvalues that are rather uniformly distributed in certain disks. In order to construct such matrices we start with real nonsymmetric $(N \times N)$ -matrices with normally distributed random entries that are generated with `randn` in MATLAB. We then compute the eigenvalues of these matrices in MATLAB and form diagonal (and hence normal) matrices D_N with these eigenvalues scaled by $1/\sqrt{N}$. We apply MATLAB's `gmres` function with $x_0 = 0$ to linear algebraic systems with the matrices

$$D_N, \quad 1.2I + D_N, \quad \text{and} \quad 2I + 0.5D_N.$$

The right-hand sides b are normalized random vectors, also generated with `randn` in MATLAB. We use $N = 100$, and we repeat the computation 100 times.

Observations: The left part of Fig. 13 shows the eigenvalues of the 100 matrices D_N and the boundary of the unit disk. The analogous illustrations for the other matrices look similar, with the centers of the disks moved to $c = 1.2$ and $c = 2$, and the radius $\rho = 0.5$ in the last case. We observe three distinctly different types of GMRES convergence behavior in the right part of Fig. 13:

- For the matrices D_N the method makes almost no progress until the very end of the iteration.

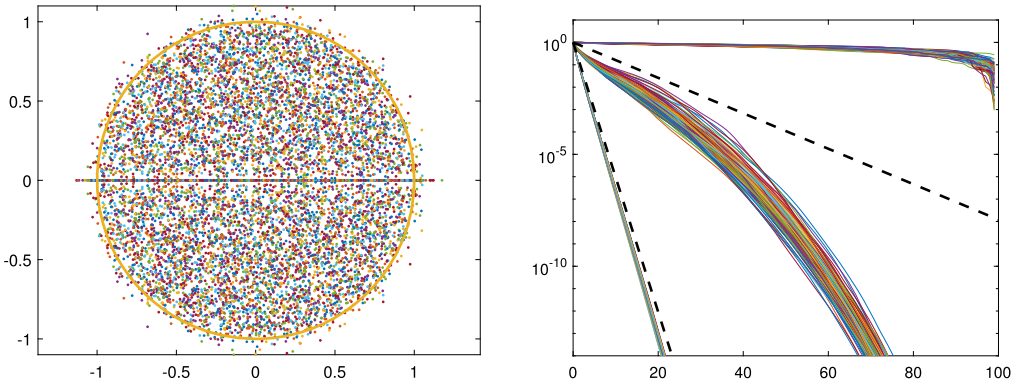


Fig. 13. Left: Eigenvalues of 100 matrices D_N and the boundary of the unit disk. Right: Relative GMRES residual norms for linear algebraic systems with matrices D_N , $1.2I + D_N$, and $2I + 0.5D_N$, and the upper bounds from (18).

- For the matrices $1.2I + D_N$ the method converges faster than for the matrices D_N , the convergence accelerates in later iterations, and the convergence curves exhibit larger variations.
- For the matrices $2I + 0.5D_N$ the method converges linearly and very quickly.

The two dashed lines in the right part of Fig. 13 show the values $(\rho/|c|)^k$ from the upper bound (18) corresponding to the disks with center $c = 1.2$ and radius $\rho = 1$, and with center $c = 2$ and radius $\rho = 0.5$. In the first case the bound captures the rate of convergence in the initial iterations, and in the second case it perfectly matches the convergence curve.

Explanation: According to Girko’s Circular Law, the eigenvalues of the matrices D_N become uniformly distributed in the unit disk for $N \rightarrow \infty$; see the Introduction of [16] for a summary of results in this context. This explains the eigenvalue distributions that are shown in the left part of Fig. 13.

The worst-case behavior of GMRES for normal matrices is completely described by the values of the polynomial min-max approximation problem on the matrix eigenvalues, and these values also give an indication of the actual behavior; see the discussion of the bound (17). Obviously, since the GMRES iteration polynomials are normalized at the origin, the worst-case GMRES (and usually also the actual GMRES) makes almost no progress when the eigenvalues are spread through the unit disk, as for the matrices D_N .

Most of the normal matrices $1.2I + D_N$ have their eigenvalues in the disk centered at $c = 1.2$ and with radius $\rho = 1$. For these matrices the worst-case bound (18) gives a reasonable description of the actual GMRES convergence particularly in the initial phase of the iteration. However, since the bound (18) is linear, its combination with (17) cannot describe the (nonlinear) GMRES acceleration observable in later iterations; cf. Section 2

for similar observations in the context of the $\kappa(A)$ -bound for the CG method. In comparison with CG, understanding the convergence acceleration of GMRES is challenging (and largely open) even for normal matrices. The polynomial min-max approximation problem on a (finite) discrete set in the complex plane is much more difficult to handle than the same problem on a (finite) discrete set on the real line.

Finally, most of the normal matrices $2I + 0.5D_N$ have their eigenvalues located in the disk centered at $c = 2$ and with radius $\rho = 0.5$. For these matrices GMRES converges so fast that no convergence acceleration becomes visible.

3.3. MGS-GMRES is normwise backward stable

Main point: In MGS-GMRES, complete loss of orthogonality of the computed Krylov subspace basis means convergence of the normwise relative backward error to the maximal attainable accuracy, and vice versa. Therefore, the MGS-GMRES method is normwise backward stable.

Setup: The experiment investigates the relation between the loss of orthogonality of the Krylov subspace basis vectors computed in finite precision arithmetic using the modified Gram-Schmidt (MGS) variant of the Arnoldi algorithm (measured by $\|I - V_k^T V_k\|_F$), and the convergence of the normwise relative backward error $\|b - Ax_k\|_2 / (\|b\|_2 + \|A\|_2 \|x_k\|_2)$ in the corresponding MGS-GMRES method.¹⁷

We consider the example matrices `fs1836` and `sherman2` from Matrix Market.¹⁸ The matrix `fs1836` is of size $N = 183$ and has a condition number of approximately 1.0×10^7 . The matrix `sherman2` is of size $N = 1000$ and has a condition number of approximately 2.4×10^7 . Both matrices are diagonalizable, and the condition number of their eigenvector matrices computed by MATLAB are approximately 1.7×10^{11} and 9.6×10^{11} , respectively. We use the right-hand side $b = Ax$, where $x = [1, \dots, 1]^T / \sqrt{N}$. Since MATLAB's `gmres` function is based on the Householder variant of the Arnoldi algorithm, we use in this experiment our own MGS-GMRES implementation, starting with $x_0 = 0$.

Observations: As shown in Fig. 14, throughout the iteration the product

$$\|I - V_k^T V_k\|_F \times \frac{\|b - Ax_k\|_2}{\|b\|_2 + \|A\|_2 \|x_k\|_2}$$

is almost constant, and close to the machine precision (approximately 10^{-16}). The orthogonality of the basis vectors is completely lost only when the normwise backward

¹⁷ For a discussion of the practical relevance of the normwise relative backward error when solving linear algebraic systems we refer to any good textbook on numerical linear algebra. Its use as a stopping criterion for GMRES is discussed in Section 3.6.

¹⁸ <https://math.nist.gov/MatrixMarket/>.

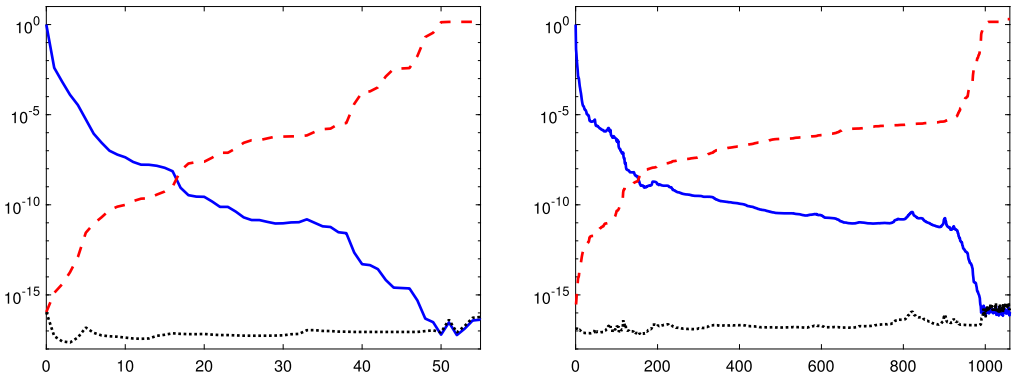


Fig. 14. MGS-GMRES normwise relative backward errors (solid blue) and loss of orthogonality (dashed red), and the product of the two (dotted) for linear algebraic systems with the matrices `fs1836` (left) and `sherman2` (right).

error of the MGS-GMRES iteration has reached its maximal attainable accuracy level. We point out that this is a significant difference to the finite precision behavior of CG and other methods based on short recurrences. Then, apart from very special situations (see Section 2.5), not only a loss of orthogonality but also a loss of rank in the computed subspace may occur, which leads to a delay of convergence; see Section 2.3 and the detailed explanations in, e.g., [67, Section 5.9] and [77].

