Notes

There are a number of textbooks and monographs on numerical linear algebra, and a particularly notable group have been appearing in the second half of the 1990s. Rather than give a full survey, we highlight three current books that every reader who wishes to go further with this subject should be aware of:

- Golub and Van Loan, *Matrix Computations*, 3rd ed. [GoVa96],
- Higham, *Accuracy and Stability of Numerical Algorithms* [Hig96],
- Demmel, *Applied Numerical Linear Algebra* [Dem97].

The book by Golub and Van Loan, in its earlier editions, has long been the bible of this field—encyclopedic in its coverage and its references to the literature. The book by Higham is another encyclopedic treatment, exceedingly careful about details, with an emphasis on stability but full of algorithmic information and insights of all kinds. The book by Demmel has almost the same title as the present volume but is entirely different in style, being more focused on latest developments and considerations of computer architecture, less on mathematical foundations.

Other texts on numerical linear algebra include [Cia89], [Dat95], [GMW91], [Hag88], [Ste73], and [Wat91].

Excellent texts are also available on various more specialized subjects, including least squares, eigenvalue problems, and iterative methods. These are listed in the appropriate paragraphs below. For direct sparse matrix methods, not covered in this book, two standard texts are [GeLi81] and

[DER86]. For software, also not covered here, some of the landmark contributions are LAPACK [And95] and its predecessors EISPACK [Smi76] and LINPACK [DBMS79], the Basic Linear Algebra Subprograms (BLAS) developed for simplifying coding of linear algebra operations and maximizing efficiency on particular machines [DDDH90], the MATLAB repository managed by The MathWorks, Inc. (`http://www.mathworks.com`), and the Netlib automatic software distribution system (`http://www.netlib.org`), which has processed about 13 million requests as of this writing.

Finally, we mention that when it comes to matters of nonnumerical linear algebra, our own habit is always to turn first to the two remarkable volumes by Horn and Johnson, [HoJo85] and [HoJo91].

We turn now to notes on the individual Lectures of this text.

Lecture 1. Matrix-Vector Multiplication. It is impossible to understand the spirit of twentieth-century numerical linear algebra without learning to think in terms of operations on rows and columns of matrices. Virtually all the standard algorithms are normally conceived in this way, though modifications appear when it comes to exploiting sparsity.

In principle, the fastest algorithms for many problems may be recursive ones that involve manipulations of submatrices and thus require a different way of thinking. For example, Klyuyev and Kokovkin-Shcherbak showed in 1965 that solving an $m \times m$ system of equations solely by row and column operations requires $O(m^3)$ operations [KlKo65], but the subsequent work of Strassen and others (Lecture 32) improved this to $O(m^{2.81})$ and below by recursive fracturing of the matrix into smaller blocks [Str69]. The divide-and-conquer algorithm for computing eigenvalues (Lecture 30) is another example where row and column operations are not enough. It is possible that in the next century, the importance of such algorithms will grow to the point that new ways of thinking will come to prevail in numerical linear algebra, but we are not there yet.

Determinants were central to linear algebra in the nineteenth century, but their importance has diminished. For one perspective on the reasons, see [Axl95].

In one form or another, the material of this first lecture can be found in numerous textbooks, such as [Str88]. If there is another text that takes square matrices by default to have dimensions $m \times m$ rather than $n \times n$, however, we have not found it.

Lecture 2. Orthogonal Vectors and Matrices. The content of this lecture is standard material in linear algebra, which generalizes in the infinite-dimensional case to standard material in the theory of Hilbert spaces.

Algorithms based on orthogonal matrices became widespread in the early 1960s with the work of Householder, Francis, Givens, Wilkinson, Golub, and others, as it came to be recognized that such algorithms combine theoretical elegance with outstanding properties of numerical stability. The rapid spread of this point of view can be seen in Wilkinson's 1965 monograph [Wil65] and in the classic textbooks [Ste73] and [LaHa95] (first published in 1974).

