ZAMM · Z. Angew. Math. Mech. **85**, No. 5, 307 – 325 (2005) / **DOI** 10.1002/zamm.200410185

On numerical stability in large scale linear algebraic computations

Plenary lecture presented at the 75th Annual GAMM Conference, Dresden/Germany, 22–26 March 2004

Z. Strakoš[1,*] **and J. Liesen**[2,**]

[1] Institute of Computer Science, Academy of Sciences of the Czech Republic, Pod Vodárenskou věží 2, 182 00 Prague 8, Czech Republic

[2] Institute of Mathematics, Technical University of Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany

Received 13 May 2004, accepted 5 December 2004
Published online 7 April 2005

Key words linear algebraic systems, eigenvalue problems, convergence, numerical stability, backward error, accuracy, Lanczos method, conjugate gradient method, GMRES method
MSC (2000) 65F10, 65F15, 65G20, 65G50

Numerical solving of real-world problems typically consists of several stages. After a mathematical description of the problem and its proper reformulation and discretisation, the resulting linear algebraic problem has to be solved. We focus on this last stage, and specifically consider numerical stability of iterative methods in matrix computations. In iterative methods, rounding errors have two main effects: They can delay convergence and they can limit the maximal attainable accuracy. It is important to realize that numerical stability analysis is not about derivation of error bounds or estimates. Rather the goal is to find algorithms and their parts that are safe (numerically stable), and to identify algorithms and their parts that are not. Numerical stability analysis demonstrates this important idea, which also guides this contribution. In our survey we first recall the concept of backward stability and discuss its use in numerical stability analysis of iterative methods. Using the backward error approach we then examine the surprising fact that the accuracy of a (final) computed result may be much higher than the accuracy of intermediate computed quantities. We present some examples of rounding error analysis that are fundamental to justify numerically computed results. Our points are illustrated on the Lanczos method, the conjugate gradient (CG) method and the generalised minimal residual (GMRES) method.

1 Introduction

Numerical solution of real-world problems, sometimes labelled as *scientific computing*, combines tools from the areas of a given application, applied mathematics, numerical analysis, numerical methods, matrix computations, and computer science. For example, a part of reality can be described (in mathematical abstraction) by a system of differential and/or integral equations. After choosing a proper formulation of the mathematical model, the existence and uniqueness of its analytic solution is investigated. Subsequently, the continuous problem is discretised. Coefficients determining the discrete approximation are then computed by solving a linear algebraic problem. At all stages the approximation steps are accompanied by errors. The main types of errors are approximation errors of the model, discretisation errors of the finite dimensional formulation, and truncation and/or rounding errors of the numerical solution of the linear algebraic problem.

1.1 Errors in mathematical modelling and scientific computing

The stages in the solution of a typical real-world problem described by differential equations are schematically shown in Fig. 1 and Fig. 2. Any successful solution process starts and ends at the real-world problem stage. Going down the structure represents constructing an approximate solution. Going up represents an interpretation of the results which should always include understanding of the errors. The analysis of errors in the part of the process starting and ending with the mathematical model is called *verification* in the PDE literature. It aims to verify that the equations constituting the mathematical model were solved correctly (modulo an acceptable inaccuracy). *Validation* of a mathematical model, on the other hand, asks to which extent the mathematical model and its numerical solution describe the real-world problem, see, e.g., the discussion by Babuška [7] and Oden et al. [49]. Each stage of the solution process requires its own knowledge and expertise. A mistake at any of the stages can hardly be compensated for by excellence at the others.

The PDE literature on error analysis typically does not consider the specific contribution of the last stage (truncation and/or rounding errors). It often concentrates only on discretisation errors (for an example of a more general discussion we refer to [81]). A somewhat paradoxical nature of this fact with respect to rounding errors was pointed out by Parlett

[*] Corresponding author, e-mail: strakos@cs.cas.cz, Phone: +00 420 266 053 290, Fax: +00 420 286 585 789

[**] Second author, e-mail: liesen@math.tu-berlin.de

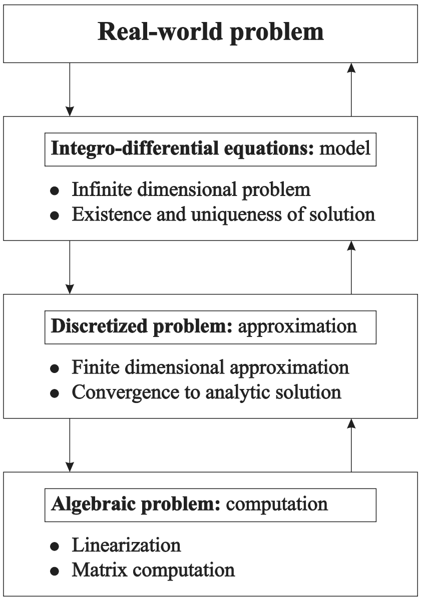
Real-world problem
Integro-differential equations: model
• Infinite dimensional problem
• Existence and uniqueness of solution
Discretized problem: approximation
• Finite dimensional approximation
• Convergence to analytic solution
Algebraic problem: computation
• Linearization
• Matrix computation

Fig. 1 Stages in the numerical solution process of a real-world problem.

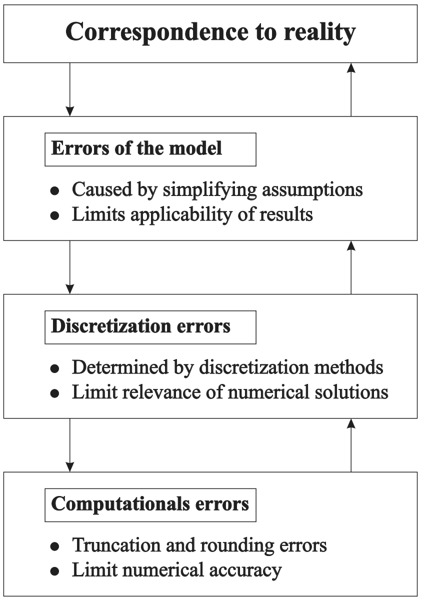
Correspondence to reality
Errors of the model
• Caused by simplifying assumptions
• Limits applicability of results
Discretization errors
• Determined by discretization methods
• Limit relevance of numerical solutions
Computationals errors
• Truncation and rounding errors
• Limit numerical accuracy

Fig. 2 Errors in the numerical solution process of a real-world problem.

in his essay devoted to the work of Wilkinson [61, pp. 19–20]. Numerical stability and condition number checks as well as knowledge-based recommendations concerning solvers are listed among the missing features of general purpose finite element programs for structural mechanics, cf. the recent monograph edited by Stein [69, pp. 3–4]. The situation in other application areas is not significantly different. When the error at the computational stage is not properly integrated into the error analysis of the whole solution process, assuming that the computational stage provides (or with a high accuracy approximates) the *exact solution* of the discretised problem, we may have to deal with the following possible consequences:

- Either the computation of the approximate solution of the algebraic problem consumes unnecessary time and resources by aiming at an unnecessarily high accuracy,
- or a computational error which is not under control impinges on the other stages and spoils the numerical solution.

The first consequence can limit the size or the level of detail of the model by negatively affecting the required computation time. The second consequence is even more dangerous. In the worst case it can lead to wrong (e.g. physically incorrect) results that have little or no relation to the actual real-world problem.

1.2 Specification of the subject

We concentrate on methods for solving large linear algebraic systems of the form

$$Ax = b\,, \tag{1}$$

where A is an N by N real or complex matrix, and the right hand side b is a real or complex vector of length N. We also consider the related problem of computing eigenvalues of the matrix A. The title of our paper reflects its content and it is worth three comments:

1. *Numerical stability* analysis, as explained above, is an essential part of scientific computing. Unless rounding errors are kept under control, things may go wrong due to numerical instabilities.
2. *Large scale* means that we consider large problems that arise in real-world applications. Their solution process can typically not be based on standard textbook algorithms that are applied in the style of cookbook recipes. Rather these problems require expertise from many areas. The numerical linear algebra part, in particular, requires the combination of iterative and direct methods. By combining both, we can strengthen their advantages and suppress their weaknesses. For example, direct techniques, such as incomplete factorisations and approximate inverses, may greatly enhance the error reduction capability of individual iterations at the price of making the iterations more expensive. Direct techniques may also increase robustness of the combined solver. The principal contribution of the iterative part is the possibility of stopping at some desired accuracy level. This, however, requires a meaningful stopping criterion which balances computational errors with discretisation errors and other errors in the solution process.

3. *Linear* attributes to the problem to be solved (linear algebraic systems or eigenvalue problems), not to phenomena which must be dealt with in the process of construction, analysis and application of modern iterative methods. In fact, modern iterative methods of numerical linear algebra, such as Krylov subspace methods, are strongly nonlinear.

1.3 Characterisation of convergence is a nonlinear problem

Answering the question as to how fast we can get an acceptable approximate solution in modern large scale linear algebraic solvers, such as preconditioned Krylov subspace methods, requires approaches radically different from the convergence theory of classical iterative methods, such as SOR or Chebyshev semiiteration. As pointed out for example by Hackbusch [34, p. 270], the terms *convergence* and *asymptotical convergence rate* lose their meaning, because Krylov subspace methods typically reach (in exact precision) the exact solution in a finite number of iterations. Hence no limit can be formed. In finite precision arithmetic this finite termination property is lost. But this is not why we consider Krylov subspace methods, such as the conjugate gradient (CG) method [36] and its generalisations, *iterative*. Rather the reason is that these methods are of practical interest only if a sufficiently accurate approximate solution is found in a small number of iterations (usually significantly smaller than the dimension N of the linear algebraic problem). Consequently, we must study the method's behaviour from the first iteration, which generally represents a very difficult nonlinear phenomenon in a finite-dimensional space. Even in the symmetric positive definite case, the convergence of the CG method does not only depend on the spectrum of A but also on the right hand side of the linear system, which is related to boundary conditions and the external field. For interesting examples related to the one-dimensional Poisson equation we refer to the work of Beckermann and Kuijlaars [9], see also Liesen and Tichý [46]. In more general cases the situation is significantly more complicated, since the spectrum or other simple characteristics of the matrix cannot be relied upon as an indicator of the convergence behaviour, partially because the role of the specific right hand sides can be much more pronounced.