Explanation: The full explanation of the phenomenon observed in Fig. 14 is based on a detailed analysis of rounding errors that occur in MGS-GMRES; see the papers [13, 87,84] or the summary in [67, Section 5.10]. In this analysis it is shown (under some technical assumptions, e.g., that A is not too close to being singular) that the loss of orthogonality in the Krylov subspace basis V_k computed in finite precision arithmetic using the MGS variant of the Arnoldi algorithm is essentially controlled by the condition number $\kappa([\gamma v_1, AV_k D_k])$, where $\gamma \in \mathbb{R}$ and $D_k \in \mathbb{R}^{k \times k}$ are suitable chosen scalings. Since $r_k = r_0 - AV_k t_k = \|r_0\|v_1 - AV_k t_k$, the conditioning of the matrix $[\gamma v_1, AV_k D_k]$ can be related to the residual norm $\|b - Ax_k\|_2$, and in a second step also to the normwise relative backward error $\|b - Ax_k\|_2 / (\|b\|_2 + \|A\|_2 \|x_k\|_2)$. This yields a rigorous proof of the numerically observed behavior of MGS-GMRES. It is worth noting that working on this challenge led to revisiting the theoretical foundations of (scaled) total least squares problems; see, e.g., [86].

Note that the (more costly) GMRES implementation based on the Householder variant of the Arnoldi algorithm, which is used, e.g., in MATLAB, is also normwise backward stable [13]. In the classical Gram-Schmidt (CGS) variant, however, the orthogonality is lost too quickly to guarantee backward stability of the corresponding CGS-GMRES implementation; see [32] for a rounding error analysis of the CGS orthogonalization algorithm.

3.4. GMRES convergence for approximately computed preconditioning

Main point: In practical computations, preconditioning can not be performed exactly due to rounding errors. More substantially, theoretical preconditioning often has to be intentionally approximated with a rather relaxed accuracy in order to prevent prohibitive cost. Therefore theoretical results, which hold for exact preconditioners, must be used with caution when applied to practical heuristics. Moreover, when using an (approximate) preconditioner, the norm of the true residual of the final computed approximation should be checked.

Setup: We set up a linear algebraic system $\mathcal{A}x = b$, where

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

comes from a discretization of a Navier-Stokes model problem in IFISS 3.6 [17], and $b = [1, \dots, 1]^T / \sqrt{N}$. We run the `navier_testproblem` with the (mostly default) parameters.¹⁹ The matrix $A \in \mathbb{R}^{n \times n}$ is nonsymmetric, and $B \in \mathbb{R}^{m \times n}$ has full rank m . For our chosen model problem parameters we have $n = 578$ and $m = 256$.

We consider the block diagonal preconditioner

$$\mathcal{P} = \begin{bmatrix} A & 0 \\ 0 & S \end{bmatrix},$$

where $S = BA^{-1}B^T$ is the Schur complement, and we are interested in the behavior of GMRES for the preconditioned system $\mathcal{P}^{-1}\mathcal{A}x = \mathcal{P}^{-1}b$. The top block of the preconditioner \mathcal{P} is given by the explicitly known matrix A . In order to simulate exact preconditioning, we compute the bottom block S using MATLAB's backslash operator for the inversion of A . We use the matrix resulting from this computation as the “exact” preconditioner \mathcal{P} . We then apply our own implementation of MGS-GMRES with $x_0 = 0$ to $\mathcal{A}x = b$ and to the exactly preconditioned system $\mathcal{P}^{-1}\mathcal{A}x = \mathcal{P}^{-1}b$, where we compute $\mathcal{P}^{-1}\mathcal{A}$ and $\mathcal{P}^{-1}b$ again using MATLAB's backslash operator.

In order to simulate the effects of intentionally inexact preconditioning, we illustrate how the GMRES convergence behavior changes when instead of explicitly computing $\mathcal{P}^{-1}\mathcal{A}$ and $\mathcal{P}^{-1}b$, we apply inner GMRES iterations for approximating solutions of linear algebraic systems with \mathcal{P} in every step of the outer GMRES iteration (the so-called inner-outer iterations). We stop the inner iterations (starting with the initial vector $x_0 = 0$) when the relative residual norm reaches the respective tolerances 10^{-8} , 10^{-4} , and 10^{-2} . In this way we obtain three GMRES convergence curves for approximately

¹⁹ Cavity; regularized; 16x16 grid; uniform grid; Q1-P0; viscosity: 1/100; hybrid; Picard: 1; Newton: 1; nonlinear tolerance: 1.1*eps; uniform streamlines.

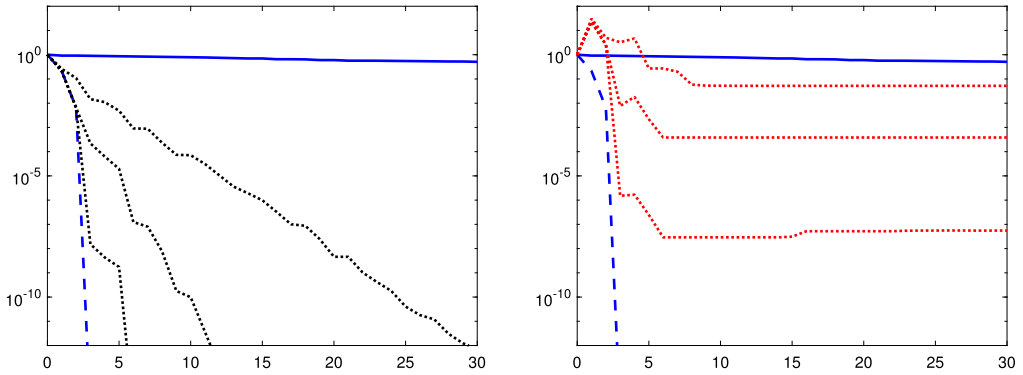


Fig. 15. Both parts show the relative residual norms $\|r_k\|_2/\|r_0\|_2$ of GMRES for the unpreconditioned Navier-Stokes system (solid blue) and for the exactly preconditioned system (dashed blue). The left part also shows the preconditioned relative residual norms $\|\mathcal{P}^{-1}(b - \mathcal{A}x_k)\|_2/\|\mathcal{P}^{-1}r_0\|_2$ of GMRES for the approximately preconditioned systems with inner iteration tolerances 10^{-8} , 10^{-4} , and 10^{-2} (dotted black). Relaxing the tolerance slows down the GMRES convergence. The right part also shows the relative residual norms $\|r_k\|_2/\|r_0\|_2$ of GMRES for the approximately preconditioned systems with inner iteration tolerances 10^{-8} , 10^{-4} , and 10^{-2} (dotted red). Relaxing the tolerance worsens the final accuracy.

preconditioned systems that simulate the decreasing accuracy of performing preconditioning.

We stress that in this example we only illustrate that inexactness in performing preconditioning can cause a substantial departure of the GMRES convergence behavior from the behavior guaranteed by theoretical results which assume that the preconditioning is performed exactly. By no means we aim to suggest practical preconditioning strategies, or to study inner-outer iterations with particular preconditioners. (Note that in practice a Krylov subspace method combined with multigrid preconditioning is often a viable approach.)

Observations: In the left part of Fig. 15 we show the relative residual norms $\|r_k\|_2/\|r_0\|_2$ of GMRES applied to $\mathcal{A}x = b$ (solid blue) and to the exactly preconditioned system $\mathcal{P}^{-1}\mathcal{A}x = \mathcal{P}^{-1}b$ (dashed blue), as well as the *preconditioned relative residual norms* $\|\mathcal{P}^{-1}(b - \mathcal{A}x_k)\|_2/\|\mathcal{P}^{-1}r_0\|_2$ of GMRES applied to the approximately preconditioned systems with the three tolerances for the inner solves (dotted black). We see that for the unpreconditioned system GMRES makes virtually no progress, and that for exact preconditioner it converges in three steps. The speed of convergence measured by the preconditioned relative residual norm slows down when we relax the accuracy of applying the preconditioner.

The solid blue and dashed blue curves in right part of Fig. 15 are the same as in the left part. The other three curves (dotted red) show the actual relative residual norms $\|r_k\|_2/\|r_0\|_2$ when GMRES is applied to the approximately preconditioned systems with the three different tolerances for the inner iterations as above. We observe that these curves are not monotonically decreasing, and that in each case the maximal attainable accuracy is approximately on the accuracy level of the inner iteration.

Explanation: As shown in the instructive and widely cited paper [79], the minimal polynomial of the (nonsingular) preconditioned matrix $\mathcal{P}^{-1}\mathcal{A}$ is given by $(z-1)(z^2-z-1)$, and hence this matrix has the three distinct eigenvalues 1 and $(1 \pm \sqrt{5})/2$. Thus, in exact computations, GMRES applied to the exactly preconditioned system $\mathcal{P}^{-1}\mathcal{A}x = \mathcal{P}^{-1}b$ converges to the exact solution in at most three steps; see the dashed blue lines in Fig. 15. As clearly pointed out in [79], the degree of the minimal polynomial is essential for this property. For mathematical convergence of GMRES to the exact solution in at most k steps, it is not sufficient that A has only k distinct eigenvalues. For this the matrix additionally must be diagonalizable; cf. the example with $\lambda_1 = \dots = \lambda_N = 1$ in Section 3.1.

In practical computations we usually do not form the preconditioner explicitly, nor do we form the preconditioned matrix. Instead, we use inner iterations for the linear algebraic systems with \mathcal{P} . In addition, these inner iterations are usually based only on an approximation of \mathcal{P} , which is obtained, for example, by approximating a Schur complement. Clearly, the exact mathematical properties of the preconditioned matrix no longer hold for the approximate preconditioning, and therefore cannot be used for a rigorous analysis of the GMRES convergence behavior for the approximately preconditioned systems. The inexactness of the preconditioner can negatively affect the convergence behavior, both in terms of the rate of convergence and the maximal attainable accuracy, as illustrated in Fig. 15.