Lecture 3. Norms. Though the use of norms has long been a feature of functional analysis, it has been slower to become standard in linear algebra, and even today, these ideas are often not emphasized in nonnumerical linear algebra texts and courses. The explanation for this is probably that linear algebra is historically rooted in algebra rather than in analysis, and hence makes sense in vector spaces more general than $\mathbb{R}^m$ and $\mathbb{C}^m$. Most scientific applications, however, lead to real or complex numbers, for which analysis is meaningful as well as algebra. In any application with a notion of "size," norms are probably useful. One certainly needs them if one wants to talk about convergence.

The importance of norms in numerical linear algebra was emphasized in the 1964 book by Householder [Hou64] and in the brief 1967 text on Gaussian elimination and related matters by Forsythe and Moler [FoMo67].

In infinite dimensions, the use of dual norms as in Exercise 3.6 becomes the Hahn–Banach theorem [Kat76].

Lectures 4 and 5. The Singular Value Decomposition. The SVD for matrices was discovered independently by Beltrami (1873) and Jordan (1874) and again by Sylvester (1889), and related work was done by Autonne (1915), Takagi (1925), Williamson (1935), Eckart and Young (1939), and others. The infinite-dimensional generalization was developed in the context of integral equations by Schmidt (1907) and Weyl (1912); see [Smi70]. For historical discussions, see [HoJo91] and [Ste93].

Despite these deep roots, the SVD did not become widely known in applied mathematics until the late 1960s, when Golub and others showed that it could be computed effectively and used as the basis for many stable algorithms. Even after that time, perhaps because of numerical analysts' preoccupation with numerical stability, the mathematical world was slow to recognize the fundamental nature of the SVD. Again, the explanation may be the difference between algebra and analysis, for what makes the SVD so important is ultimately its analytic properties, as exemplified by Theorem 5.8. The importance of eigenvalues, by contrast, has been appreciated from the beginning, for eigenvalues are essentially algebraic in nature.

Closely related to the SVD is the *polar decomposition*, the representation of a matrix as a product of a symmetric positive definite matrix and a unitary matrix.

In the theory of Hilbert spaces, a compact operator is one that can be approximated by operators of finite rank, that is, one whose singular values decrease to zero.

Lecture 6. Projectors. Projectors are involved, explicitly or implicitly, whenever one expands a vector in a basis, and orthogonal projectors are one and the same as solutions of linear least squares problems. Perhaps it is unusual to make a discussion of projectors the starting point of a treatment of these matters, but only mildly so.

For a discussion of some relationships between norms of projectors and angles between complementary subspaces, see [IpMe95]. A full treatment of angles between subspaces is generally based on the *CS decomposition* [GoVa96].

Lecture 7. QR Factorization. The distinction between full and reduced QR factorizations appears wherever these ideas are applied, which means throughout numerical linear algebra, but this text is unusual in making the distinction explicit. More usually the QR factorization is defined in its full form, and columns of Q and rows of R are then stripped away as needed in applications. The same applies to the distinction between the full and reduced SVD.

The recognition of the importance of matrix factorizations for linear algebra computations is entirely a product of the computer age, beginning in the 1950s.

Concerning spectral methods for the numerical solution of partial differential equations, see [CHQZ88].

Lecture 8. Gram–Schmidt Orthogonalization. The idea of Gram (1883) and Schmidt (1907) is old and widely familiar, but its interpretation as a QR factorization is new to most students. In our view, this interpretation is an invaluable way to fix the Gram–Schmidt idea precisely in one's mind. The term QR factorization is due to Francis [Fra61].

The superiority of modified over classical Gram–Schmidt was first established by Rice (1966) and Björck (1967). Details and references are given in [Bjö96] and [Hig96].

Drawing pictures to calculate operation counts is nonstandard in respectable textbooks, since the same results are easily derived algebraically. But since we use the pictures in classroom teaching, we decided, why not include them in the book?