Such difficulties can be demonstrated on the following simple two-dimensional convection-diffusion model problem,

$$-\nu\Delta u + w \cdot \nabla u = 0 \quad \text{in} \quad \Omega\,, \tag{2}$$

$$u = g \quad \text{on} \quad \partial\Omega\,, \tag{3}$$

where the scalar valued function $u(\eta_1, \eta_2)$ represents the concentration of the transported quantity, $w = [w_1, w_2]^T$ the wind, ν the scalar diffusion parameter, and Ω the unit square. When the problem is convection-dominated, i.e. $\nu \ll \|w\|$, the Galerkin finite element discretisation leads to nonphysical oscillations of the discretised solution. This has been known for several decades, and the model problem (2)–(3) has for many years been used to test various stabilisation techniques such as the streamline upwind Petrov Galerkin (SUPG) discretisation, see [38, 11, 48]. For a recent description and examples based on bilinear finite elements and a regular grid we refer to the work of Elman and Ramage [19, 20]. The resulting linear algebraic systems have also been used as challenging examples for convergence analysis of iterative solvers. For example, the generalised minimal residual (GMRES) method [67] applied to such systems typically exhibits an initial period of slow convergence followed by a faster decrease of the residual norm. Ernst conjectured in [21] that the duration of the initial phase is governed by the number of steps needed for boundary information to pass from the inflow boundary across the (discretised) domain following the longest streamline of the velocity field. He also illustrated that for these PDE-related linear algebraic problems eigenvalues alone give misleading information about convergence. He focused in his analysis on the field of values. Using the eigendecomposition of the discretised operator, Fischer et al. analysed in [23] the choice of parameters in the SUPG discretisation and their relation to convergence of GMRES. Since the analyses in [21] and [23] are based on the discretised operator only, they can not explain the dependence of the length of the initial period of slow convergence on the paricular right hand side of the linear system, and hence on the boundary conditions. Using properly chosen operator-based tools such as the polynomial numerical hulls [33], it is possible, however, to describe the worst case convergence behaviour.

In our paper [45], see also [44], we consider a regular grid with bilinear elements, and a wind aligned with the η_2-axis, i.e. $w = [0, 1]^T$. The eigenvalues and eigenvectors of the discretised operator are known analytically, but the transformation to the eigenvector coordinates is highly ill-conditioned. Therefore any analysis based on this transformation must involve a rather complicated pattern of cancellation of potentially huge components of the initial residual (right hand side) in the individual eigenspaces, otherwise the results are quantitatively useless. Instead of using this technically complicated and physically unnatural approach, we propose another idea. Assume that a well-conditioned transformation of a given linear algebraic system yields a new system with a structure of the matrix, not necessarily diagonal, for which the GMRES convergence can with the transformed right hand side easily be analysed. Then the geometry of the space is not significantly distorted by the transformation, and using the particular structure of the transformed system we can describe the GMRES convergence for the original problem. Following [16, 19, 20], the transformation used in [45] is orthonormal, and the transformed system is block diagonal with tridiagonal Toeplitz blocks. The GMRES convergence for individual tridiagonal Toeplitz systems is then analysed by linking it to the GMRES convergence for scaled Jordan blocks. This is possible because of the dominance of convection over diffusion in the model problem. Such approach clearly describes the relationship between the boundary conditions in the model problem and the initial phase of slow GMRES convergence for the discretised algebraic system. It cannot, however, be used for the subsequent phase of convergence. Although [45] presents some preliminary qualitative considerations, that problem still remains open.

1.4 Main focus and organisation of the paper

Rounding errors can delay convergence and limit the maximal attainable accuracy. In solving linear algebraic systems arising from mathematical modelling of real-world problems, the required accuracy is usually not high, and therefore limitations of the maximal attainable accuracy typically need not be considered. Still, numerical stability analysis is fundamental *to justify the accuracy of the computed results*.

Our paper is organised as follows. Sect. 2 presents the backward error as an illustration of backward stability analysis. An interesting consequence is given in Sect. 3: The number of significant digits in the intermediate quantities computed in finite precision arithmetic may be quite irrelevant to the accuracy of the final output. Sect. 4 presents examples of the link between numerical stability and computational cost, as well as an example of stopping criteria justified by rounding error analysis. Closing remarks summarise the most important points.

2 Backward error and backward stability

At the algebraic stage of the solution process of a real-world problem, cf. Fig. 1, a goal is to find an approximate solution for the linear algebraic system (1). We assume that the system is large, which requires incorporation of an iterative method in combination with direct techniques such as preconditioning. The principal questions are how the accuracy of the approximate algebraic solution should be measured and when the iteration should be stopped.

Clearly, a stopping procedure must include a reliable evaluation of the computational error combining two components: The truncation error due to preliminary stopping of the iteration, and rounding errors. Whenever we stop iterations using some stopping criteria, we must know whether the computed approximation gives *relevant* information about the solution of the real-world problem despite the presence of rounding errors. As mentioned above, we will not consider cases in which the maximal attainable accuracy plays a role in the evaluation of the computational error.

In an ideal situation errors at all three stages (model – discretisation – computation) should be in balance. Suppose that we have a perturbation theory of the model, and that we are able to express the discretisation errors and computational errors *backwards* as perturbations of the original model. Then it seems reasonable to stop the iteration process on the algebraic level when the discretisation and computational contributions to the whole backward error are in a desired proportion (which is problem dependent) to the error of the model.

We apply the concept of perturbations and backward error, which was fully developed in the fundamental work of Wilkinson [79, 80] in the context of numerical methods for solving algebraic problems. Due to the error of the mathematical model and the discretisation error, cf. Fig. 2, the resulting particular linear algebraic system $Ax = b$ represents *a whole class* of admissible systems. Each system in this class corresponds (possibly in a stochastic sense) to the original real-world problem. Differences between linear systems in this class (or, say, between $Ax = b$ and any other system in this class) correspond to the size of the model and discretisation errors. For example, the values of material constants or some other characteristics used in the formulation of the mathematical model are often determined only to one or two digits of accuracy. Subsequently, replacing the infinite dimensional problem (PDE) by a finite dimensional one introduces (part of) the discretisation error. Additional errors occur when the entries of A and b have to be computed with the help of numerical quadrature.

As a consequence, with respect to the original real-world problem, a solution $\tilde{x}$ of

$$(A + \Delta A)\,\tilde{x} = b + \Delta\, b \qquad (4)$$

is as good as the solution x of $Ax = b$ when the perturbations ΔA and Δb are *small*.

2.1 Relative residual and normwise backward error

Consider an approximate solution x_n computed at the nth iteration of an iterative algorithm. Then

$$Ax_n = b - r_n\,, \qquad r_n = b - Ax_n\,. \qquad (5)$$

Thus $-r_n$ represents the unique perturbation Δb of the right hand side b such that x_n is the exact solution of the perturbed system

$$Ax_n = b + \Delta b\,. \tag{6}$$

The relative size of the perturbation restricted to the right hand side is $\|r_n\|/\|b\|$ ($\|\cdot\|$ in this paper denotes the Euclidean norm, but any other norm could be used here too). With $x_0 = 0$ this represents the widely used relative residual norm $\|r_n\|/\|r_0\|$. With $x_0 \neq 0$ the relative residual norm lacks this backward error interpretation, and for $\|r_0\| \gg \|b\|$ it represents a rather insecure measure of convergence. In fact, a nonzero x_0 containing no useful information about x, e.g. a random x_0, might lead to a completely "biased" r_0 with $\|r_0\| \gg \|b\|$. Such a choice potentially creates an illusion of fast convergence to a high relative accuracy, all measured by the relative residual norm. For examples see [59, relation (2.8), and the discussion of Figs. 7.9 and 7.10], where the source and the danger of such illusions is outlined. Hegedüs [35] suggested that a simple way around this difficulty is to rescale the initial approximation. Given a preliminary initial guess x_p, it is easy to determine the scaling parameter $\zeta_{\min}$ such that

$$\|r_0\| \;=\; \|b - Ax_p\zeta_{\min}\| \;=\; \min_{\zeta} \|b - Ax_p\zeta\|, \qquad \zeta_{\min} \;=\; \frac{b^* Ax_p}{\|Ax_p\|^2}\,. \tag{7}$$

Thus, by setting $x_0 = x_p\zeta_{\min}$, we ensure $\|r_0\| \leq \|b\|$. The extra cost for implementing this little trick is negligible; it should be used whenever a nonzero x_0 is considered. Still, x_p should be based on information about the problem, otherwise it can, even with (7), delay convergence.

The cases in which b is inaccurate while A is known accurately are rather rare. Therefore we need to allow perturbations in both A and b. The *backward error* for x_n as an approximate solution for $Ax = b$ is a measure of the amounts by which both A and b have to be perturbed so that x_n is the exact solution of the perturbed system

$$(A + \Delta A)\, x_n = b + \Delta b\,. \tag{8}$$

As shown by Rigal and Gaches [63], also see [37, Theorem 7.1], the *normwise relative backward error* of x_n, defined by

$$\beta(x_n) \;\equiv\; \min\{\, \beta \,:\, (A + \Delta A)\, x_n = b + \Delta b\,,\; \|\Delta A\| \leq \beta\|A\|\,,\; \|\Delta b\| \leq \beta\|b\|\}\;, \tag{9}$$

satisfies

$$\beta(x_n) = \frac{\|r_n\|}{\|b\| + \|A\|\;\|x_n\|} = \frac{\|\Delta A_{\min}\|}{\|A\|} = \frac{\|\Delta b_{\min}\|}{\|b\|}\,. \tag{10}$$

In other words, $\beta(x_n)$ is equal to the norm of the *smallest* relative perturbations in A and b such that x_n exactly solves the perturbed system.

We strongly believe that if no other (more problem-specific and more sophisticated, see [2, 3, 4]) criterion is available, this relative backward error should always be preferred to the (relative) residual norm $\|r_n\|/\|r_0\|$. In practice $\|A\|$ has to be replaced by some approximation – when available – or simply by the Frobenius norm of A. The theoretical reasons for preferring the relative backward error are well known, see for example [1, 37] and also [4, 3]. In [53], the backward error idea has been used to derive a family of stopping criteria which quantify levels of confidence in A and b. These stopping criteria have then been implemented in generally available software [54] for solving linear algebraic systems and least squares problems. The relative normwise backward error is recommended and used by numerical analyst, see for example [8, 24]. It is known that the residual norm can be very misleading and easily misinterpreted. It is surprising and somewhat alarming that $\|r_n\|/\|r_0\|$ remains in use as the main (and usually the only) indicator of convergence of iterative processes.

If the backward error is small, the computed approximate solution x_n is an *exact solution of a nearby problem*. The forward error $\|x - x_n\|$ can be bounded using perturbation theory, see [37] for a collection of corresponding results. But the size of the worst-case bounds for $\|x - x_n\|$, though an important indicator of possible inaccuracies, does not always tell the whole story. For ill-conditioned matrices, for example, x_n can be computed with a small backward error $\beta(x_n)$. The corresponding perturbation bound for $\|x - x_n\|$ may not ensure, however, a single digit of accuracy of the computed approximate solution x_n, when compared with the exact solution x of the (possibly inaccurate) algebraic problem (1). Still, x_n can be perfectly acceptable approximate solution with respect to the underlying real-world problem.

It should be noted that normwise backward errors ignore the structure of nonzero elements of A, as well as the relative size and importance of the individual entries in A and b. An alternative is using componentwise backward errors, see [1, 37]. For A large and sparse, however, the use of componentwise criteria can become expensive. Moreover, it is not clear whether the componentwise approach is in general preferable to the normwise approach. This is particularly true in light of the nature of iterations with matrix-vector products as basic building blocks, as well as in the context of model and discretisation errors.