Many publications, in particular in the context of linear algebraic systems in saddle point form, give bounds on the eigenvalues of inexactly preconditioned matrices, with the goal to show that under inexact preconditioning the few eigenvalues with large multiplicity are replaced by a few clusters of eigenvalues. Related to the clustering heuristics, it is worth considering the next two points. First, for highly nonnormal matrices even small perturbations can make these clusters very large. Second, in order to be meaningful for understanding the GMRES convergence behavior, such a perturbation argument for the eigenvalues needs to be complemented by additional arguments, since there is no guarantee that an approximately preconditioned matrix remains diagonalizable, and the convergence behavior for nonnormal matrices does not depend on the eigenvalues only; see Section 3.1. In particular cases, where the theoretical results on the distribution of eigenvalues indeed explain the results of practical computations, one should try to identify the particular properties (in the words of Lanczos “the inner nature”) of the problem that makes this possible, starting with the mathematical model of the computationally investigated phenomenon. (See also the experiment in the next Section 3.5.)

It is important to note that when applied to a preconditioned system, whether the preconditioner is computed exactly or not, GMRES minimizes the Euclidean norm of the preconditioned residual norm in every step. And since $\|\mathcal{P}^{-1}(b - \mathcal{A}x_k)\|_2 = \|b - \mathcal{A}x_k\|_{(\mathcal{P}\mathcal{P}^*)^{-1}}$, this can be interpreted as the minimization of the residual in a norm that depends on the preconditioner. As pointed out in [18, p. 193] in the context of the MINRES method, “one must be careful not to select a preconditioner

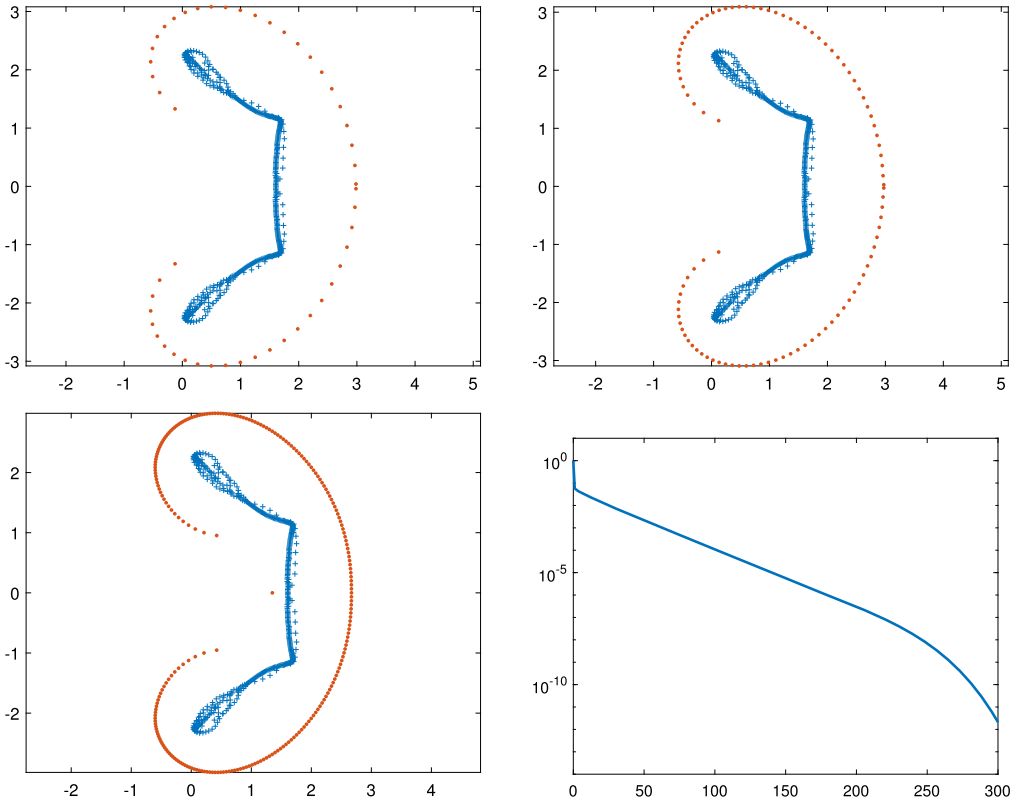


Fig. 16. Eigenvalues of the 500×500 Grcar matrix A computed by MATLAB (blue pluses) and the harmonic Ritz values (red dots) at GMRES steps 50, 100 and 200 (top left, top right, and bottom left), and relative GMRES residual norms for $Ax = b$ with $b = [1, \dots, 1]^T / \sqrt{N}$ and $x_0 = 0$ (bottom right).

following approximately 250 steps. In the other three parts of Fig. 16 the (blue) pluses show the (approximate) eigenvalues of A computed by MATLAB’s `eig` function. (Note that because of the severe ill-conditioning of the eigenvalue problem, this computation is, for the eigenvalues with large imaginary parts, affected by rounding errors. A thorough discussion of the spectrum of the Grcar matrix and of its approximation can be found in [107].) The (red) dots show the roots of the GMRES polynomials, also called the harmonic Ritz values (see, e.g., [67, Section 5.7.1] for mathematical characterizations), at iterations 50, 100, and 200 (top left, top right, and bottom left, respectively). During the iteration, these roots fill up more and more of the same curve that “surrounds” the eigenvalues of A , but overall they fail to move any closer towards the eigenvalues, and hence towards the roots of the minimal polynomial of A .

Explanation: In the CG method, the polynomial of degree k providing the minimum in (2) indeed approximates the minimal polynomial of the matrix A (assuming that all η_i are nonzero) in the sense of solving the simplified Stieltjes moment problem (see [88]) or, equivalently, in the sense of determining the nodes of the associated k -point Gauss quadrature; see, e.g., [67, Section 3.5] and [72, Section 5.2]. As demonstrated in the

examples in Section 2, this does not mean that there exists a *simple revealing relationship* between the eigenvalues of A and the Ritz values $\theta_\ell^{(k)}$ in (3), not even for tightly clustered eigenvalues. Of course, there exist many beautiful properties due to the underlying orthogonal polynomials, and many results about convergence of Ritz values to the eigenvalues. But this is not the same as defining a meaningful measure in relation to approximation of the minimal polynomial.

For GMRES the situation is even more complicated, since analogues of (2) and (3) are not available for the GMRES polynomial. There exist generalizations of the Gauss quadrature through the matrix formulation of the Vorobyev moment problem (see, e.g., [67, Section 3.7.4] and [101]) which apply to the Arnoldi algorithm for approximating eigenvalues and to the closely related FOM method for solving linear systems; see, e.g., [96, Section 6.4]. In this sense, and *only in this sense*, the GMRES polynomial at step k approximates the minimal polynomial of A . However, there is no apparent way how these very involved relationships can give meaningful insights into the location of the k harmonic Ritz values in relation to the roots of the minimal polynomial of A . And indeed, the harmonic Ritz values may remain far from the eigenvalues of A throughout the iteration, although the GMRES residual norms decrease reasonably quickly. This important point is illustrated in Fig. 16.

In many publications it is claimed that the main idea behind Krylov subspace methods is to approximate the minimal polynomial of A . The arguments given above explain why this interpretation is misleading.²⁰ In the context of solving large linear algebraic systems, it can lead to further misconceptions for two main reasons:

Reason 1: The number of iterations performed in practice is typically many orders of magnitude smaller than the degree of the minimal polynomial in question. This remains true also in cases where an “ideal” preconditioner guarantees that, in theory, the degree of the minimal polynomial of the exactly preconditioned matrix is very small. As discussed in Section 3.4, in practice we do not precondition exactly, which results in replacing the few eigenvalues (with large multiplicities) by a few clusters of eigenvalues. The argument continues that we can utilize these clusters for approximating the minimal polynomial of the inexactly preconditioned matrix. However, even if such a matrix is diagonalizable (which is in general not obvious), its minimal polynomial has a very large degree (usually equal to the size of the problem). Approximating clusters of eigenvalues by single roots of the iteration polynomial does not work, apart from particular cases, regardless of how tight the clusters are. This is explained in Section 2.2 for a much easier case. It may work under some specific circumstances and restrictions that have to be clearly stated whenever the argument is used.

Reason 2: The mathematical term “approximation” should be used only with a precise description of the measure that is used for evaluating the accuracy of the approximation.

²⁰ The interpretation may be motivated by the widespread use of Krylov subspaces for solving eigenvalue problems with A , and it somewhat resonates with the original paper of Krylov published in 1931 [58], which deals with computing eigenvalues.