Lecture 9. MATLAB. As of 1996, about 150 textbooks in various fields of mathematics, science, and engineering have been published based on MATLAB, and the number is growing. Virtually all researchers in numerical linear algebra worldwide use MATLAB as their preferred programming language and environment, and in the Computer Science Department at Cornell, it is the principal language of all the numerical analysis courses. Information about MATLAB can be obtained from The MathWorks, 24 Prime Park Way, Natick, MA 01760, USA, tel. 508-647-7000, fax 508-647-7001, `info@mathworks.com`, `http://www.mathworks.com`.

Lecture 10. Householder Triangularization. Householder triangularization was introduced in a classic four-page paper in 1958 [Hou58]. (Householder reflectors themselves had been previously used as early as 1932, by

Turnbull and Aitken.) For thirty years, researchers in numerical linear algebra have gathered triennially for a conference on the state of their art, and these conferences are now known as Householder Symposia.

To make Householder reflections stable, it is not necessary to choose the sign as we have described. Alternative methods are described in [Par80] and [Hig96].

The symmetry between triangular orthogonalization and orthogonal triangularization is not novel mathematically, but as far as we are aware, it has not been stated in this epigrammatic form before.

For a beautiful and surprising connection between the modified Gram–Schmidt and Householder algorithms, see [BjPa92] or [Bjö96].

Lecture 11. Least Squares Problems. Who should get credit for the idea of least squares fitting? This question led to one of the great priority disputes in the history of mathematics, between Gauss, who invented the method in the 1790s, and Legendre, who first published it in 1805 (the same year in which Gauss invented the fast Fourier transform, which he also didn't publish). The honor was worth fighting over, as few ideas in mathematics have as far-reaching implications as least squares, but the fight brought honor to nobody; see [Sti86].

Troublesome square systems of equations, whose solutions may not seem to behave as they ought, arise frequently in discretization processes in scientific computing. The inverses of finite sections of an infinite matrix, for example, do not always converge as one might like to the sections of the infinite inverse matrix [Böt95]. Difficulties of this kind can often be avoided by looking at rectangular finite matrices instead and solving a least squares problem. This is just what was done in passing from Figure 11.1 to Figure 11.2.

The classic text by Lawson and Hanson gives a beautiful introduction to how numerical linear algebraists think about least squares problems; a lengthy appendix in the 1995 edition summarizes developments since the book's original publication in 1974 [LaHa95]. Other valuable introductions are presented in [Str88] and [GMW91]. A definitive work on numerical methods for least squares problems has recently been published by Björck [Bjö96], and it is here that one should turn for a full presentation of the state of the art.

Lecture 12. Conditioning and Condition Numbers. The idea of the condition number of a matrix was introduced in 1948 by Alan Turing [Tur48], the same Turing who founded theoretical computer science, who predicted the possibility of chemical waves long before they were discovered in the laboratory, and who contributed to the "Enigma" code-breaking effort that helped end the control of the Atlantic by German submarines in World War II.

A classic, more general paper on the subject of conditioning is [Ric66]; see also [Geu82].

The derivation of condition numbers is a special case of perturbation theory, and the definitive reference on perturbation theory for matrices and linear operators is [Kat76].

We have presented a simplified picture in that we only discuss normwise as opposed to componentwise condition numbers. For the latter increasingly important topic, see [Hig96] and [Dem97]. An example of a componentwise idea that we have omitted is that of the *Skeel condition number*, first proposed in [Ske79].

In many cases, the condition number of a well-posed problem is inversely related, at least approximately, to the distance to the nearest ill-posed problem. This point of view originated in a classic unpublished paper by Kahan [Kah72] and was developed in detail by Demmel [Dem87].

As mentioned in the text, Example 12.4 comes from Feynman [Fey85], who regrettably does not mention that the punch line of his story depends on ill-conditioning.

Concerning the ill-conditioning of roots of polynomials illustrated in Figure 12.1, two recent papers are [EdMu95] and [ToTr94], where pointers to earlier literature by Wilkinson and others can be found.