2.2 A simple example

The presented elementary ideas are demonstrated on the following example suggested by Liesen and Tichý. Consider the two-dimensional Poisson equation

$$-\Delta u = 32(\eta_1 - \eta_1^2 + \eta_2 - \eta_2^2) \tag{11}$$

on the unit square with zero Dirichlet boundary conditions. The exact solution is given by

$$u(\eta_1, \eta_2) = 16(\eta_1\eta_2 - \eta_1\eta_2^2 - \eta_1^2\eta_2 + \eta_1^2\eta_2^2)\,. \tag{12}$$

We discretise the problem using linear finite elements on a regular triangular grid with the meshsize h. Then the resulting linear algebraic system (1) is formed with approximation errors both in A and b of order h^2. The matrix A is symmetric and positive definite. For the approximate solution of (1) we apply the CG method, and stop the iteration whenever the normwise backward error drops below the level h^α, i.e. our stopping criterion is

$$\frac{\|b - Ax_n\|}{\|b\| + \|A\|\,\|x_n\|} < h^\alpha\,, \tag{13}$$

where $\alpha > 0$ is a parameter. Clearly, α should not be smaller than 2, otherwise the computational error can become significantly larger than the discretisation error. However, it should not be much larger than 2 either, since otherwise we will spend unnecessary work by enforcing the computational error to be much smaller than the discretisation error. The situation is illustrated on Fig. 3 and Fig. 4. The first one shows the discretisation error $u - x$ for $h = 1/101$ (x is approximated for our purpose sufficiently accurately by applying the standard MATLAB direct solver to $Ax = b$). Here x is a function of h, but we omit that in our notation. Fig. 4 shows the total error $u - x_n$ obtained by using the CG method with stopping criterion (13) for $\alpha = 3$. Clearly, both errors are of the same order of magnitude (10^{-4}). Increasing α would smooth $u - x_n$ closer to the form of $u - x$. The computed solution does not change significantly with increasing α, but the computational cost does.

This simple example shows the advantage of stopping iterations whenever discretisation and computational errors are in balance. Such balance is of course problem-dependent. In our experiment, for example, the gradient of the solution is for small α not well approximated; for getting a good gradient approximation the value of α would have to be much larger than for the simple approximation of the solution. It might also be desirable to evaluate the balance in a more sophisticated way. The principle, however, remains the same. The sophisticated stopping criteria can be considered a backward perturbation [3] and can be linked with other variational crimes (e.g. [74]). Our example also shows that, apart from some very special situations, a common textbook comparison of direct and iterative solvers which is based on the *same accuracy level* of the computed approximations makes little sense in practical computations. An iteration should always be accompanied by measuring the size of the error (in an appropriate way) and it should be stopped when a desired accuracy level is reached. In practical problems the sufficient accuracy level is frequently many orders of magnitude lower than the accuracy obtained with direct solvers.

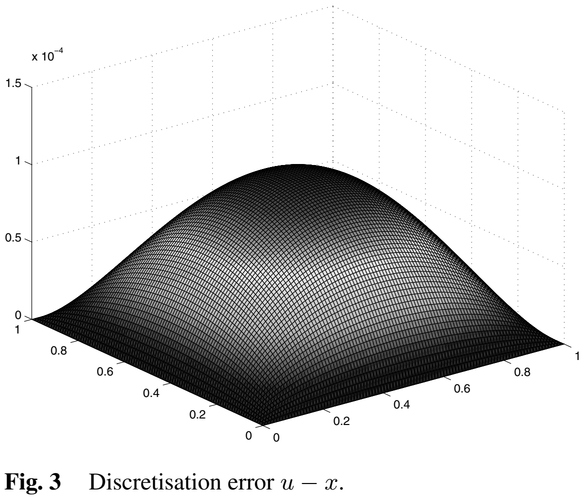

Fig. 3 Discretisation error $u - x$.

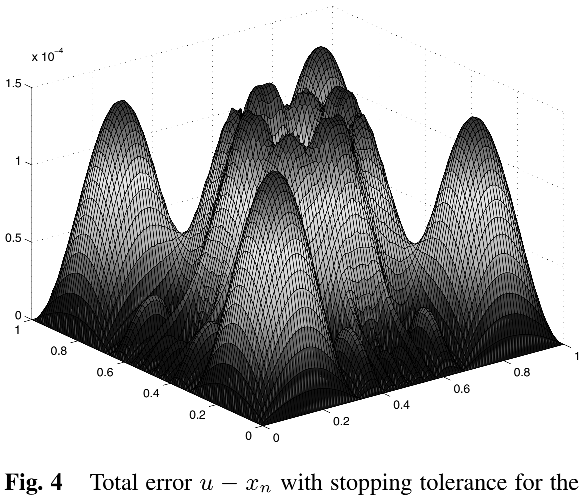

Fig. 4 Total error $u - x_n$ with stopping tolerance for the normwise backward error set to h^3.

2.3 Summary

Using the normwise backward error is numerically safe, because it can be computed with negligible additional rounding error (which can be easily bounded) and there are no hidden assumptions that may be violated in finite precision arithmetic. Moreover, the normwise backward error can be viewed as a practical application of the backward analysis idea which was present in the work of several founders of modern scientific computing such as Turing, and Goldstine and Von Neumann, and which was mathematically fully formalised and promoted (in the context of algebraic solvers) by Wilkinson, see the description in [61]. In the backward analysis we ask and answer the question as to how close the problem (8), which is solved *exactly* by x_n, is to the (original algebraic) problem (1), which is solved *approximately* by x_n. Perhaps this is our primary concern, given that the data A and b represent the original real-world problem inaccurately anyway. For numerical stability analysis the backward analysis in the sense of Wilkinson was a revolution – it allowed us to separate properties of a given problem (its sensitivity to perturbations) from the numerical stability properties of the methods, algorithms and their implementations [79, 80, 37].

In numerical stability analysis of iterative methods the backward analysis principle is developed further. For a fixed n the errors in iterations 1 through n are not only mapped backwards to perturbations of the original data, but the mapping can lead to "perturbed" problems of larger dimensionality which preserve some key information. The next section will recall results in this direction with a surprising consequence. In particular, it will demonstrate a general philosophical difficulty of mechanical forward error evaluation based on intermediate quantities.

3 Intermediate quantities and accuracy of final results

Assume that we wish to guarantee the accuracy of a final result computed by a given algorithm using given data. It may seem that if we wish to guarantee a prescribed number of digits in the final result, then we should compute all intermediate results with at least the same number of accurate digits. This suggestive view is, however, generally not correct. On the contrary, as formulated by Parlett [61, p. 22],

> *"... the number of significant digits in the intermediate numbers generated in a computation may be quite irrelevant to the accuracy of the output."*

This quote shows the importance and strength of numerical stability analysis, and backward stability analysis in particular. Though rounding errors on the roundoff unit level present in elementary computer operations can be considered "random", the way these tiny elementary rounding errors are spread through the computation is anything but random. Vital correlations between inaccurately computed quantities can lead to highly accurate final results. In order to understand the way elementary rounding errors affect the computed results we need a deep mathematical understanding of the algorithm, and to perform a thorough numerical stability analysis. That might be complicated, lengthy and full of unpleasant detailed bounds and formulas. Its goal is, however, to achieve understanding, which can usually be formulated in a very simple and elegant way.

3.1 Backward-like analysis of the Lanczos method

To be more specific, consider the Lanczos method [42] which is frequently used for computing dominant eigenvalues of Hermitian matrices and operators (generalisations of the Lanczos algorithm can also be used for solving non-Hermitian eigenproblems, but we will not consider that here). Given a Hermitian N by N matrix A and an initial vector q_1 of length N, the Lanczos method in exact arithmetic determines in the iterations 1 through n an N by n matrix Q_n with the first column q_1, such that

$$Q_n^* \, A \, Q_n = T_n \, , \quad \text{and} \quad Q_n^* Q_n = I_n \, , \tag{14}$$

where T_n is an n by n Hermitian tridiagonal matrix and I_n is the n by n identity matrix (the columns of Q_n are orthonormal). Eigenvalues of T_n are considered approximations of the (usually dominant) eigenvalues of A. We will present some more details about the Lanczos method in Sect. 4, for a thorough description and analysis see the book by Parlett [60] and the seminal paper of Paige [55].

In the presence of rounding errors, the computed analogy $\tilde{Q}_n$ of the exact Q_n does not have orthogonal columns. Even worse, the columns of $\tilde{Q}_n$ may quickly become numerically linearly dependent. Moreover, for the computed $\tilde{Q}_n$ and $\tilde{T}_n$,

$$\tilde{Q}_n^* \, A \, \tilde{Q}_n \neq \tilde{T}_n \, , \tag{15}$$

and most of the computed entries in $\tilde{T}_n$ may not exhibit a single digit of accuracy. They may differ from the analogous entries in T_n by orders of magnitude, which means that

$$\tilde{T}_n - T_n \quad \text{is large.} \tag{16}$$

Still, the backward error-like analysis of Greenbaum [28], see also [31], which is based on the results by Paige [55], shows that, and also why, (16) does not mean a total disaster. This analysis shows that there exist

- an M by M matrix $\overline{A}_n$, where $M \geq N$, possibly $M \gg N$, $\overline{A}_n$ having all its eigenvalues close to the eigenvalues of A;
- an M by n matrix $\overline{Q}_n$ having orthonormal columns such that

$$\overline{Q}_n^* \, \overline{A}_n \, \overline{Q}_n = \tilde{T}_n \, . \tag{17}$$

Consequently, the highly inaccurate *computed* matrix $\tilde{T}_n$ can be viewed as a result of the *exact precision* Lanczos algorithm applied to a different problem, possibly of much larger dimensionality, but preserving the very fundamental property of the original problem: All eigenvalues of $\overline{A}_n$ lie nearby the original eigenvalues of A. Since the exact arithmetic relations hold for $\overline{A}_n$, $\overline{Q}_n$ and $\tilde{T}_n$, the eigenvalues of the matrix $\tilde{T}_n$ can be used for approximating the eigenvalues of $\overline{A}_n$, and therefore the eigenvalues of A.

3.2 Summary

We have seen that in the application of the Lanczos method the number of correct digits in the computed elements of $\tilde{T}_n$ is *irrelevant* for the accuracy of the approximations to the eigenvalues of A. Hence, despite (16), the eigenvalues of A can be approximated to high accuracy using $\tilde{T}_n$. However, using $\tilde{T}_n$, the eigenvalues of A are not approximated in the same order and speed as they would be approximated with the exact T_n. Indeed, $\tilde{T}_n$ may produce multiple approximations of the original eigenvalues, with the multiplicities generated due to the rounding error amplifications in the application of the Lanczos method. Consequently, approximation of some other eigenvalues of A can in finite precision arithmetic be delayed (for details we refer to [30, 29, 70, 27, 71]). If we want to prevent these side effects, we must apply some correction procedure such as partial reorthogonalization [62]. That will not come without a significant cost in both computer time and memory. Numerical stability analysis tells us when it is reasonable to pay extra expenses, and when paying such expenses is nothing but an unreasonable waste of resources.

But how can we recognise that an eigenvalue of $\tilde{T}_n$ is close enough to some eigenvalue λ_i of the original matrix A ? This will be explained in the following section.