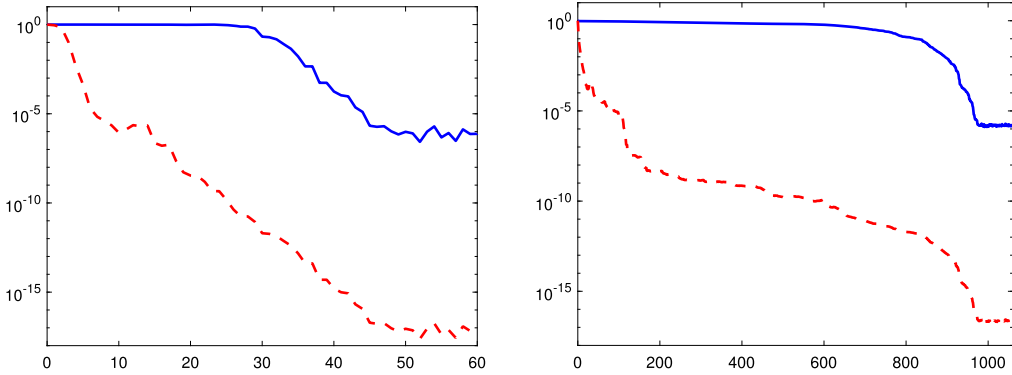


Fig. 17. Residual 2-norms (solid blue) and normwise relative backward errors (dashed red) of GMRES for linear algebraic systems with `fs1836` (left) and `sherman2` (right).

In the context of Krylov subspace methods the flaw is not in using the term “approximation” in relation to the minimal polynomial of the system matrix. The flaw is either in not specifying any measure at all, or in a vague association of such a measure with the locations of the roots of the iteration polynomials.

3.6. Residual versus normwise backward error, and stopping criteria for GMRES

Main point: The residual 2-norm, which is minimized by GMRES, is commonly used as a stopping criterion for GMRES. The normwise relative backward error represents an important and practically relevant alternative.

Setup: We consider linear algebraic systems with the matrices `fs1836` and `sherman2` from Matrix Market (see Section 3.3), and right-hand sides $b = [1, \dots, 1]^T / \sqrt{N}$. We apply our own MGS-GMRES implementation to $Ax = b$, starting with $x_0 = 0$, and compute the relative residual norms $\|r_k\|_2 / \|b\|_2$ and the normwise relative backward errors $\|r_k\|_2 / (\|b\|_2 + \|A\|_2 \|x_k\|_2)$. For the computation of $\|A\|_2$ we use MATLAB’s `norm` function.

Observations: We can see in Fig. 17 that the behavior of the relative residual norm and the normwise relative backward error may be very different. In both cases the relative residual norm initially stagnates and eventually reaches a level of approximately 10^{-6} . The normwise relative backward error, on the other hand, decreases quickly from the start of the iteration and reaches the machine precision level. (Note that the backward error curves for `fs1836` and `sherman2` look a bit different than in Fig. 14, since here we use different right-hand sides.)

Explanation: Given any approximation x_k of the exact solution x , it is useful to ask which linear algebraic system is solved exactly, i.e., for which ΔA and Δb we have

$$(A + \Delta A)x_k = b + \Delta b;$$

see Wilkinson [113,114]. In many applications both A and b result from approximations, measurements or discretizations. From an application point of view it is therefore often sufficient to know that x_k solves a nearby system, i.e., a system where the perturbations ΔA and Δb are small. The perturbation restricted to the right-hand side gives

$$Ax_k = b + \Delta b, \quad \text{where} \quad \Delta b = -r_k = Ax_k - b.$$

Thus, if $\|r_k\|_2$ is small, then x_k solves a system with the same matrix and a slightly perturbed right-hand side. The relative size of the perturbation is $\|r_k\|_2/\|b\|_2$. For $x_0 = 0$ this is the relative residual norm, which commonly is used as a stopping criterion for GMRES. For $x_0 \neq 0$, however, the relative residual norm $\|r_k\|_2/\|r_0\|_2$ does not have a backward error interpretation, and when $\|r_0\|_2 \gg \|b\|_2$ its use as a measure of convergence requires caution; see Section 3.8 below.

As shown by Rigal and Gaches [93], the normwise relative backward error (plotted in Fig. 17) gives the size of the smallest possible perturbations ΔA_{\min} and Δb_{\min} for which x_k solves the perturbed system exactly. More precisely, defining

$$\beta(x_k) := \min \{ \beta : (A + \Delta A)x_k = b + \Delta b, \|\Delta A\|_2 \leq \beta\|A\|_2, \|\Delta b\|_2 \leq \beta\|b\|_2 \}$$

we have

$$\beta(x_k) = \frac{\|r_k\|_2}{\|b\|_2 + \|A\|_2\|x_k\|_2} = \frac{\|\Delta A_{\min}\|_2}{\|A\|_2} = \frac{\|\Delta b_{\min}\|_2}{\|b\|_2}.$$

Consequently, the backward error analysis yields a valid argument for using the normwise relative backward error instead of the relative residual norm as a stopping criterion for GMRES.

In our example, the relative residual norms in Fig. 17 eventually stagnate close to 10^{-6} . This shows that in both cases x_k eventually solves a system with the matrix A and a rather large perturbation of the right-hand side. But since the normwise relative backward errors decrease to the level of the machine precision, we know that x_k eventually solves a system with very small perturbations of A and b .

3.7. GMRES convergence for different right-hand sides

Main point: The convergence behavior of GMRES can depend strongly on the right-hand side, and hence convergence analysis based only on the matrix may not be descriptive.

Setup: The first example is a variation of the *Frank matrix*, which is a test matrix of upper Hessenberg form generated by `gallery('frank', N, N)` in MATLAB. We “flip” this matrix and consider

$$F_N = \begin{bmatrix} N & N-1 & N-2 & \cdots & 1 \\ N-1 & N-1 & N-2 & \cdots & 1 \\ & N-2 & N-2 & \cdots & 1 \\ & & \ddots & \ddots & \vdots \\ & & & & 1 & 1 \end{bmatrix} \in \mathbb{R}^{N \times N}.$$

We use $N = 16$, which yields a highly nonnormal matrix. Computations with MATLAB’s `eig` and `cond` functions yield an eigenvector matrix X with $\kappa(X) \approx 1.1 \times 10^{13}$. We apply our implementation of MGS-GMRES with $x_0 = 0$ to the systems $F_N x = b^{(j)}$, where $b^{(1)} = [1, \dots, 1]^T / \sqrt{N}$, and $b^{(2)}$ is a normalized random vector with normally distributed entries, generated using `randn` in MATLAB.

The second example is a discretized convection-diffusion problem that was studied in [66]; see also [67, Section 5.7.5]. Here the SUPG discretization with stabilization parameter δ of the problem

$$-\nu(u_{xx} + u_{yy}) + u_y = 0 \text{ in } \Omega = (0, 1) \times (0, 1), \quad u = g \text{ on } \partial\Omega,$$

leads to linear algebraic systems $Ax = b$ with $A = A(h, \delta, \nu)$ and $b = b(h, \delta, g)$. We use the discretization size $h = 1/25$ (leading to $A \in \mathbb{R}^{N \times N}$ with $N = h^{-2} = 625$) and fixed parameters $\nu = 0.01$, $\delta = 0.3$, but 25 different boundary conditions g . These boundary conditions set $g = 0$ everywhere on $\partial\Omega$ except for a certain part of the right side of $\partial\Omega$; see [66, Example 2.2] for details. The essential point is that we have only one matrix A , but 25 different right-hand sides $b^{(1)}, \dots, b^{(25)}$. The matrix A is highly nonnormal (here $\kappa(X) \approx 2.5 \times 10^{17}$), and we again apply our implementation of MGS-GMRES with $x_0 = 0$.

Observations: The relative GMRES residual norms for the two different matrices and the corresponding different right-hand sides are shown in Fig. 18. For the flipped Frank matrix the solid blue curve corresponds to $b^{(1)}$, and dashed red curve to $b^{(2)}$. Apparently, GMRES converges much faster for $b^{(1)}$ than for $b^{(2)}$.

In the convection-diffusion problem we have for each $j = 1, \dots, 25$ a right-hand side $b^{(j)}$ for which GMRES has an initial phase of slow convergence (almost stagnation) for exactly $j - 1$ steps. After the initial phase, the GMRES convergence speed is almost the same for all right-hand sides.

Explanation: The matrices used in this example are highly nonnormal, and the convergence bound (17) is therefore of little use. Careful analyses of the relationship between A and b on a case-by-case basis are required to understand the GMRES convergence. For the SUPG discretized convection-diffusion problem such an analysis reveals that the length of the initial stagnation phase of GMRES for different boundary conditions depends on how many steps it takes to propagate the boundary information across the discretized domain by repeated multiplication with the matrix A ; see [66]. That analysis does not explain, however, how the convergence after the initial phase of stagnation depends on the parameters of the problem.

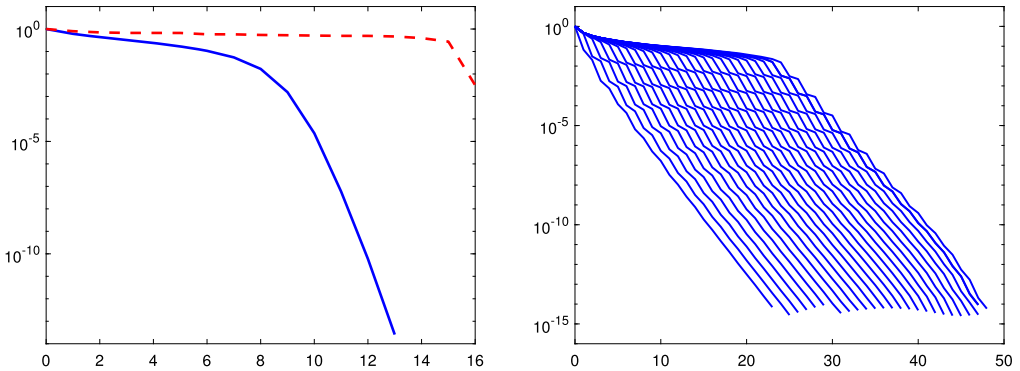


Fig. 18. Relative residual norms of GMRES for linear algebraic systems with the flipped Frank matrix (left) and two different right-hand sides, and a matrix from the SUPG discretization convection-diffusion model problem and 25 different right-hand sides (right).