The result that Lebesgue constants for equispaced interpolation in n points grow asymptotically like $2^n/(e\,(n-1)\log n)$ was proved by Turetskii in 1940, but is not widely known. For historical comments, see [TrWe91].

Random matrices are of interest to statisticians and physicists as well as mathematicians, and the answers to the various parts of Exercise 12.3 can be found in [Ede88], [Gir90], [Meh91], and [TrVi98].

Lecture 13. Floating Point Arithmetic. Floating point arithmetic was first implemented as early as 1947, and from that point on, for many years, the details of the implementations by different manufacturers varied in ways hard to keep track of. The subject was simplified magnificently by the introduction and widespread adoption of the IEEE standard in the 1980s. For careful discussions of the issues involved, see [Gol91] and [Hig96].

Exercise 13.3 comes from Chapter 1 of [Dem97]. A similar plot for a sixth-order polynomial appears in Chapter 3 of [Code80]. Bob Lynch tells us that this example is due to Dave Dodson.

The results of Exercise 13.4, for which we thank Toby Driscoll, sometimes astonish people.

Lectures 14 and 15. Stability. The notion of backward stability is standard, and that of stability, reasonably so, but to define them formally via a precise interpretation of $O(\epsilon_{\text{machine}})$ is unusual. Most numerical analysts prefer to leave these ideas informal, so that they can be adapted to the particular features of different problems as needed. There are good reasons for this point of view, and we do not by any means claim that the course we have followed is the only proper one. Indeed, as mentioned in the text, for arbitrary problems of scientific computing, conditions involving $O(\epsilon_{\text{machine}})$ are probably too strict as a basis for definitions of stability.

Much the same formal definitions as ours can be found in [deJ77], a paper that has had less influence than it deserves.

Backward error analysis is one of the great ideas of numerical analysis, which made possible all the error estimates of numerical linear algebra that appear in this book. Credit for the development of this idea may be given to von Neumann and Goldstine, Turing, Givens, and Wilkinson. In recent years backward error analysis has been rediscovered by researchers in chaotic dynamical systems and developed under the name of *shadowing* [HYG88].

Lecture 16. Stability of Householder Triangularization. The astonishing difference between the low accuracy of the computed matrix factors Q and R individually and the high accuracy of their product exemplifies why backward error analysis is so powerful. In the 1950s and 1960s Wilkinson showed that similar effects occur in virtually every matrix algorithm. The first author was lucky enough to hear lectures by Wilkinson on these matters, which conveyed the wonder of such effects with unforgettable enthusiasm.

Theorems 16.1 and 16.2 are due to Wilkinson [Wil65], and proofs can also be found in §18.3 of [Hig96]. In the remainder of this book we state a number of stability theorems without proof. In most cases a proof, or a reference to another source containing a proof, can be found in [Hig96].

Lecture 17. Stability of Back Substitution. Carrying out a rounding error analysis in full detail can be deeply satisfying; some students have found this the most exciting lecture of the book. The results are originally due to Wilkinson; see [Wil61], [FoMo67], [Hig96].

Our remark at the end of this lecture indicates why we prefer to state results in terms of $O(\epsilon_{\text{machine}})$ rather than give explicit constants. Many numerical analysts feel differently, however, including N. J. Higham [Hig96], and we admit that it is reassuring to know that in most cases, explicit constants have been worked out and recorded in print.

Exercise 17.3, involving random matrices with entries ±1, is based on [TrVi98] and subsequent developments from that paper.

Lecture 18. Conditioning of Least Squares Problems. The literature on this subject is not especially easy to read, partly because of the complication of rank-deficiency, which we have ignored. Several of the results in this area were first derived by Wedin [Wed73], and a paper by Stewart summarizes many of the key issues [Ste77]. The 1990 book by Stewart and Sun goes further, but is difficult reading [StSu90], and a good place to go for recent information is [Bjö96]. The papers [Geu82] and [Gra96] give exact condition numbers with respect to the Frobenius norm. For the 2-norm, the bottom row of Theorem 18.1 represents upper bounds; as far as we are aware, exact results are not known.