4 Examples of the mathematical theory of finite precision computations

Rounding errors in finite precision computations have another consequence: From the point of view of a formal definition of numerical algorithms, scientific computing lacks proper theoretical foundations. Some mathematicians feel that rounding errors prevent the existence of any elegant mathematical theory covering finite precision computations. Some theoretical computer scientists miss a formal model of computing with floating point numbers and, consequently, a complexity theory analogous to the complexity theory of combinatorial (and direct) algorithms. For a survey of related questions and an outline of the program for resolving them we refer to [10, 68]. An interesting related discussion can be found in [40].

The pessimistic conclusion is that there are practically *no results* linking complexity and numerical stability of numerical computations [13]. In scientific computing, however, the question of the cost of obtaining a satisfactory approximate solution of a given problem *for a given particular data set or class of data* is frequently more important than the question about complexity of the abstract problem which covers the *worst-case data*. In practical problems data rarely correspond to the worst case, and efficient algorithms typically take advantage of all specific information which can be found during the computation. If complexity is replaced by *computational cost*, then the pessimistic view does not apply. There are many results linking computational cost with numerical stability. For some important iterative methods (including the Lanczos method, CG and GMRES) there exist mathematical explanations of their behaviour in finite precision arithmetic. Before presenting some results, we recall the basic mathematical relationship between the Lanczos method, CG and Gauss quadrature. For proofs and detailed explanations we refer to [71] and to the original literature pointed out in that paper.

4.1 The Lanczos method, the CG method and Gauss quadrature

Given an N by N Hermitian matrix A and a starting vector q_1 of length N, the Lanczos method generates (in exact arithmetic, which is assumed throughout this subsection) a sequence of orthonormal vectors $q_1, q_2, \ldots$ via the following recurrence:

Given q_1, define $q_0 = 0$, $\beta_1 = 0$, and for $n = 1, 2, \ldots$, let

$$\begin{aligned} \alpha_n &= (Aq_n - \beta_n q_{n-1}, q_n) \, , \\ w_n &= Aq_n - \alpha_n q_n - \beta_n q_{n-1} \, , \end{aligned} \tag{18}$$

$$\begin{aligned}\beta_{n+1} &= \|w_n\|\,,\\ q_{n+1} &= w_n/\beta_{n+1}\,.\end{aligned}$$

Here $(\cdot,\cdot)$ denotes the Euclidean inner product. Denoting $Q_n = [q_1, \ldots, q_n]$, and

$$T_n = \begin{pmatrix} \alpha_1 & \beta_2 & & \\ \beta_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_n \\ & & \beta_n & \alpha_n \end{pmatrix}, \tag{19}$$

the recurrence (18) can be written in the matrix form

$$\boxed{\quad A \quad}\;\boxed{Q_n} \;=\; \boxed{Q_n}\;\boxed{T_n} \;+\; \boxed{O\,}\,,$$

where the last matrix on the right hand side is equal to $\beta_{n+1}q_{n+1}e_n^T$ (e_n denotes the nth column of the n by n identity matrix).

Assume that the matrix A is Hermitian positive definite. The standard implementation of the CG method was given in [36, (3:1a)–(3:1f)]:

Given x_0, define $r_0 = b - Ax_0$, $p_0 = r_0$, and for $n = 1, 2, \ldots$, let

$$\begin{aligned}\gamma_{n-1} &= (r_{n-1}, r_{n-1})/(p_{n-1}, Ap_{n-1}),\\ x_n &= x_{n-1} + \gamma_{n-1}\, p_{n-1},\\ r_n &= r_{n-1} - \gamma_{n-1}\, Ap_{n-1},\\ \delta_n &= (r_n, r_n)/(r_{n-1}, r_{n-1}),\\ p_n &= r_n + \delta_n\, p_{n-1}\,.\end{aligned} \tag{20}$$

The residual vectors $\{r_0, r_1, \ldots, r_{n-1}\}$ form an orthogonal basis and the direction vectors $\{p_0, p_1, \ldots, p_{n-1}\}$ form an A-orthogonal basis of the nth Krylov subspace $\mathcal{K}_n(A, r_0)$,

$$\mathcal{K}_n(A, r_0) \equiv \mathrm{span}\{r_0, Ar_0, \ldots A^{n-1}r_0\}\,. \tag{21}$$

The nth CG approximation x_n minimises the energy norm of the error over the affine subspace $x_0 + \mathcal{K}_n(A, r_0)$, i.e.,

$$\|x - x_n\|_A \;\equiv\; ((x - x_n), A(x - x_n))^{1/2} \;=\; \min_{z \in x_0 + \mathcal{K}_n(A, r_0)} \|x - z\|_A\,. \tag{22}$$

Consider $q_1 = r_0/\|r_0\|$. Then the link between the Lanczos and the CG methods can be explained in two lines: Using the change of variables

$$x_n = x_0 + Q_n\, y_n\,, \tag{23}$$

the coefficients y_n used to form the CG approximation x_n are determined by solving

$$T_n y_n = \|r_0\| e_1\,. \tag{24}$$

We now present what we consider the essence of both the Lanczos and the CG method. Denote the eigendecomposition of A by

$$\begin{aligned}A &= U\,\mathrm{diag}\,(\lambda_1, \ldots, \lambda_N)\,U^*\,, \quad && \lambda_1 \le \ldots \le \lambda_N\,,\\ U &= [u_1, \ldots, u_N]\,, && U^*\,U = U\,U^* = I_N\,,\end{aligned} \tag{25}$$

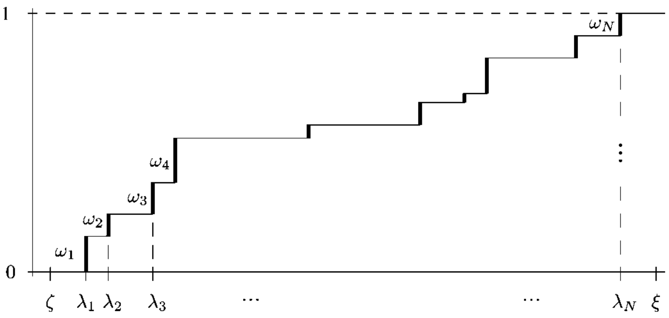

Fig. 5 Distribution function $\omega(\lambda)$.

and consider the squared size of the components of q_1 in the individual invariant eigenspaces of A,

$$\omega_i = |(q_1, u_i)|^2\,, \quad \sum_{i=1}^{N} \omega_i = 1\,. \tag{26}$$

Then A and q_1 determine the following distribution function $\omega(\lambda)$ with N points of increase at the eigenvalues of A,

$$\begin{aligned} \omega(\lambda) &= 0 && \text{for} \quad \lambda < \lambda_1\,, \\ \omega(\lambda) &= \textstyle\sum_{l=1}^{i} \omega_l && \text{for} \quad \lambda_i \le \lambda < \lambda_{i+1}\,, \\ \omega(\lambda) &= 1 && \text{for} \quad \lambda_N \le \lambda\,, \end{aligned} \tag{27}$$

see Fig. 5, and the corresponding Riemann-Stieltjes integral

$$\int_{\zeta}^{\xi} f(\lambda)\, d\omega(\lambda) \;=\; \sum_{i=1}^{N} \omega_i f(\lambda_i) \tag{28}$$

(for $\zeta \le \lambda_1$ and $\lambda_N \le \xi$).

The nth iteration of the CG method is determined by (23)–(24). The matrix T_n is symmetric positive definite, with the eigendecomposition

$$\begin{aligned} T_n &= S_n \operatorname{diag}(\theta_1^{(n)}, \dots, \theta_n^{(n)})\, S_n^*\,, && \theta_1^{(n)} \le \dots \le \theta_n^{(n)}\,, \\ S_n &= [s_1^{(n)}, \dots, s_n^{(n)}]\,, && S_n^* S_n = S_n S_n^* = I_n\,. \end{aligned} \tag{29}$$

Consider the squared size of the components of e_1 in the individual invariant eigenspaces of T_n (the squared size of the first entries of the eigenvectors of T_n),

$$\omega_j^{(n)} = |(e_1, s_j^{(n)})|^2\,, \quad \sum_{j=1}^{N} \omega_j^{(n)} = 1\,. \tag{30}$$

Then T_n and e_1 determine the distribution function $\omega^{(n)}(\lambda)$ with n points of increase at the eigenvalues of T_n,

$$\begin{aligned} \omega^{(n)}(\lambda) &= 0 && \text{for} \quad \lambda < \theta_1^{(n)}\,, \\ \omega^{(n)}(\lambda) &= \textstyle\sum_{l=1}^{j} \omega_l^{(n)} && \text{for} \quad \theta_j^{(n)} \le \lambda < \theta_{j+1}^{(n)}\,, \\ \omega^{(n)}(\lambda) &= 1 && \text{for} \quad \theta_n^{(n)} \le \lambda\,. \end{aligned}$$

Now comes the key point. The Riemann-Stieltjes integral determined by $\omega^{(n)}(\lambda)$,

$$\int_{\zeta}^{\xi} f(\lambda)\, d\omega^{(n)}(\lambda) \;=\; \sum_{j=1}^{n} \omega_j^{(n)} f(\theta_j^{(n)})\,, \tag{31}$$

(here $\zeta \le \theta_1^{(n)}$ and $\theta_n^{(n)} \le \xi$) is nothing but the n-point Gauss quadrature approximation of the original Riemann-Stieltjes integral (28) determined by $\omega(\lambda)$. Since $\omega(\lambda)$ contains all essential information about A and q_1 (apart from the change of variables represented by the eigenvectors), and, similarly, $\omega^{(n)}(\lambda)$ contains all essential information about T_n and e_1, we may conclude:

The Lanczos method and the CG method can be viewed as matrix formulations of the Gauss quadrature approximation of some underlying Riemann-Stieltjes integral.

This relationship is essentially known since the paper by Hestenes and Stiefel [36]. Despite intense efforts of Golub and his many collaborators, who promoted and used its various forms for decades, it has not been fully appreciated by the scientific computing community. Omitting details, whenever we use the algorithms (18) or (20), we in fact perform Gauss quadrature. This observation nicely illustrates deep links of modern numerical linear algebra to other disciplines, and shows the strongly nonlinear character of many of its problems.

4.2 Accuracy of eigenvalue approximations computed by the Lanczos method

In finite precision computations the computed quantities in the Lanczos method satisfy

$$AQ_n = Q_n T_n + \beta_{n+1} q_{n+1} e_n^T + F_n \tag{32}$$

where

$$\|F_n\| \leq n^{1/2} \|A\| \varepsilon_1 \,, \tag{33}$$

and ε_1 is proportional to the machine precision, see [50, 51, 52, 60]. *Here we skip any specific notation for computed quantities*. It may seem that since the matrix F_n, which accounts for effects of local rounding errors, is small in norm, nothing dramatic may happen. Just the opposite is true; the effects of rounding errors seem to be devastating. The computed Lanczos vectors $q_1, q_2, \dots, q_n$ can quickly loose not only their mutual orthogonality, but also their linear independence. However, Paige showed in his Ph.D. thesis in 1971 [50] that the mathematical elegance of the exact arithmetic theory can to a large extent be saved. He proved that loss of orthogonality follows a beautiful mathematical structure. We will present a consequence of his theory which demonstrates some of the difficulties which had to be handled, and also the beauty of the conclusions.