3.8. GMRES convergence for nonzero initial approximate solutions

Main point: Using a nonzero x_0 and measuring GMRES convergence with the relative residual norms can lead to an illusion of fast convergence. The relative residual norms can decrease rapidly due to dominant information created in r_0 by matrix-vector multiplication. Such information may not be present in the right-hand side, therefore the error can at the same time remain large, which can also lead to a premature stopping of the computation.

Setup: We consider a linear algebraic system $Ax = b$, where $A \in \mathbb{R}^{240 \times 240}$ is the matrix `steam1` from Matrix Market and $b = [1, \dots, 1]^T / \sqrt{N}$. The matrix A is nonsymmetric with $\kappa(A) \approx 2.8 \times 10^7$. We compute $x = A^{-1}b$ using MATLAB’s backslash operator. We apply our implementation of MGS-GMRES to $Ax = b$ with $x_0 = 0$, and with a normalized random x_0 generated using `randn` in MATLAB.

Observations: The relative residual norms $\|r_k\|_2 / \|r_0\|_2$ and the relative error norms $\|x - x_k\|_2 / \|x - x_0\|_2$ of GMRES are shown in the left and right plots of Fig. 19, respectively. The solid black curves correspond to $x_0 = 0$, and the dashed red curves to the normalized random x_0 .

The left plot shows that relative residual norms for the normalized random x_0 decrease quickly, and reach approximately the machine precision level after about 200 iterations. For $x_0 = 0$ the relative residual norms decrease much slower, and eventually reach a level of approximately 10^{-10} . On the other hand, the right plot shows that the relative error norms for both initial vectors are almost the same throughout the iteration, and approximately the same level of accuracy is reached in the end.

Explanation: When the matrix A is ill conditioned, a nonzero x_0 may lead to $\|r_0\|_2 = \|b - Ax_0\|_2 \gg \|b\|_2$. (In our example we have $\|r_0\|_2 \approx 4.5 \times 10^6$ for the normalized random x_0 .) The vector r_0 then contains an *artificially created bias* towards the dominant

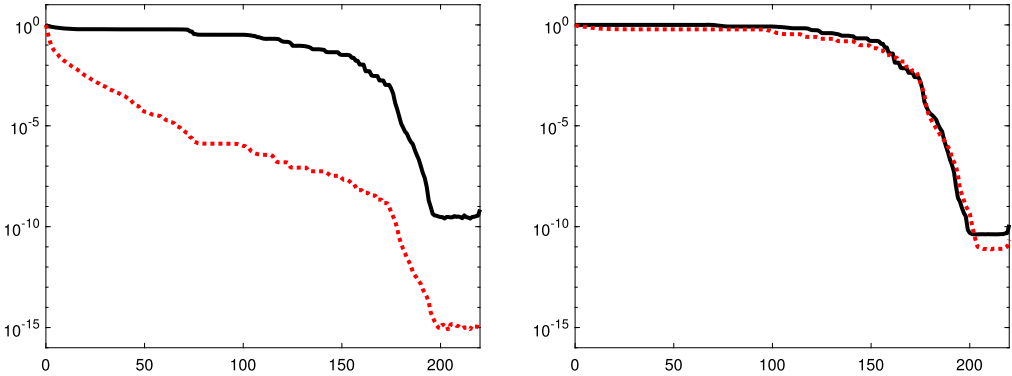


Fig. 19. Relative residual norms (left) and relative error norms (right) of GMRES applied to a linear algebraic system with the matrix `steam1`, starting with $x_0 = 0$ (solid black) and a normalized random x_0 (dotted red).

information in the matrix A (such as large eigenvalues or singular values) that may not be related to the solution $x = A^{-1}b$. Elimination of this bias by GMRES can lead to a fast reduction of the residual norms particularly in the initial phase of the iteration. This creates an illusion of fast convergence (measured by the residual norms), and can lead to a premature stopping although x_k is actually not close to x . Similar examples with different matrices can be seen in [67, Figure 5.13] and [87].

In order to avoid an illusion of fast convergence and a premature stopping, while still using a nonzero x_0 , one can use the rescaling $\zeta_{\min}x_0$, where $\zeta_{\min} = (b^T Ax_0) / \|Ax_0\|_2^2$ solves the approximation problem $\min_{\zeta} \|b - \zeta Ax_0\|_2$; see [67, p. 318]. In our example $|\zeta_{\min}| \approx 2.5 \times 10^{-8}$ holds for the normalized random x_0 , and the residual and error norm curves of GMRES started with $\zeta_{\min}x_0$ are indistinguishable from the solid black curves in Fig. 19.

4. Concluding remarks

As an algorithmic idea, Krylov subspace methods are in their seventies, and their mathematical roots are closely related to much older objects like continued fractions, moments, and quadrature. It might seem that these methods are fully understood, but many questions about their behavior still remain open. Focusing on the CG and the GMRES methods, we have presented an approach towards understanding Krylov subspace methods through a sequence of easily reproducible examples. In some cases this has led to clarification of common misunderstandings about the methods or to the formulation of open problems.

As argued in this paper, we believe that nonlinearity is the main mathematical asset as well as the beauty of Krylov subspace methods. The core misunderstanding in this context is the confusion of the nonlinear and data-adaptive behavior of the methods with linear convergence bounds. In general, the linearization of nonlinear phenomena is a highly useful technique that is applied throughout mathematics. However, it is

important to realize that linearization can capture the behavior of strongly nonlinear phenomena only locally. Asymptotic convergence factors are based on linearization at infinity. When solving finite linear algebraic systems and methods with a mathematical finite termination property, this should be used with caution. Formally we do not have any limit at infinity, and practically we are interested in a few initial iterations (in comparison to the large number of unknowns). When solving linear operator equations in infinite dimensional Hilbert spaces, the convergence rate can become faster than any linear rate as the iterations proceed.

This paper illustrates that the behavior of Krylov subspace methods for some model problems can not be extrapolated to their behavior for practical problems. The condition number bounds, or arguments using clustered eigenvalues, can be useful under specific assumptions and with accepting the associated limitations. For a nice example, which was not discussed in this paper, we refer to the very insightful discussion of spectral equivalence of operators and operator preconditioning by Faber, Manteuffel, and Parter [22]. The goal in this area is to construct preconditioners that guarantee condition number bounds independent of the discretization (and the PDE problem) parameters. This is certainly desirable and useful, but it might not be the final step in guaranteeing fast convergence of Krylov subspace methods. As they point out,

“For a fixed [discretization parameter], using a preconditioning strategy based upon an equivalent operator may not be superior to classical methods [...]. Equivalence alone is not sufficient for a good preconditioning strategy. One must also choose an equivalent operator for which the bound [guaranteeing fast convergence] is small. The above observations indicate that a more precise measure of the ‘closeness’ of two operators is required to evaluate preconditioning strategies.”

Collaboration of researchers working on analysis of Krylov subspace methods, researchers in other fields (such as numerical PDEs), and practitioners who use the methods and are aware of the wider context, can lead to new ideas and paths of research. The approach started in the remarkable paper by Nielsen, Tveito and Hackbusch [81], and continued later in [30] and other works quoted in Section 2.3 can serve as an example.

The class of Krylov subspace methods is nowadays frequently seen as a computational toolbox whose mathematical investigation is more or less finished. The remaining open problems, if known to a broader audience at all, are being accepted as difficult and not worth investigating. We argue that Krylov subspace methods should still be seen as mathematical objects worth studying. Any progress in their understanding, even of their mathematical fundamentals, will bring us a step further in exploiting their full nonlinear computational potential.

Declaration of competing interest

There is no competing interest.

Data availability

Data will be made available on request.

Acknowledgements

We thank Tyler Chen, Justus Ramme, Daniel Szyld, and Jan Zur for helpful suggestions, and Jan Papež for help with Fig. 7.

Appendix A

A.1. Modern relevance of early works about Krylov subspace methods

This paper is not focused on the history Krylov subspace methods. For that purpose we refer to the remarkable commented collection of works between 1948 and 1976 in the survey by Golub and O’Leary [34], and, e.g., to the historical notes in [67]. We believe, however, that a thorough knowledge of the early works on Krylov subspace methods is very useful for understanding the strengths and weaknesses in the present use of these methods, and for obtaining perspectives for possible future developments.