The geometric view of these conditioning questions is not always described explicitly, but one place where it is emphasized is [vdS75].

The differentiation of pseudoinverses is not useful just for stability analysis;

it also has algorithmic consequences. An influential paper in this area is [GoPe73].

Exercise 18.1 comes from [GMW91].

Lecture 19. Stability of Least Squares Algorithms. This is standard material, discussed in many books, including [Bjö96], [GoVa96], and [Hig96]. The subject of QR factorization with column pivoting is a large one belonging to the general area of *rank-revealing factorizations*; see [Bjö96] and [ChIp94].

Lecture 20. Gaussian Elimination. There is nothing unusual here except our deferral of this topic to the middle of the book. Gauss himself worked with positive definite systems around 1809; Jacobi extended the elimination idea to general matrices around 1857. The interpretation as a matrix factorization was first developed by Dwyer in 1944 [Dwy44].

Lecture 21. Pivoting. The terms "partial" and "complete" are due to Wilkinson in the 1950s, but pivoting was already being used as early as 1947 by von Neumann and Goldstine.

Numerous variants of the pivoting idea have found application in various computations of linear algebra. One example is the technique of *threshold pivoting*, in which one relaxes the pivot condition so that the pivot element need not be the largest in its column as long as it is within a prescribed factor of the largest. Though such a strategy may diminish the stability of the algorithm, it provides additional freedom that may be used to pick orderings that minimize fill-in in the treatment of sparse matrices. See [DER86].

Lecture 22. Stability of Gaussian Elimination. In the mid-1940s it was predicted by Hotelling and von Neumann and others that Gaussian elimination must be unstable because of exponentially compounding rounding errors, making the method unsuitable for problems of dimensions greater than a few dozen. By the early 1950s, computational experience had revealed that the algorithm was stable after all. Explaining this observation was a major theoretical challenge, and Wilkinson became famous for his contributions to the subject, which reduced the question of stability to the question of the size of the growth factor. Wilkinson's analysis was recorded in a landmark paper of 1961 [Wil61].

Wilkinson and his contemporaries did not address the problem of why, in practice, nothing like the worst-case growth factor is ever observed. In *The Algebraic Eigenvalue Problem* he comments, "experience suggests that though such a bound is attainable it is quite irrelevant for practical purposes" [Wil65], and similar remarks appear in texts from the 1960s to the 1990s. The first substantial paper on the behavior of growth factors was [TrSc90], which gave empirical evidence and other arguments that the phenomenon of practical stability is entirely statistical. The present lecture of this book, making the connection between large growth factors and exponentially skewed column spaces, represents the first explanation in print of this statistical phenomenon;

a fuller analysis is forthcoming.

Recently Wright [Wri93] and Foster [Fos94] have constructed examples of matrices for which Gaussian elimination is unstable which, though they apparently did not in fact arise in actual computations, plausibly might have done so.

Lecture 23. Cholesky Factorization. Cholesky factorization can be described, and programmed, in many different ways, and this lecture offers just one of the possibilities. As a method that takes advantage of a kind of structure of the matrix A (positive definiteness), Cholesky factorization is just the tip of an iceberg. Methods for all kinds of structured matrices have been devised, including symmetric indefinite, banded, arrowhead, Vandermonde, Toeplitz, Hankel, and other matrices; see [GoVa96].

As technology advances, the ingenious ideas that make progress possible tend to vanish into the inner workings of our machines, where only experts may be aware of their existence. So it often is with numerical algorithms, never of much interest to the public, yet hidden inside most of the appliances we use. Exercise 23.3 illustrates this phenomenon in a small way. Traditionally, an engineer wanting to solve a system of equations would choose the right method based on the properties of the system, but high-level tools like Matlab's " \ " prefer to make these decisions by themselves. Still, by careful experimentation we can still deduce some of the advances in numerical analysis underlying those decisions.