Assume, for simplicity, that the eigenvalues and eigenvectors of the computed T_n can be determined exactly. This assumption is not too restrictive, since they can indeed be computed very accurately. Given an eigenpair $\theta_j^{(n)}$, $s_j^{(n)}$ of T_n, the value of $\theta_j^{(n)}$ is considered an approximation to some eigenvalue of A, and $z_j^{(n)} = Q_n s_j^{(n)}$ an approximation of the corresponding eigenvector. The $\theta_j^{(n)}$ and $z_j^{(n)}$ are called the Ritz values and vectors. How can we determine whether $\theta_j^{(n)}$ and $z_j^{(n)}$ are indeed good approximations of an eigenvalue and eigenvector of A? We limit ourselves to the question about eigenvalues (a slightly more complicated case of eigenvectors can be found in [55, 60, 70]). A simple linear algebra exercise gives

$$\begin{aligned} \min_i |\lambda_i - \theta_j^{(n)}| &\leq \|Az_j^{(n)} - \theta_j^{(n)} z_j^{(n)}\| \,/\, \|z_j^{(n)}\| \\ &\leq (|e_n^T s_j^{(n)}| \beta_{n+1} + n^{1/2} \|A\| \varepsilon_1) \,/\, \|z_j^{(n)}\| \,. \end{aligned} \tag{34}$$

It seems that all is under control, since in exact arithmetic $\|z_j^{(n)}\| = 1$. If the norm of the computed vector $z_j^{(n)}$ is close to one, then, considering that $n^{1/2}\|A\|\varepsilon_1$ is a worst-case bound for some small quantity, the accuracy of $\theta_j^{(n)}$ is also computationally determined by the value

$$\delta_{nj} = |e_n^T s_j^{(n)}| \beta_{n+1} \,, \tag{35}$$

which can easily be determined from the bottom entry of the vector $s_j^{(n)}$. However, in finite precision computations the norm of $z_j^{(n)}$ cannot be guaranteed to be close to one. The vector $z_j^{(n)}$ is computed as a linear combination of the columns of Q_n, and since they can become numerically linearly dependent, $\|z_j^{(n)}\|$ can become very small. In order to justify (35) as an accuracy test in finite precision computations, we must resolve the difficulty represented by possibly vanishing $\|z_j^{(n)}\|$ in the denominator of (34). An ingenious analysis of Paige [55, pp. 241 & 249] lead to the following result: For any pair $\theta_j^{(n)}$, $z_j^{(n)}$ determined at the iteration n of a finite precision arithmetic Lanczos computation, it holds that

$$\min_i |\lambda_i - \theta_j^{(n)}| \leq \max \left\{ 2.5(\delta_{nj} + n^{1/2} \, \|A\| \, \varepsilon_1), \; [(n+1)^3 + \sqrt{3}\, n^2] \, \|A\| \, \varepsilon_2 \right\} , \tag{36}$$

$$|(z_j^{(n)}, q_{n+1})| = |\varepsilon_{jj}^{(n)}| / \delta_{nj} \,, \tag{37}$$

where $|\varepsilon_{jj}^{(n)}| \leq \|A\|\varepsilon_2$, and ε_1 and ε_2 are multiples of the machine precision.

Summarising, small δ_{nj} implies convergence of $\theta_j^{(n)}$ to some eigenvalue of A, and this holds in exact as well as in finite precision arithmetic. Moreover, the orthogonality of the newly computed Lanczos vector q_{n+1} can in a finite precision computation be lost only in the directions of the converged Ritz vectors.

This result is truly fascinating. It allows us to verify the accuracy of the results of finite precision Lanczos computations practically at no cost. But this is only possible as a consequence of the numerical stability theory developed by Paige. Without that, the computed δ_{nj} would give no guarantee whatsoever about the closeness of $\theta_j^{(n)}$ to some eigenvalue of the matrix A. For further discussion we refer to [71, pp. 69–70].

4.3 Estimating error norms in the CG method (20) for $Ax = b$

With $f(\lambda) \equiv \lambda^{-1}$ the relation between the integrals (28) and (31) can be described in the following way. The integral (28) becomes equal to $\|x - x_0\|_A^2/\|r_0\|^2$, and the value of its nth Gauss quadrature approximation (31) is the difference between this and the error in the nth CG iteration measured by $\|x - x_n\|_A^2/\|r_0\|^2$,

$$\frac{\|x - x_0\|_A^2}{\|r_0\|^2} \;=\; n\text{-point Gauss quadrature} \;+\; \frac{\|x - x_n\|_A^2}{\|r_0\|^2}\,. \tag{38}$$

This relation was developed in [14] in the context of moments. It was a subject of extensive work motivated by estimation of the error norms in CG in the papers [22, 25] and [27]. Work in this direction continued and led to the papers [26, 47, 12].

Based on the idea from [27, pp. 28–29], we can eliminate the unknown term $\|x-x_0\|_A^2/\|r_0\|^2$ by subtracting the identities for iterations n and $n + d$, where d is a positive integer. Then, multiplying by $\|r_0\|^2$,

$$\|x - x_n\|_A^2 \;=\; \mathrm{EST}^2 + \|x - x_{n+d}\|_A^2 \tag{39}$$

where

$$\mathrm{EST}^2 = \|r_0\|^2\,[(n + d)\text{-point Gauss quadrature} - n\text{-point Gauss quadrature}]\,. \tag{40}$$

The energy norm of the error in the CG method is strictly decreasing. When $\|x - x_n\|_A^2 \gg \|x - x_{n+d}\|_A^2$, EST gives a tight lower bound for $\|x - x_n\|_A$.

The value of EST^2 can be determined in different ways. In [26] it has been proposed to find it as a difference between two continued fractions (without computing the fractions themselves; that approach was improperly used in [27]). Another possibility is to evaluate

$$\mathrm{EST}^2 \;=\; r_0^T(x_{n+d} - x_n)\,, \tag{41}$$

see [78]. The value of EST^2 can also be derived without using Gauss quadrature as a direct consequence of [36, Theorem 6.1]

$$\mathrm{EST}^2 = \sum_{i=n}^{n+d-1} \gamma_i \|r_i\|^2\,, \tag{42}$$

where both γ_i and $\|r_i\|^2$ are available directly from the conjugate gradient algorithm, see (20).

In exact arithmetic all formulas for EST^2 lead to identical results. In finite precision arithmetic, however, they can differ substantially. What is their relevance in finite precision computations? This question cannot be answered without a thorough numerical stability analysis. As in Sect. 4.2, the goal of such an analysis is very practical. We need to justify the estimates for the energy norm of the error that should replace or complement the existing convergence measures.

This numerical stability question about the estimates for the energy norm of the error in the CG method was first posed in [27]. It was also partially answered in that paper for the estimates using continued fractions. A detailed analysis followed in [71], which proved that the estimate (42) is numerically stable and it can be used in finite precision arithmetic computations while the estimate (41) is, in general, numerically *unstable*. Interested readers are referred also to the recent manuscript [73], which is less technical, but which offers, in addition, an easy introduction to estimating norms of the errors in the preconditioned CG method.

We next illustrate our results by an example. As in [73, 72] we use matrices from the collection Cylshell by R. Kouhia (http://www.hut.fi/˜kouhia/) that is available from the Matrix Market (http://math.nist.gov/MatrixMarket/) library of test problems. Matrices in the Cylshell collection correspond to low order finite element discretisations of cylindrical shell elements, loaded in such a way that only the last element of the right hand side b is nonzero. These matrices exhibit large

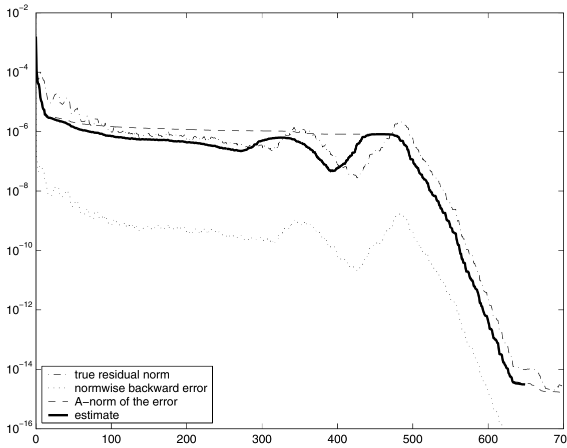

Fig. 6 Convergence characteristics and the lower bound for the energy norm of the error computed using (42) when the preconditioned CG method is applied to a system from the Cylshell collection, $d = 50$.

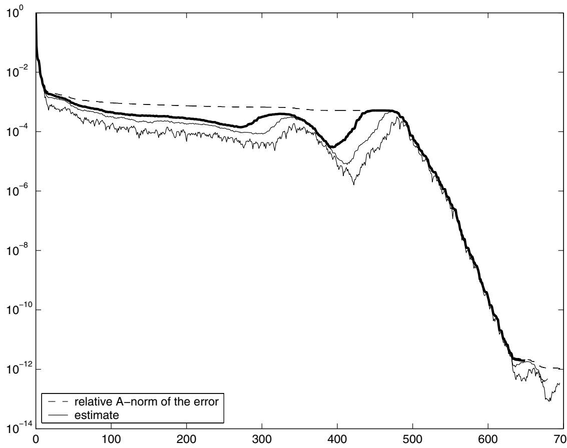

Fig. 7 Influence of the parameter d on the tightness of the bound (42). The tightness of the lower bound improves with increasing d (here $d = 4$, 20, and 50).

condition numbers and the algebraic problems are very difficult to precondition using standard techniques such as incomplete Cholesky decompositions. We use matrices from this collection repeatedly because they allow us to demonstrate nice features of the bounds presented above, but they also reveal possible difficulties with their application. We used the matrix $s3rmt3m3$ with $N = 5357$, containing 207123 nonzero elements. The experiments were performed in MATLAB 6.5 on a PC with machine precision 10^{-16} using MATLAB 6.5. We used the preconditioned CG method as implemented in MATLAB with MATLAB's incomplete Cholesky preconditioner (threshold $= 10^{-5}$).

Fig. 6 shows the value of EST computed using (42) (bold solid line) for $d = 50$ together with the values of the energy norm of the error $\|x - x_n\|_A$ (dashed line), the residual norm $\|b - Ax_n\|$ (dash-dotted line) and the normwise backward error $\|b - Ax_n\|/(\|b\| + \|A\|\,\|x_n\|)$ (dotted line). We see that if the value $\|x - x_n\|_A$ decreases rapidly with n, then the lower bound (42) is very tight. When the decrease of $\|x - x_n\|_A$ is slow, the bound might not be monotonic and it can also significantly differ from the actual value. This is definitely a drawback which has to be considered (the bound should be used with other convergence measures). One should also note the behaviour of the residual norm and the normwise backward error. They both are significantly non-monotonic.

Fig. 7 shows besides the relative energy norm of the error, $\|x - x_n\|_A/\|x - x_0\|_A$ (dashed line) its estimates obtained using (42) for different values of the parameter d (here $d = 4$, 20, and 50). We can see that larger d improves the quality of the bound. Apart from the rather small value $d = 4$, the differences are not dramatic.