The papers of Lanczos, Hestenes and Stiefel published in 1952–53 [53,62–64] went far beyond the construction of a numerical algorithm for solving linear algebraic systems. They also considered the approximation of eigenvalues, and covered related mathematical background including orthogonal polynomials, continued fraction expansions, and Gauss quadrature. The paper [63] considers “purification” of the starting vector of components along the large eigenvalues of the matrix A using Chebyshev polynomials, followed by application of a version of CG. It explains in detail the sharp distinction between the action of the Chebyshev polynomials and CG polynomials, where the former addresses the continuum of an interval while the latter “attenuate due to the nearness of their zeros” to the eigenvalues of A ; cf. [63, p. 46 and Fig. 2]. The paper [64] suggests in this context polynomial preconditioning much earlier than other publications, but this work has been largely overlooked; see also the commentaries [95,99]. The papers [53,62–64] also address quality measures of the computed iterative approximations. Remarkably, the early researchers also analyzed computer implementations of the methods, including the effects of rounding errors.

The papers of Karush [57] and Hayes [52] from 1952 and 1954, respectively, prove superlinear convergence of CG for $\mathcal{A} = \text{Id} + \mathcal{C}$, where Id is the identity and \mathcal{C} is a compact operator, concluding that the CG rate of convergence for such operators exceeds any given linear rate as the iteration proceeds. This result was, decades later, repeatedly rediscovered. A bit later (with the original print in Russian from 1958), Vorobyev put this new development into the context of a computationally feasible formulation of the method of moments [111]; see also the insightful paper by Brezinski [5] on its generalizations and the relationship with other related concepts, including the approach by Lanczos.

Unfortunately, many results of these early works remain almost unnoticed, and the common state-of-the-art literature even contains views that contradict some of them. On the other hand, a lack of theoretical understanding does not prohibit the practical application of the methods. Thus, the following quote from Forsythe’s paper from 1953 [24] is to some extent still valid for the class of Krylov subspace methods today:

“It is remarkable how little is really understood about most of the methods for solving $Ax = b$. They are being used nevertheless, and yield answers. This disparity between theory and practice appears to be typical of the gulf between the *science* and the *art* of numerical analysis.”

These words, written in 1952 and published in 1953, were prophetic for the situation in the area of Krylov subspace methods for decades to come.

Two common beliefs frequently repeated in the numerical analysis literature are that CG was originally considered as a direct method, and that it was, after unsatisfactory computational experiences, abandoned for about two decades as a numerical approach for linear systems. As mentioned above, CG was in the original papers considered as an iterative method. The story of considering it as direct method started later. Moreover, while the interest in CG within the numerical analysis community temporarily declined in the 1950s and 1960s (and was only revived later by Reid, Axelsson, Evans, Concus, Golub, O’Leary, Paige, Saunders, Meijerink, van der Vorst, and others in the 1970s), it actually was used by practitioners in the 1950s and 1960s; see, e.g., the collection of works given in [34]. Many important theoretical ideas appeared independently of numerical linear algebra in different fields; see, e.g., the developments in computational physics and chemistry [40] (cf. also the independent developments in [39]), [92,98], as well as in the works on matching moments leading to model reduction of large dynamical systems in [41].

The history of Krylov subspace methods illustrates discontinuities in the development of science and even loss of knowledge, as well as an unhealthy lack of communication between different fields.

A.2. Preconditioning and analysis of Krylov subspace methods

Practical computations with Krylov subspace methods require preconditioning. From a purely algebraical point of view, this is typically interpreted as the transformation of the original linear algebraic system into an equivalent system that is, for the given iterative method, more easily solvable. Another view is the acceleration of convergence of the computed approximations through transformation of the iterative process. For symmetric positive definite matrices and CG, the goal of preconditioning is typically stated as (see, e.g., [4]): “Hopefully, the transformed (preconditioned) matrix will have a smaller condition number, and/or eigenvalues clustered around one.” When this can be achieved, and a small condition number guarantees an acceptable approximate so-

lution within a few iterations (which can happen, e.g., for preconditioners constructed via domain decomposition methods incorporating coarse components) the matter seems to be resolved. However, if in such cases tight bounds are available for the extreme eigenvalues of the preconditioned matrix, then it can be computationally competitive or even more efficient to apply the Chebyshev semiiterative method instead of CG. This alternative approach is easily parallelizable and numerically stable. Similar to the quote above, the following insight of Forsythe [24, p. 318] is still valid and not fully appreciated today:

“The belief is widespread that the condition of a system $Ax = b$ has a decisive influence on the convergence of an iterative solution [...]; this can not always be true. Even when it is true for an iterative process, it may be possible actually to take advantage of the poor condition of $Ax = b$ in converting the slow process into an accelerated method which converges rapidly. There is a great need for clarification of the group of ideas associated with ‘condition.’”

As illustrated in Section 2.4 in this paper, the nonlinear adaptivity of CG to the location of the individual eigenvalues indicates that a smaller condition number does not necessarily lead to faster convergence. One of the examples in Section 2.4 is taken from the paper [30], which contains a substantial theoretical result on approximating *all eigenvalues of the preconditioned matrix*, and hence shows that information about all eigenvalues for important PDE problems can be available a priori at a negligible computational cost. In addition, recent results of Colbrook, Horning, and Townsend [9,10] show how to compute smoothed approximations of spectral measures for infinite dimensional self-adjoint operators.

A large family of preconditioners in numerical PDEs is motivated by the spectral or norm equivalence of operators in infinite dimensional Hilbert spaces (hence the name *operator preconditioning*); see, e.g., the beautiful and rarely quoted early papers by Concus and Golub published in 1973 [11], and by Faber, Manteuffel, and Parter published in 1990 [22]. Infinite dimensional operator preconditioning and algebraic preconditioning in the context of using CG are linked together in [72]. In particular, it is shown that any algebraic preconditioning can be put into the operator preconditioning framework. This reveals the close mathematical connection between the choice of the inner product in the infinite dimensional Hilbert spaces in operator preconditioning and the choice of the discretization basis determining the associated linear algebraic problem. A survey of the abstract framework of operator preconditioning based on decomposition of infinite dimensional Hilbert subspaces is given in [54].

Many results in the literature are devoted to problem-specific preconditioning. The overall state-of-the-art in the area of preconditioning still remains well-characterized by the following quote from the book of Saad published in 2003 [96, p. 297]:

“Finding a good preconditioner to solve a given sparse linear system is often viewed as a combination of art and science. Theoretical results are rare and some methods work surprisingly well, often despite expectations.”

This underlines the need for further research on preconditioning, which can not be separated from further analysis of Krylov subspace methods.

References

- [1] M. Arioli, V. Pták, Z. Strakoš, Krylov sequences of maximal length and convergence of GMRES, *BIT* 38 (1998) 636–643.
- [2] W.E. Arnoldi, The principle of minimized iteration in the solution of the matrix eigenvalue problem, *Q. Appl. Math.* 9 (1951) 17–29.
- [3] S.F. Ashby, T.A. Manteuffel, P.E. Saylor, A taxonomy for conjugate gradient methods, *SIAM J. Numer. Anal.* 27 (1990) 1542–1568.
- [4] M. Benzi, Preconditioning techniques for large linear systems: a survey, *J. Comput. Phys.* 182 (2002) 418–477.
- [5] C. Brezinski, The methods of Vorobyev and Lanczos, *Linear Algebra Appl.* 234 (1996) 21–41.
- [6] E. Carson, Z. Strakoš, On the cost of iterative computations, *Philos. Trans. R. Soc. A* 378 (2020) 20190050.
- [7] E.C. Carson, Communication-avoiding Krylov subspace methods in theory and practice, PhD thesis, University of California, Berkeley, 2015.
- [8] E.C. Carson, M. Rozložník, Z. Strakoš, P. Tichý, M. Tůma, The numerical stability analysis of pipelined conjugate gradient methods: historical context and methodology, *SIAM J. Sci. Comput.* 40 (2018) A3549–A3580.
- [9] M. Colbrook, A. Horning, Specsolve: spectral methods for spectral measures, arXiv preprint arXiv: 2201.01314, 2022.
- [10] M. Colbrook, A. Horning, A. Townsend, Computing spectral measures of self-adjoint operators, *SIAM Rev.* 63 (2021) 489–524.
- [11] P. Concus, G.H. Golub, Use of fast direct methods for the efficient numerical solution of nonseparable elliptic equations, *SIAM J. Numer. Anal.* 10 (1973) 1103–1120.
- [12] S. Cools, E.F. Yetkin, E. Agullo, L. Giraud, W. Vanroose, Analyzing the effect of local rounding error propagation on the maximal attainable accuracy of the pipelined conjugate gradient method, *SIAM J. Matrix Anal. Appl.* 39 (2018) 426–450.
- [13] J. Drkošová, A. Greenbaum, M. Rozložník, Z. Strakoš, Numerical stability of GMRES, *BIT* 35 (1995) 309–330.
- [14] V. Druskin, L. Knizhnerman, Krylov subspace approximation of eigenpairs and matrix functions in exact and computer arithmetic, *Numer. Linear Algebra Appl.* 2 (1995) 205–217.
- [15] J. Duintjer Tebbens, G. Meurant, H. Sadok, Z. Strakoš, On investigating GMRES convergence using unitary matrices, *Linear Algebra Appl.* 450 (2014) 83–107.
- [16] A. Edelman, The probability that a random real Gaussian matrix has k real eigenvalues, related distributions, and the circular law, *J. Multivar. Anal.* 60 (1997) 203–232.
- [17] H.C. Elman, A. Ramage, D.J. Silvester, IFISS: a computational laboratory for investigating incompressible flow problems, *SIAM Rev.* 56 (2014) 261–273.
- [18] H.C. Elman, D.J. Silvester, A.J. Wathen, *Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics*, Numerical Mathematics and Scientific Computation, second ed., Oxford University Press, Oxford, 2014.
- [19] M. Embree, The tortoise and the hare restart GMRES, *SIAM Rev.* 45 (2003) 259–266.
- [20] V. Faber, W. Joubert, E. Knill, T. Manteuffel, Minimal residual method stronger than polynomial preconditioning, *SIAM J. Matrix Anal. Appl.* 17 (1996) 707–729.
- [21] V. Faber, J. Liesen, P. Tichý, Properties of worst-case GMRES, *SIAM J. Matrix Anal. Appl.* 34 (2013) 1500–1519.
- [22] V. Faber, T.A. Manteuffel, S.V. Parter, On the theory of equivalent operators and application to the numerical solution of uniformly elliptic partial differential equations, *Adv. Appl. Math.* 11 (1990) 109–163.