Lecture 24. Eigenvalue Problems. This is all standard material, though the emphasis is different from what one would find in a nonnumerical text. For example, we mention the Schur factorization, which is important in computations, but not the Jordan canonical form, which usually is not, for reasons explained in [GoWi76].

Gerschgorin's theorem (Exercise 24.1) has many generalizations, some of which are reviewed in [BrRy91] and [BrMe94].

The abbreviations "ew" and "ev" (Exercise 24.1) are not standard, but perhaps they should be. We find them indispensable in the classroom.

Lecture 25. Overview of Eigenvalue Algorithms. Though more than thirty years old, Wilkinson's *The Algebraic Eigenvalue Problem* [Wil65] is still a valuable reference for details on all kinds of questions related to the computation of eigenvalues. For symmetric matrix problems, the 1980 book by Parlett is a standard reference and makes excellent reading [Par80]. For more recent developments, see [Dem97].

Though it is not mentioned in many textbooks, the $O(\log(|\log(\epsilon_{\text{machine}})|))$ iteration count of Exercise 25.2 applies to superlinearly converging algorithms all across scientific computing.

Lecture 26. Reduction to Hessenberg or Tridiagonal Form. The reduction of a matrix to Hessenberg form can also be carried out by nonunitary operations, and the asymptotic operation count is only half that of (26.1). In principle, nonunitary reductions are not always stable, but in practice they work very well. In the EISPACK software library of the 1970s [Smi76], nonunitary reduction was recommended as the default and unitary reduction was offered as an alternative. In the more recent LAPACK library [And95], only unitary reductions are provided for. Why is (nonunitary) Gaussian elimination the standard method for linear systems while unitary operations are standard for eigenvalue problems? Though unitary reductions are convenient for estimating eigenvalue condition numbers and related purposes, there seems to be no entirely compelling answer. The explanation may be that in view of the greater complexity of the eigenvalue problem, involving both a direct phase and an iterative one, numerical analysts have been less willing to take chances with stability.

For more on pseudospectra, including computed examples, see [Tre91] and [Tre97].

Lecture 27. Rayleigh Quotient, Inverse Iteration. Inverse iteration originated with Wielandt in the 1940s; for a history, see [Ips97]. For details on the phenomenon that an ill-conditioned matrix does not cause instability (Exercise 27.5), see [PeWi79], [Par80], or [GoVa96].

The convergence of the Rayleigh quotient iteration and its nonsymmetric generalization was analyzed in a sequence of papers by Ostrowski in the late 1950s [Ost59].

One of the best-known algorithms for computing zeros of polynomials is that of Jenkins and Traub. As pointed out in the original paper [JeTr70] and discussed also in the appendix of [ToTr94], the Jenkins–Traub iteration can be interpreted as a scheme for taking advantage of sparsity in a Rayleigh quotient iteration applied to a companion matrix, so that the work per step is reduced from $O(m^3)$ to $O(m)$.

Lectures 28 and 29. QR Algorithm. The QR algorithm was invented independently in 1961 by Francis [Fra61] and Kublanovskaya [Kub61], based on the earlier LR algorithm of Rutishauser, and came into worldwide use through the software package EISPACK [Smi76]. Our presentation is adapted from [Wat82]. Extensive discussions are given in [Par80] and [Wat91].

The computation of eigenvalues of matrices is one of the problems that has been most extensively studied by numerical analysts, and the amount of understanding incorporated in state-of-the-art software such as LAPACK [And95] is very great. Our "practical" Algorithm 28.2 certainly does not mention all the subtleties that must be addressed for robust computation. For example, when the QR algorithm is implemented in practice, the shifts are introduced in a more stable implicit manner by means of "chasing the bulge." See [Par80], [GoVa96], or [Dem97], where discussions of the properties of various shifts can also be found.

Lecture 30. Other Eigenvalue Algorithms. Jacobi's major paper on his eigenvalue algorithm appeared in 1846 [Jac46]; he used the method to find eigenvalues of a 7×7 matrix associated with the seven planets then known in the solar system. A classic modern reference is [FoHe60], and more recent developments, including the variant based on 4×4 blocks and quarternions, can be found in [Mac95] and the references therein. Because it avoids the tridiagonalization step, the Jacobi algorithm when carefully implemented is more accurate than the QR algorithm in a componentwise sense; see [DeVe92].