Fig. 8 shows in addition to $\|x - x_n\|_A/\|x - x_0\|_A$ (dashed line) and its estimate obtained using (42) with $d = 50$ (bold solid line) also its estimate obtained using (41) with $d = 50$ (solid line). We can observe that (41) gives for $n \geq 500$ quite misleading information. Though the formula (41) can be evaluated with a high accuracy proportional to the machine precision, it should not be applied in finite precision computations. It has been derived using the strong assumption about

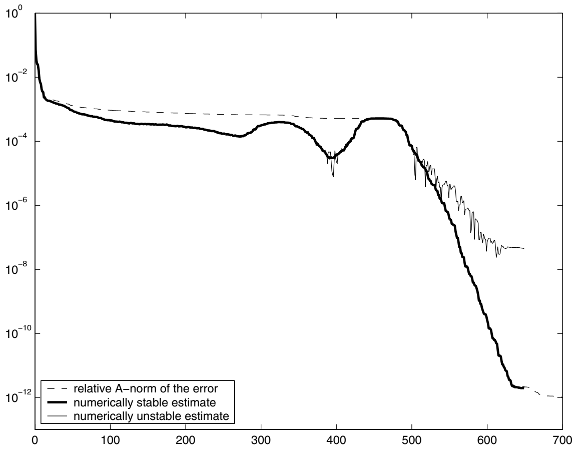

Fig. 8 Stable and unstable *lower* bounds for the energy norm of the error. The numerically unstable bound (41) can in finite precision arithmetic give values that significantly *overestimate* the true relative energy norm of the error.

preserving global orthogonality among the residual vectors in the CG method. Once this assumption is violated by using finite precision arithmetic, (41) is completely disqualified for estimating the energy norm of the CG error.

We can point out again the importance of numerical stability analysis. It tells us that a given stopping criterion derived under some particular assumptions can with no restrictions be used in finite precision computations, while the use of some other (equivalent in exact arithmetic) stopping criterion derived under different assumptions can lead to computational disasters.

4.4 Loss of orthogonality and convergence behaviour in the GMRES method

In Sect. 1.3 we have already mentioned the GMRES method proposed by Saad and Schultz [67]. In this section we present some results concerning the numerical behaviour of this important method.

GMRES is widely used for solving unsymmetric linear algebraic systems arising from the discretisation of partial differential equations. In iteration n, the method minimises the Euclidean norm of the residual $r_n = b - Ax_n$ over x_n in the affine space $x_0 + \mathcal{K}_n(A, r_0)$. Theoretical results about the GMRES residual norms therefore provide lower bounds for the residual norms of other methods that use the same Krylov subspaces. Several mathematically equivalent implementations have been proposed in the literature. These may differ, however, in finite precision arithmetic. It is therfore essential to identify the optimal ones which should be used in practical computations. In addition to that, a strong relationship between convergence of GMRES and loss of orthogonality among the computed basis vectors of the Krylov subspaces has been noticed in some GMRES implementations. It is important to find a theoretical explanation for this phenomenon.

In exact arithmetic, GMRES can be described as follows. It starts with an initial approximation x_0, computes the initial residual $r_0 = b - Ax_0$, and then determines a sequence of approximate solutions $x_1, \ldots, x_n$ such that $x_n \in x_0 + \mathcal{K}_n(A, r_0)$, and hence $r_n \in r_0 + A\mathcal{K}_n(A, r_0)$. The choice of x_n is based on the *minimal residual principle*

$$\|r_n\| \;=\; \min_{z \in x_0 + \mathcal{K}_n(A, r_0)} \|b - Az\| \,, \tag{43}$$

which can be equivalently formulated as the *orthogonal projection principle*

$$r_n \perp A\mathcal{K}_n(A, r_0) \,. \tag{44}$$

For a nonsingular matrix A, both (43) and (44) determine the unique sequence of approximate solutions $x_1, \ldots, x_n$, see [67].

Now let $v_1 \equiv r_0/\|r_0\|$, $w_1 \equiv Av_1/\|Av_1\|$, and consider two sequences of orthonormal vectors, $v_1, v_2, \ldots$ and $w_1, w_2, \ldots$, such that for each n,

$$\mathcal{K}_n(A, r_0) = \operatorname{span}\{v_1, \ldots, v_n\} \,, \quad V_n \equiv [v_1, \ldots, v_n] \,, \quad V_n^* V_n = I_n \,, \tag{45}$$

$$A\mathcal{K}_n(A, r_0) = \operatorname{span}\{w_1, \ldots, w_n\} \,, \quad W_n \equiv [w_1, \ldots, w_n] \,, \quad W_n^* W_n = I_n \,. \tag{46}$$

Then the minimal residual principle (43) can be formulated as

$$\|r_n\| \;=\; \min_y \|r_0 - AV_n y\| \tag{47}$$

$$\;=\; \min_t \|r_0 - W_n t\| \,. \tag{48}$$

The residual r_n is therefore the least squares residual for the least squares problems $AV_n y \approx \|r_0\|\, v_1$ and $W_n t \approx \|r_0\|\, v_1$.

We recall two main approaches which explicitly compute the basis vectors $v_1, v_2, \ldots, v_n$, respectively $v_1, w_1, \ldots, w_{n-1}$, defined in (45) and (46). In the first approach, the approximate solution x_n is expressed as

$$x_n \;=\; x_0 \;+\; V_n\, y_n \,, \tag{49}$$

which leads to the classical GMRES method of Saad and Schultz [67]. In the second approach the approximate solution is expressed as

$$x_n \;=\; x_0 \;+\; [v_1, W_{n-1}]\, t_n \tag{50}$$

for some t_n. Its implementation is more straightforward than the one based on (49), and hence it was called "simpler GMRES" [77]. On the other hand, the approximate solution is in this approach determined via the basis vectors $v_1, w_1, \ldots, w_{n-1}$, which are *not mutually orthogonal* (v_1 is in general not orthogonal to $w_1, \ldots, w_{n-1}$; here we mean the exact arithmetic relationship, not a deterioration of orthogonality due to rounding errors). This fact raises some suspicions concerning potential numerical problems of this approach. These problems will be studied in the following subsection.

For completeness, we mention that a variety of methods based on either (43) or (44) have been proposed that neither explicitly compute the vectors $v_1, v_2, \ldots, v_n$, nor the vectors $v_1, w_1, \ldots, w_{n-1}$. For example, the method by Khabaza [41] uses the vectors $r_0, Ar_0, \dots, A^{n-1}r_0$; Orthomin [76], Orthodir [83], Generalised Conjugate Gradient (GCG) [5, 6] and Generalised Conjugate Residual (GCR) [17, 18] compute an $A^T A$-orthogonal basis of $\mathcal{K}_n(A, r_0)$. These methods played an important role in the development of the field and they could be useful in some applications. They are, however, numerically less stable than the classical implementation of GMRES. For further details see [43, 65].

4.4.1 Simpler GMRES is potentially unstable

We are going to explain, while omitting details which can be found in [43], that the GMRES implementation based on (50) is potentially numerically unstable. The key argument is given by the following identity for the relative residual norm (see [43, relations (3.5) and (3.6)]),

$$\frac{\|r_n\|}{\|r_0\|} \;=\; \sigma_{\min}([v_1,\, W_n])\, \sigma_1([v_1,\, W_n]) \;=\; \frac{2\,\kappa([v_1,\, W_n])}{\kappa([v_1,\, W_n])^2 + 1}\,, \tag{51}$$

where $\sigma_{\min}(\cdot)$ denotes the minimal singular value and $\sigma_1(\cdot)$ the maximal singular value of the given matrix, and $\kappa(\cdot) \equiv \sigma_1(\cdot)/\sigma_{\min}(\cdot)$ the corresponding condition number. Identity (51) shows that the conditioning of the basis $[v_1,\, W_n]$ of the Krylov subspace $\mathcal{K}_{n+1}(A, r_0)$ is fully determined (except for an unimportant multiplicative factor) by the size of $\|r_n\|/\|r_0\|$, and vice versa. In particular,

$$\kappa([v_1, W_n])^{-1} \;\le\; \frac{\|r_n\|}{\|r_0\|} \;\le\; 2\,\kappa([v_1, W_n])^{-1}\,, \tag{52}$$

so that the relative residual norm is small if and only if $[v_1,\, W_n]$ is ill-conditioned.

How does this affect the numerical stability of simpler GMRES? The basis W_n is computed by a recursive columnwise QR-factorisation of the matrix $[Av_1, AW_{n-1}]$, i.e.

$$A[v_1, W_{n-1}] = [Av_1, AW_{n-1}] = W_n G_n\,, \tag{53}$$

where G_n is the n-by-n upper triangular factor in the QR-factorisation. Using (50), the vector t_n solves the least squares problem

$$\|r_n\| \;=\; \min_t \|r_0 - A[v_1, W_{n-1}]\, t\| \;=\; \min_t \|r_0 - W_n G_n\, t\|\,. \tag{54}$$

Now suppose, for clarity, that W_n is computed in the numerically most stable way, and that the orthogonality among its columns is in finite precision computations preserved up to a small multiple of the machine precision ε. Then (51)–(52) hold, up to a small multiple of the machine precision, also for the quantities computed using finite precision arithmetic. Hence the residual norm and the conditioning of the matrix $[v_1, W_{n-1}]$ are in finite precision computations strongly related. A decrease of $\|r_n\|$ necessarily leads to ill-conditioning of the computed $[v_1, W_{n-1}]$. But if $[v_1, W_{n-1}]$, and hence $A[v_1, W_{n-1}]$, is ill-conditioned, which *must happen* for $\|r_n\|$ getting small, then the computed G_n will also be ill-conditioned. This can result in a *large error* in the computed t_n. We stress that the principal source of this error is not connected to the conditioning of A. Hence simpler GMRES is potentially unstable even for very well-conditioned matrices A. Because of the different choice of the basis ((45) instead of (46)) this numerical trouble cannot occur in classical GMRES.

Summarising, minimal residual Krylov subspace methods can be formulated and implemented using different bases and different orthogonalisation processes. This section shows that using different bases is important in getting revealing theoretical results about convergence of the method, and a correct choice of basis is fundamental for getting numerically stable implementations. We have explained that using the best orthogonalisation technique in building the basis does not compensate for a possible loss of accuracy in the given implementation which is caused by a poor choice of the basis.

4.4.2 Loss of orthogonality and convergence in modified Gram-Schmidt GMRES

In the rest of this paper we will focus on the classical GMRES formulation based on (47) and (49), and we will study numerical stability of various implementations based on different orthogonalisation processes for building up the matrix V_n in (45). When V_n is computed using Householder reflections, then the rounding error analysis of the QR-factorisation developed by Wilkinson [80, pp. 152–161, 236, and 382–388] proves that (unless A is close to numerically singular) the loss of orthogonality among $v_1, \dots, v_n$ is proportional to the machine precision ε [15, relation (2.4)]. With approximately orthonormal V_n the idea behind the rounding error analysis of the whole algorithm is straightforward. Replacing computed V_n by a proper nearby matrix with exactly orthonormal columns (see [15, Lemma 3.3]) proves that in the Householder reflections-based implementations of GMRES, the backward error at the final step is proportional to the machine precision ε [15, Corollary 4.2].