- [23] R. Fletcher, Conjugate gradient methods for indefinite systems, in: Numerical Analysis (Proc 6th Biennial Dundee Conf., Univ. Dundee, Dundee, 1975), in: Lecture Notes in Math., vol. 506, Springer, Berlin, 1976, pp. 73–89.
- [24] G.E. Forsythe, Solving linear algebraic equations can be interesting, *Bull. Am. Math. Soc.* 59 (1953) 299–329.
- [25] R.W. Freund, G.H. Golub, N.M. Nachtigal, Iterative solution of linear systems, *Acta Numer.* 1 (1992) 57–100.
- [26] R.W. Freund, N.M. Nachtigal, QMR: a quasi-minimal residual method for non-Hermitian linear systems, *Numer. Math.* 60 (1991) 315–339.
- [27] F.R. Gantmacher, On the algebraic analysis of Krylov’s method of transforming the secular equation, in: Trans. Second Math. Congress, II, 1934, pp. 45–48 (in Russian).
- [28] T. Gergelits, Analysis of Krylov subspace methods, Master’s thesis, Charles University, Faculty of Mathematics and Physics, 2013.
- [29] T. Gergelits, I. Hnětynková, M. Kubínová, Relating computed and exact entities in methods based on Lanczos tridiagonalization, in: International Conference on High Performance Computing in Science and Engineering, Springer, 2017, pp. 73–87.
- [30] T. Gergelits, K.-A. Mardal, B.F. Nielsen, Z. Strakoš, Laplacian preconditioning of elliptic PDEs: localization of the eigenvalues of the discretized operator, *SIAM J. Numer. Anal.* 57 (2019) 1369–1394.
- [31] T. Gergelits, B.r.F. Nielsen, Z. Strakoš, Numerical approximation of the spectrum of self-adjoint operators in operator preconditioning, *Numer. Algorithms* 91 (2022) 301–325.
- [32] L. Giraud, J. Langou, M. Rozložník, J. van den Eshof, Rounding error analysis of the classical Gram-Schmidt orthogonalization process, *Numer. Math.* 101 (2005) 87–100.
- [33] G.H. Golub, G. Meurant, Matrices, moments and quadrature. II. How to compute the norm of the error in iterative methods, in: Direct Methods, Linear Algebra in Optimization, Iterative Methods, Toulouse, 1995/1996, *BIT* 37 (1997) 687–705.
- [34] G.H. Golub, D.P. O’Leary, Some history of the conjugate gradient and Lanczos algorithms: 1948–1976, *SIAM Rev.* 31 (1989) 50–102.
- [35] G.H. Golub, Z. Strakoš, Estimates in quadratic formulas, *Numer. Algorithms* 8 (1994) 241–268.
- [36] G.H. Golub, H.A. van der Vorst, Closer to the solution: iterative linear solvers, in: The State of the Art in Numerical Analysis, York, 1996, in: Inst. Math. Appl. Conf. Ser. New Ser., vol. 63, Oxford Univ. Press, New York, 1997, pp. 63–92.
- [37] G.H. Golub, R.S. Varga, Chebyshev semi-iterative methods, successive over-relaxation iterative methods, and second order Richardson iterative methods. I, *Numer. Math.* 3 (1961) 147–156.
- [38] G.H. Golub, R.S. Varga, Chebyshev semi-iterative methods, successive over-relaxation iterative methods, and second order Richardson iterative methods. II, *Numer. Math.* 3 (1961) 157–168.
- [39] G.H. Golub, J.H. Welsch, Calculation of Gauss quadrature rules, *Math. Comput.* 23 (1969) 221–230; *Math. Comput.* 23 (1969) A1–A10 (Addendum).
- [40] R.G. Gordon, Error bounds in equilibrium statistical mechanics, *J. Math. Phys.* 9 (1968) 655–663.
- [41] W.B. Gragg, Matrix interpretations and applications of the continued fraction algorithm, *Rocky Mt. J. Math.* 4 (1974) 213–225.
- [42] A. Greenbaum, Comparison of splittings used with the conjugate gradient algorithm, *Numer. Math.* 33 (1979) 181–193.
- [43] A. Greenbaum, Behavior of slightly perturbed Lanczos and conjugate-gradient recurrences, *Linear Algebra Appl.* 113 (1989) 7–63.
- [44] A. Greenbaum, Iterative Methods for Solving Linear Systems, *Frontiers in Applied Mathematics*, vol. 17, SIAM, Philadelphia, PA, 1997.
- [45] A. Greenbaum, L. Gurvits, Max-min properties of matrix factor norms, *SIAM J. Sci. Comput.* 15 (1994) 348–358.
- [46] A. Greenbaum, V. Pták, Z. Strakoš, Any nonincreasing convergence curve is possible for GMRES, *SIAM J. Matrix Anal. Appl.* 17 (1996) 465–469.
- [47] A. Greenbaum, Z. Strakoš, Predicting the behavior of finite precision Lanczos and conjugate gradient computations, *SIAM J. Matrix Anal. Appl.* 13 (1992) 121–137.
- [48] A. Greenbaum, Z. Strakoš, Matrices that generate the same Krylov residual spaces, in: Recent Advances in Iterative Methods, in: IMA Vol. Math. Appl., vol. 60, Springer, New York, 1994, pp. 95–118.
- [49] A. Greenbaum, L.N. Trefethen, GMRES/CR and Arnoldi/Lanczos as matrix approximation problems, *SIAM J. Sci. Comput.* 15 (1994) 359–368.

- [50] M.H. Gutknecht, Z. Strakoš, Accuracy of two three-term and three two-term recurrences for Krylov space solvers, *SIAM J. Matrix Anal. Appl.* 22 (2000) 213–229.
- [51] W. Hackbusch, *Iterative Solution of Large Sparse Systems of Equations*, Applied Mathematical Sciences, vol. 95, Springer-Verlag, New York, 1994. Translated and revised from the 1991 German original.
- [52] R.M. Hayes, *Iterative Methods for Solving Linear Problems in Hilbert Space*, PhD thesis, Univ. of California at Los Angeles, Los Angeles, CA, 1954.
- [53] M.R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, *J. Res. Natl. Bur. Stand.* 49 (1952) 409–436.
- [54] J. Hrnčíř, I. Pultarová, Z. Strakoš, Decomposition into subspaces preconditioning: abstract framework, *Numer. Algorithms* 83 (2020) 57–98.
- [55] A. Jennings, Influence of the eigenvalue spectrum on the convergence rate of the conjugate gradient method, *J. Inst. Math. Appl.* 20 (1977) 61–72.
- [56] W. Joubert, A robust GMRES-based adaptive polynomial preconditioning algorithm for nonsymmetric linear systems, *SIAM J. Sci. Comput.* 15 (1994) 427–439.
- [57] W. Karush, Convergence of a method of solving linear problems, *Proc. Am. Math. Soc.* 3 (1952) 839–851.
- [58] A.N. Krylov, On the numerical solution of the equation by which the frequency of small oscillations is determined in technical problems, *Izv. Akad. Nauk SSSR, Ser. Fiz.-Mat.* 4 (1931) 491–539 (in Russian).
- [59] A.B.J. Kuijlaars, Which eigenvalues are found by the Lanczos method?, *SIAM J. Matrix Anal. Appl.* 22 (2000) 306–321.
- [60] M. Ladecký, R.J. Leute, A. Falsafi, I. Pultarová, L. Pastewka, T. Junge, J. Zeman, An optimal preconditioned FFT-accelerated finite element solver for homogenization, *Appl. Math. Comput.* 446 (2023) 127835.
- [61] M. Ladecký, I. Pultarová, J. Zeman, Guaranteed two-sided bounds on all eigenvalues of preconditioned diffusion and elasticity problems solved by the finite element method, *Appl. Math.* 66 (2021) 21–42.
- [62] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, *J. Res. Natl. Bur. Stand.* 45 (1950) 255–282.
- [63] C. Lanczos, Solution of systems of linear equations by minimized iterations, *J. Res. Natl. Bur. Stand.* 49 (1952) 33–53.
- [64] C. Lanczos, Chebyshev polynomials in the solution of large-scale linear systems, in: *Proceedings of the Association for Computing Machinery, Toronto, 1952*, Sauls Lithograph Co. (for the Association for Computing Machinery), Washington, D. C., 1953, pp. 124–133.
- [65] R.J. Leute, M. Ladecký, A. Falsafi, I. Jödicke, I. Pultarová, J. Zeman, T. Junge, L. Pastewka, Elimination of ringing artifacts by finite-element projection in FFT-based homogenization, *J. Comput. Phys.* 453 (2022) 110931.
- [66] J. Liesen, Z. Strakoš, GMRES convergence analysis for a convection-diffusion model problem, *SIAM J. Sci. Comput.* 26 (2005) 1989–2009.
- [67] J. Liesen, Z. Strakoš, *Krylov Subspace Methods, Numerical Mathematics and Scientific Computation. Principles and Analysis*, Oxford University Press, Oxford, 2013.
- [68] J. Liesen, P. Tichý, Convergence analysis of Krylov subspace methods, *GAMM Mitt. Ges. Angew. Math. Mech.* 27 (2004) 153–173.
- [69] J. Liesen, P. Tichý, Max-min and min-max approximation problems for normal matrices revisited, *Electron. Trans. Numer. Anal.* 41 (2014) 159–166.
- [70] L. Lin, Y. Saad, C. Yang, Approximating spectral densities of large matrices, *SIAM Rev.* 58 (2016) 34–65.
- [71] N.N. Luzin, On Krylov’s method for transforming the secular equation, *Izv. Akad. Nauk SSSR, Ser. Fiz.-Mat.* 7 (1931) 903–958.
- [72] J. Málek, Z. Strakoš, *Preconditioning and the Conjugate Gradient Method in the Context of Solving PDEs*, *SIAM Spotlights*, vol. 1, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2015.
- [73] G. Meurant, *The Lanczos and Conjugate Gradient Algorithms. From Theory to Finite Precision Computations, Software, Environments, and Tools*, vol. 19, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2006.
- [74] G. Meurant, On prescribing the convergence behavior of the conjugate gradient algorithm, *Numer. Algorithms* 84 (2020) 1353–1380.

- [75] G. Meurant, J. Duintjer Tebbens, Krylov Methods for Nonsymmetric Linear Systems—from Theory to Computations, Springer Series in Computational Mathematics, vol. 57, Springer, Cham, 2020. [2020] ©2020.
- [76] G. Meurant, J. Papež, P. Tichý, Accurate error estimation in CG, Numer. Algorithms 88 (2021) 1337–1359.
- [77] G. Meurant, Z. Strakoš, The Lanczos and conjugate gradient algorithms in finite precision arithmetic, Acta Numer. 15 (2006) 471–542.
- [78] P. Morin, R.H. Nochetto, K.G. Siebert, Convergence of adaptive finite element methods, SIAM Rev. 44 (2002) 631–658. Revised reprint of “Data oscillation and convergence of adaptive FEM” [SIAM J. Numer. Anal. 38 (2) (2000) 466–488. MR1770058 (2001g:65157) (electronic)].
- [79] M.F. Murphy, G.H. Golub, A.J. Wathen, A note on preconditioning for indefinite linear systems, SIAM J. Sci. Comput. 21 (2000) 1969–1972.
- [80] B.F. Nielsen, Z. Strakoš, A simple formula for the generalized spectrum of second order self-adjoint differential operators, SIAM Rev. (2024).
- [81] B.F. Nielsen, A. Tveito, W. Hackbusch, Preconditioning by inverting the Laplacian: an analysis of the eigenvalues, IMA J. Numer. Anal. 29 (2009) 24–42.
- [82] Y. Notay, On the convergence rate of the conjugate gradients in presence of rounding errors, Numer. Math. 65 (1993) 301–317.
- [83] C.C. Paige, Accuracy and effectiveness of the Lanczos algorithm for the symmetric eigenproblem, Linear Algebra Appl. 34 (1980) 235–258.
- [84] C.C. Paige, M. Rozložník, Z. Strakoš, Modified Gram-Schmidt (MGS), least squares, and backward stability of MGS-GMRES, SIAM J. Matrix Anal. Appl. 28 (2006) 264–284.
- [85] C.C. Paige, M.A. Saunders, Solution of sparse indefinite systems of linear equations, SIAM J. Numer. Anal. 12 (1975) 617–629.
- [86] C.C. Paige, Z. Strakoš, Bounds for the least squares distance using scaled total least squares, Numer. Math. 91 (2002) 93–115.
- [87] C.C. Paige, Z. Strakoš, Residual and backward error bounds in minimum residual Krylov subspace methods, SIAM J. Sci. Comput. 23 (2002) 1898–1923.
- [88] J.W. Pearson, J. Pestana, Preconditioners for Krylov subspace methods: an overview, GAMM-Mitt. 43 (2020) e202000015.
- [89] I. Pultarová, M. Ladecký, Two-sided guaranteed bounds to individual eigenvalues of preconditioned finite element and finite difference problems, Numer. Linear Algebra Appl. 28 (2021) e2382.
- [90] L. Reichel, L.N. Trefethen, Eigenvalues and pseudo-eigenvalues of Toeplitz matrices, Linear Algebra Appl. 162 (164) (1992) 153–185.
- [91] J.K. Reid, On the method of conjugate gradients for the solution of large sparse systems of linear equations, in: Large Sparse Sets of Linear Equations (Proc. Conf., St. Catherine’s Coll., Oxford, 1970), Academic Press, London, New York, 1971, pp. 231–254.
- [92] W.P. Reinhardt, l^2 discretization of atomic and molecular electronic continua: moment, quadrature and j -matrix techniques, Comput. Phys. Commun. 17 (1979) 1–21.
- [93] J.-L. Rigal, J. Gaches, On the compatibility of a given solution with the data of a linear system, J. Assoc. Comput. Mach. 14 (1967) 543–548.
- [94] Y. Saad, Krylov subspace methods for solving large unsymmetric linear systems, Math. Comput. 37 (1981) 105–126.
- [95] Y. Saad, Commentary on Lanczos’s “Chebyshev polynomials in the solution of large-scale linear systems”, in: Cornelius Lanczos Collected Published Papers with Commentaries, vol. VI, North Carolina State University, Raleigh, North Carolina, 1998, pp. 3-527–3-528.
- [96] Y. Saad, Iterative Methods for Sparse Linear Systems, second ed., Society for Industrial and Applied Mathematics, Philadelphia, PA, 2003.
- [97] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (1986) 856–869.
- [98] L. Schlessinger, C. Schwartz, Analyticity as a useful computational tool, Phys. Rev. Lett. 16 (1966) 1173–1174.
- [99] G.W. Stewart, Lanczos and linear systems, University of Maryland, Preprint UMIACS-TR-91, 1991.
- [100] Z. Strakoš, On the real convergence rate of the conjugate gradient method, Linear Algebra Appl. 154 (156) (1991) 535–549.
- [101] Z. Strakoš, Model reduction using the Vorobyev moment problem, Numer. Algorithms 51 (2009) 363–379.

- [102] Z. Strakoš, P. Tichý, On error estimation in the conjugate gradient method and why it works in finite precision computations, *Electron. Trans. Numer. Anal.* 13 (2002) 56–80.
- [103] Z. Strakoš, P. Tichý, Error estimation in preconditioned conjugate gradients, *BIT* 45 (2005) 789–817.
- [104] K.-C. Toh, GMRES vs. ideal GMRES, *SIAM J. Matrix Anal. Appl.* 18 (1997) 30–36.
- [105] L.N. Trefethen, Pseudospectra of matrices, in: *Numerical Analysis 1991*, Dundee, 1991, in: *Pitman Res. Notes Math. Ser.*, vol. 260, Longman Sci. Tech., Harlow, 1992, pp. 234–266.
- [106] L.N. Trefethen, D. Bau III, *Numerical Linear Algebra*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.
- [107] L.N. Trefethen, M. Embree, *Spectra and Pseudospectra. The Behavior of Nonnormal Matrices and Operators*, Princeton University Press, Princeton, NJ, 2005.
- [108] A. van der Sluis, H.A. van der Vorst, The rate of convergence of conjugate gradients, *Numer. Math.* 48 (1986) 543–560.
- [109] H.A. van der Vorst, Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.* 13 (1992) 631–644.
- [110] H.A. van der Vorst, *Iterative Krylov Methods for Large Linear Systems*, Cambridge Monographs on Applied and Computational Mathematics, vol. 13, Cambridge University Press, Cambridge, 2003.
- [111] Y.V. Vorobyev, *Methods of Moments in Applied Mathematics*, Gordon and Breach Science Publishers, New York, 1965. Translated from the Russian by Bernard Seckler.
- [112] P. Wesseling, P. Sonneveld, Numerical experiments with a multiple grid and a preconditioned Lanczos type method, in: *Approximation Methods for Navier-Stokes Problems (Proc. Sympos., Univ. Paderborn, Paderborn, 1979)*, in: *Lecture Notes in Math.*, vol. 771, Springer, Berlin, 1980, pp. 543–562.
- [113] J.H. Wilkinson, *The Algebraic Eigenvalue Problem*, Monographs on Numerical Analysis, The Clarendon Press Oxford University Press, New York, 1988, 1st ed. published 1965.
- [114] J.H. Wilkinson, *Rounding Errors in Algebraic Processes*, Dover Publications Inc., New York, 1994. Reprint of the 1963 original published by Prentice-Hall, Englewood Cliffs, NJ.