Consequently, in the Householder reflections based GMRES the ultimate backward error and residual norms are essentially the same as those guaranteed by direct solving of the system $Ax = b$ via the Householder or Givens QR-factorisations.

Preserving orthogonality of the columns in the computed V_n close to ε is costly. The commonly used GMRES implementations use the modified Gram-Schmidt (MGS) orthogonalisation for computing V_n, which turns out to be much cheaper than the Householder reflections based GMRES. However, the orthogonality among the vectors $v_1, v_2, \ldots, v_n$ is typically gradually lost, which eventually leads to a loss of linear independence. Consequently, modified Gram-Schmidt GMRES (MGS GMRES) can not be analysed using the approach from [15], where everything relied upon the fact that V_n has almost orthonormal columns. How much is lost in terms of convergence and the ultimate attainable accuracy? This question is answered next.

When the MGS orthogonalisation is used, the computed vectors $v_1, \ldots, v_n$ depend on the (ill-)conditioning of the matrix $[r_0, AV_n]$. More specifically, the loss of orthogonality among the computed basis vectors is bounded by

$$\|I - V_{n+1}^* V_{n+1}\|_F \;\le\; \kappa([r_0\gamma, AV_nD_n])\, O(\varepsilon), \tag{55}$$

for all $\gamma > 0$ and positive diagonal n by n matrices D_n, here $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. One possibility is to scale the columns of $[r_0\gamma, AV_nD_n]$ so they have unit length. That is, take

$$\gamma = \|r_0\|^{-1}, \quad D_n = \mathrm{diag}\,(\|Av_1\|^{-1}, \ldots, \|Av_n\|^{-1})\,. \tag{56}$$

The corresponding condition number and the bound (55) would then be no more than a factor $\sqrt{n+1}$ away from its minimum, see [75], so this is a nearly optimal scaling. Other convenient choices are discussed in [59]. Extensive experimental evidence suggests that for the nearly optimal scaling (56), the bound (55) is tight, and usually

$$\|I - V_{n+1}^* V_{n+1}\|_F \;\approx\; \kappa([r_0\gamma, AV_nD_n])\, O(\varepsilon)\,. \tag{57}$$

It was observed that when MGS was used, leading to MGS GMRES, the loss of orthogonality in V_{n+1} was accompanied by a decreasing relative residual norm $\|r_n\|/\|r_0\|$, see [32] and also [66]. That is, significant loss of orthogonality in MGS GMRES apparently did not occur before convergence measured by $\|r_n\|/\|r_0\|$ occurred. This behaviour was analysed numerically in [32, 64] and a partial quantitative explanation which corresponded to our intuition was offered there. GMRES approximates r_0 by the columns of AV_n, therefore the condition number of $[r_0, AV_n]$ has to be related to the GMRES convergence. A stronger and more complete theoretical explanation of the observed behaviour is derived in [57, 58, 59, 56].

We will now describe the main observation in detail. Consider a plot with two lines obtained from a MGS GMRES finite precision computation. One line represents the normwise relative backward error $\|r_n\|/(\|b\| + \|A\|\,\|x_n\|)$ and the other the loss of orthogonality $\|I - V_{n+1}^* V_{n+1}\|_F$ (both plotted using the same logarithmic scale) as a function of the iteration step n. We have observed that these two lines are always almost reflections of each other through the horizontal line defined by their intersection. For a clear example of this, see the dashed lines in Fig. 9. In other words, in finite precision MGS GMRES computations, the product of the normwise relative backward error and the loss of orthogonality (as a function of the iteration step) *is almost constant* and of the order of the machine precision. Orthogonality among the computed MGS basis vectors is effectively maintained until convergence of the normwise relative backward error (and also the relative residual norm) to the maximal attainable accuracy. Total loss of orthogonality among the computed basis vectors implies convergence of the normwise relative backward error to $O(\varepsilon)$, which is equivalent to the (normwise) *backward stability of MGS GMRES*.

Using the results of [57, 58], the main ideas of the proof are simple and elegant. In terms of formulas, we wish to prove that for the quantities computed in a finite precision arithmetic application of MGS GMRES it holds

$$\frac{\|r_n\|}{\|b\| + \|A\|\,\|x_n\|} \cdot \|I - V_{n+1}^* V_{n+1}\|_F \;=\; O(\varepsilon)\,. \tag{58}$$

A first step, which we have already discussed, consists of a formal proof of the tight relation (57) for the loss of orthogonality (for details see [59]). Using (57), the identity (58) is reduced to

$$\frac{\|r_n\|}{\|b\| + \|A\|\,\|x_n\|} \cdot \kappa\,([r_0\gamma, AV_nD_n]) \;=\; O(1)\,. \tag{59}$$

Our efforts in proving the last identity have led to solving fundamental and very difficult problems in the seemingly very loosely related area of scaled total least squares, see [57, 58].

The proof itself (as yet in some details incomplete) is, however, technical and tedious. Therefore in [59] we restrict ourselves to proving and discussing exact arithmetic results about the product of the normwise relative backward error $\|r_n\|/(\|b\| + \|A\|\,\|x_n\|)$ and the condition number $\kappa([r_0\gamma, AV_nD_n])$. A detailed rounding error analysis, together with the results relating the genuine loss of orthogonality $\|I - V_{n+1}^* V_{n+1}\|_F$ to the relative backward error, is still in progress.

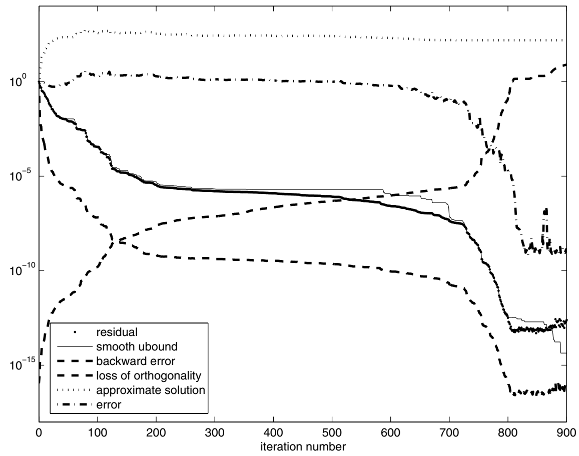

Fig. 9 Convergence characteristics of MGS GMRES applied to SHERMAN2 with b from Matrix Market and $x_0 = 0$.

For illustration of the results mentioned here we include an example for the matrix SHERMAN2 from the Matrix Market collection. In Fig. 9, dots denote the norm of the directly computed relative residual ($\|b - Ax_n\|/\|r_0\|$), the dashed-dotted line the relative error ($\|x - x_n\|/\|x - x_0\|$; x was determined by the MATLAB direct solver), the mostly decreasing dashed line the normwise relative backward error ($\|b - Ax_n\|/(\|b\| + \|A\| \; \|x_n\|)$), the monotonically increasing dashed line the loss of orthogonality among the Arnoldi vectors measured in the Frobenius norm ($\|I - V_n^* V_n\|_F$), the dotted line norm of the approximate solution ($\|x_n\|$) and the solid line the smooth upper bound for the norm of the relative residual which is used in the paper [59, relation (3.9)]. For the experiment we have used the right hand side given by Matrix Market (representing discretised conditions of the real-world problem) and $x_0 = 0$. We see that convergence to maximal attainable accuracy measured by all characteristics occurs in about 800 steps. One should also note the close symmetry of the dashed lines, illustrating the results formulated above.

The smoothed upper bound (solid line) is sometimes very close to the dots, but sometimes the difference is noticeable. We cannot go into details here, but we sketch the main difficulty we have to deal with. The tightness of the bound is determined by the distance of the ratio $\delta_n \equiv \sigma_{min}([r_0\gamma, AV_nD_n])/\sigma_{min}([AV_nD_n])$ to one. In order to analyse the tightness of the bound for the norm of the relative residual, we must therefore first describe the necessary and sufficient condition for preserving the smallest singular value of a matrix while appending (or deleting) a column. This condition represents a subtle matrix theory result. Then we have to study whether this condition is satisfied in MGS GMRES computations. That leads into a quantitative formulation of the fact that although δ_n can become under some circumstances very close or even equal to one, such situation *cannot* occur after MGS GMRES has converged to some particular accuracy (cf. the iteration steps between 700 and 800 in Fig. 9 where the smooth upper bound is very tight). Summarizing, the case δ_n close to one does not represent a serious obstacle for the theory, but it makes the whole theoretical explanation of the observed facts very subtle and difficult, see [57, 58, 59].

5 Concluding remarks

Summarising, modern numerical linear algebra, which aims at solving linear algebraic problems, often exhibits strongly nonlinear properties. This is true both in exact and in finite precision arithmetic.

We have recalled the backward error principle and have illustrated the power and the beauty of backward error analysis on several examples of different nature. Among other consequences, it turns out that highly accurate final results can be achieved despite inaccurately computed intermediate quantities. Although in finite precision arithmetic some basic axioms do not hold, a theory linking the cost of numerical computations with the accuracy of the computed results can be built. This can be regarded as a mathematical theory of finite precision computation in solving particular problems or classes of problems using particular methods. This theory also shows that the exact and finite precision arithmetic parts of problems in numerical linear algebra are deeply interconnected.

Throughout this paper we have presented examples showing that analysis of methods in numerical linear algebra and of their computational behaviour can be tedious and difficult, with intermediate steps full of complicated estimates and formulas. The resulting understanding is, however, often formulated as an elegant mathematical conclusion easily described in a common language. As an example, in the Lanczos method for computing eigenvalues of Hermitian matrices and in the MGS implementation of the GMRES method, such conclusions read: *Loss of orthogonality means convergence*. Analysis leading to such deep understanding is based on unexpected and revealing links between different areas of mathematics far beyond the borders of numerical linear algebra.

Acknowledgements The authors are indebted to Petr Tichý for his help with numerical experiments, and to Anne Greenbaum, Chris Paige, and Volker Mehrmann for useful comments which improved the manuscript. The work of the first author was supported by the Czech National Program of Research Information Society under the project 1ET400300415. The work of the second author was supported by the Emmy Noether - Program of the Deutsche Forschungsgemeinschaft.

References

[1] M. Arioli, I. S. Duff, and D. Ruiz, SIAM J. Matrix Anal. Appl. **10**, 138–144 (1992).

[2] M. Arioli, Numer. Math. **97**, 1–24 (2004).

[3] M. Arioli, D. Loghin, and A. J. Wathen, Numer. Math. **99**, 381–410 (2005).

[4] M. Arioli, E. Noulard, and A. Russo, Calcolo **38**, 97–112 (2001).

[5] O. Axelsson, Linear Algebra Appl. **29**, 1–16 (1980).

[6] O. Axelsson, Numer. Math. **51**, 209–227 (1987).

[7] I. Babuška, Mathematics of the verification and validation in computational engineering, Proceedings of the Conference Mathematical and Computer Modelling in Science and Engineering, edited by M. Kočandrlová and V. Kelar, Union of Czech Mathematicians and Physicists, Prague (2003), pp. 5–12.

[8] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozzo, C. Romine, and H. A. van der Vorst, Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods (SIAM, Philadelphia, 1995).

[9] B. Beckermann and A. Kuijlaars, ETNA **14**, 1–19 (2002).

[10] L. Blum, F. Cucker, M. Shub, and S. Smale, Complexity and Real Computation (Springer-Verlag, New York, 1999).

[11] A. N. Brooks and T. J. R. Hughes, Comput. Methods Appl. Mech. Eng. **32**, 199–259 (1982).

[12] D. Calvetti, S. Morigi, L. Reichel, and F. Sgallari, Numer. Algorithms **25**, 79–88 (2000).

[13] F. Cucker, Real Computations with Fake Numbers, Lecture Notes in Computer Science 1644 (Springer Verlag, Berlin, 1999), pp. 55–73.

[14] G. Dahlquist, G. H. Golub, and S. G. Nash, Bounds for the error in linear systems, in: Proceedings of the Workshop on Semi-Infinite Programming, edited by R. Hettich (Springer Verlag, Berlin, 1978), pp. 154–172.

[15] J. Drkošová, A. Greenbaum, M. Rozložník, and Z. Strakoš, BIT **35**, 309–330 (1995).

[16] M. Eiermann, Semiiterative Verfahren für nichtsymmetrische lineare Gleichungssysteme, Habilitationsschrift, Universität Karlsruhe (1989).

[17] S. C. Eisenstat, H. C. Elman, and M. H. Schultz, SIAM J. Numer. Anal. **20**, 345–357 (1983).

[18] H. C. Elman, Iterative methods for large sparse, nonsymmetric systems of linear equations, Ph.D. Thesis, Yale University (1982).

[19] H. C. Elman and A. Ramage, Math. Comput. **72**, 263–288 (2001).

[20] H. C. Elman and A. Ramage, SIAM J. Numer. Anal. **40**, 254–281 (2002).

[21] O. G. Ernst, SIAM J. Matrix Anal. Appl. **21**, 1079–1101 (2000).<

[22] B. Fischer and G. H. Golub, On the error computation for polynomial based iteration methods, in: Recent Advances in Iterative Methods, edited by G. H. Golub, A. Greenbaum, and M. Luskin (Springer-Verlag, New York, 1994), pp. 59–67.

[23] B. Fischer, A. Ramage, D. Silvester, and A. J. Wathen, Comput. Methods Appl. Mech. Eng. **179**, 179–195 (1999).

[24] V. Fraysé, L. Giraud, S. Gratton, and J. Langou, A set of GMRES routines for real and complex arithmetic on high performance computers, CERFACS Technical Report TR/PA/03/3 (2003).

[25] G. H. Golub and G. Meurant, Matrices, moments and quadrature, in: Numerical Analysis 1993, Pitman Research Notes in Mathematics Series 303, edited by D. Griffiths and G. Watson (Longman Sci. Tech. Publ., Harlow, 1994), pp. 105–156.

[26] G. H. Golub and G. Meurant, BIT **37**, 687–705 (1997).

[27] G. H. Golub and Z. Strakoš, Numer. Algorithms **8**, 241–268 (1994).

[28] A. Greenbaum, Linear Algebra Appl. **113**, 7–63 (1989).

[29] A. Greenbaum, The Lanczos and conjugate gradient algorithms in finite precision arithmetic, in: Proceedings of the Cornelius Lanczos Centennary Conference (SIAM, Philadephia, 1994), pp. 49–60.

[30] A. Greenbaum, Iterative methods for solving linear systems, Frontiers in Applied Mathematics (SIAM, Philadelphia, PA. 1997).

[31] A. Greenbaum and Z. Strakoš, SIAM J. Matrix Anal. Appl. **13**, 121–137 (1992).

[32] A. Greenbaum, M. Rozložník, and Z. Strakoš, BIT **37**, 706–719 (1997).

[33] A. Greenbaum, J. Duinter Tebbens, and Z. Strakoš, GMRS and ideal GMRS for a convection–diffusion matrices model problem, in preparation.

[34] W. Hackbusch, Iterative Solution of Large Sparse Systems of Equations, Applied Mathematical Sciences 95 (Springer-Verlag, New York, 1994); translated and revised from the 1991 German original.

[35] C. Hegedüs, Private communication (1998).

[36] M. R. Hestenes and E. Stiefel, J. Res. Natl. Bur. Stand. (USA) **49**, 409–436 (1952).

[37] N. J. Higham, Accuracy and stability of numerical algorithms (SIAM Publications, Philadelphia, 1996).

[38] T. J. R. Hughes and A. Brooks, A multidimensional upwind scheme with no crosswind diffusion, in: Finite Element Methods for Convection Dominated Flows (Papers, Winter Ann. Meeting Amer. Soc. Mech. Engs., New York, 1979), Vol. 34 of AMD (Amer. Soc. Mech. Eng. (ASME), New York, 1979), pp. 19–35.

[39] I. C. Ipsen, BIT **40**, 524–536 (2000).

[40] A. Iserles, SIAM Rev. **42**, 739–745 (2000).

[41] I. M. Khabaza, Comput. J. **6**, 202–206 (1963/1964).

[42] C. Lanczos, J. Res. Natl. Bur. Stand. (USA) **45**, 255–282 (1950).

[43] J. Liesen, M. Rozložník and Z. Strakoš, SIAM J. Sci. Comput. **23**, 1503–1525 (2002).

[44] J. Liesen and Z. Strakoš, PAMM **3**, 551–552 (2003).

[45] J. Liesen and Z. Strakoš, submitted to SIAM J. Sci. Comput. (2004).

[46] J. Liesen and P. Tichý, Behavior of Krylov subspace methods for symmetric tridiagonal Toeplitz matrices, Preprint 34-2004, Institute of Mathematics, Technical University of Berlin, 2004.

[47] G. Meurant, Numer. Algorithms **16**, 77–87 (1997).

[48] K.W. Morton, Numerical Solution of Convection-Diffusion Problems, Chapman & Hall, London, (1996).

[49] J.T. Oden, J.C. Browne, I. Babuška, K.M. Liechti, and L.F. Demkowicz, A Computational Infrastructure for Reliable Computer Simulations, Lecture Notes in Computer Science 2660 (Springer Verlag, Heidelberg, 2003), pp. 385–392.

[50] C.C. Paige, The compution of eigenvalues and eigenvectors of very large sparse matrices, Ph.D. Thesis, Institute of Computer Science, University of London, London, UK (1971).

[51] C.C. Paige, J. Inst. Maths. Appl. **10**, 373–381 (1972).

[52] C.C. Paige, J. Inst. Math. Appl. **18**, 341–349 (1976).

[53] C.C. Paige and M.A. Saunders, ACM Trans. Math. Softw. **8**, 43–71 (1982).

[54] C.C. Paige and M.A. Saunders, ACM Trans. Math. Softw. **8**, 195–209 (1982).

[55] C.C. Paige, Linear Algebra Appl. **34**, 235–258 (1980).

[56] C.C. Paige, M. Rozložník and Z. Strakoš, Rounding error analysis of the modified Gram-Schmidt GMRES, in preparation (2004).

[57] C.C. Paige and Z. Strakoš, Numer. Math. **91**, 117–146 (2002).

[58] C.C. Paige and Z. Strakoš, Numer. Math. **91**, 93–115 (2002).

[59] C.C. Paige and Z. Strakoš, SIAM J. Sci. Comput. **23**, 1898–1923 (2002).

[60] B.N. Parlett, The Symmetric Eigenvalue Problem, Prentice-Hall Series in Computational Mathematics (Prentice-Hall Inc., Englewood Cliffs, N.J., 1980).

[61] B.N. Parlett, The contribution of J.H. Wilkinson to numerical analysis, in: A History of Scientific Computing, ACM Press Hist. Ser. (ACM, New York, 1990), 17–30.

[62] B.N. Parlett, Do we fully understand the symmetric Lanczos algorithm yet?, in: Proceedings of the Cornelius Lanczos Centenary Conference (SIAM, Philadelphia, 1994), pp. 93–107.

[63] M. Rigal and J. Gaches, J. Assoc. Comput. Mach. **14**, 543–548 (1967).

[64] M. Rozložník, Numerical stability of the GMRES method, Ph.D. Thesis, Institute of Computer Science AS CR, Prague, Czech Republic (1997).

[65] M. Rozložník and Z. Strakoš, Variants of the residual minimizing Krylov space methods, in: Proceedings of the XI-th Summer School on Software and Algorithms of Numerical Mathematics, edited by I. Marek (1995), pp. 208–225.

[66] M. Rozložník, Z. Strakoš, and M. Tůma, On the role of orthogonality in the GMRES method, in: Proceedings of SOFSEM'96, Lecture Notes in Computer Science 1175 (Springer Verlag, Heidelberg, 1996), pp. 409–416.

[67] Y. Saad and M.H. Schultz, SIAM J. Sci. Stat. Comput. **7**, 856–869 (1986).

[68] S. Smale, Complexity theory and numerical analysis, in: Acta Numerica 6 (Cambridge Univ. Press, Cambridge, 1997), pp. 523–551.

[69] E. Stein (ed.), Error-controlled Adaptive Finite Elements in Solid Mechanics (John Wiley & Sons Ltd, Chichester, 2003).

[70] Z. Strakoš, Convergence and numerical behaviour of the Krylov space methods, NATO ASI Institute Algorithms for Large Sparse Linear Algebraic Systems: The State of the Art and Applications in Science and Engineering, edited by G. Winter Althaus and E. Spedicato, (Kluwer Academic Press, Dordrecht, 1998), pp. 175–197.

[71] Z. Strakoš and P. Tichý, ETNA **13**, 56–80 (2002).

[72] Z. Strakoš and P. Tichý, PAMM **3**, 553–554 (2003).

[73] Z. Strakoš and P. Tichý, Error estimation in preconditioned conjugate gradients, submitted to BIT Numerical Mathematics (2004).

[74] G.W. Strang and G.J. Fix, An Analysis of the Finite Element Method (Prentice-Hall, Englewood Cliffs, N.J., 1973).

[75] A. van der Sluis, Numer. Math. **14**, 14–23 (1969).

[76] P. Vinsome, Orthomin, an iterative method for solving sparse sets of simultaneous linear equations, in: Proceedings of the Fourth Symposium on Reservoir Simulation (Society of Petroleum Engineers of AIME, Los Angeles, 1976), pp. 149–159.

[77] H.F. Walker and L. Zhou, Numer. Lin. Alg. Appl. **1**, 571–581 (1994).

[78] K.F. Warnick, Nonincreasing error bound for the biconjugate gradient method, unpublished report, University of Illinois at Urbana-Champaign (2000).

[79] J.H. Wilkinson, Rounding Errors in Algebraic Processes (Her Majesty's Stationery Office, London, 1963).

[80] J.H. Wilkinson, The Algebraic Eigenvalue Problem (Oxford University Press, Oxford, 1965).

[81] B.I. Wohlmuth and R.H.W. Hoppe, Math. Comput. **68**, 1347–1378 (1999).

[82] D.M. Young, Iterative Solution of Large Linear Systems (Academic Press, New York, 1971).

[83] D.M. Young and K.C. Jea, Linear Algebra Appl. **34**, 159–194 (1980).