Divide-and-conquer algorithms were introduced by Cuppen in 1981 [Cup81] and made famous by Dongarra and Sorensen [DoSo87]. The literature since then is extensive. Some of the critical developments concerning stability, as well as the idea of acceleration via the fast multipole method, were introduced by Gu and Eisenstat; see [GuEi95] and [Dem97].

Lecture 31. Computing the SVD. The era of numerical computations of the SVD began in 1965 with the publication of a paper by Golub and Kahan [GoKa65], which recommended bidiagonalization by Householder reflections for Phase 1. The idea of applying the QR algorithm for Phase 2 is sometimes credited to the same paper, but in fact, the QR algorithm is not mentioned there, nor are the papers of Francis [Fra61] referenced. The key ideas developed very quickly in the late 1960s, however, through work by Golub, Kahan, Reinsch, and Businger.

Our discussion of alternative methods for Phase 1 is taken from [Bau94], where details concerning singular vectors as well as values can be found. For information about Phase II, see [GoVa96] and [Dem97].

Lecture 32. Overview of Iterative Methods. The history of the emergence of Krylov subspace iterative methods is fascinating. The foundations were laid in the early 1950s, but the machines of that era were too slow for these methods to be superior for most problems. Not only were they not extensively used, naturally enough, but their ultimate advantages concerning asymptotic complexity were not perceived very clearly. Nowadays, it is automatic to take note of the asymptotic complexity of algorithms; in the 1950s, it was not.

On the other hand, certain "classical iterations" such as Gauss–Seidel and SOR were used extensively in the 1950s for problems arising from discretizations of partial differential equations. We have given no attention to these methods here, as they are described in many books but are of diminishing practical importance today. A classic reference on this subject is [Var62].

For sparse direct matrix algorithms, see [GeLi81] and [DER86].

What is the dimension m of a "large" matrix, as a function of time? In recent years information on the subject has been collected by Edelman, who reported in 1994, for example, that he was unaware yet of any solutions of dense systems with $m > 100{,}000$, though matrices with $m = 76{,}800$ had been treated [Ede94].

A number of books have recently been written on iterative methods; we recommend in particular the monographs by Saad on eigenvalues [Saa92] and linear systems [Saa96] and the upcoming text on linear systems by Greenbaum [Gre97]. Other books on the subject include [Axe94], with extensive information on preconditioners, [Kel95], which emphasizes generalizations to nonlinear problems, and [Bru95], [Fis96], [Hac94], and [Wei96].

Since the 1950s it has been recognized that Krylov subspace methods are applicable to linear operators, not just matrices. An early reference in this vein is [Dan71], and a recent advanced one is [Nev93].

The Krylov idea of projection onto low-dimensional subspaces sounds analogous to one of the central ideas of numerical computation—discretization of a continuous problem so that it becomes finite-dimensional. One might ask whether this is more than an analogy, and if so, whether it might be possible to combine discretization and iteration into one process rather than separately replacing ∞ by m (discretization) and m by n (iteration). The answer is certainly yes, at least in some circumstances. However, many of the possibilities of this kind have not yet been explored, and at present, most scientific computations still keep discretization and iteration separate.

Strassen's famous paper appeared in 1969 [Str69], and pointers to the algorithms with still lower exponents represented in Figure 32.2 can be found in [Pan84] and [Hig96]. The current best exponent of 2.376 is due to Coppersmith and Winograd [CoWi90].

What we have called "the fundamental law of computer science" (p. 246) does not usually go by this name. This principle is discussed in [AHU74]; we do not know where it was first enunciated.

Lecture 33. The Arnoldi Iteration. Arnoldi's original paper was written in 1951, but his intentions were rather far from current ones [Arn51]. It took a long while for the various connections between the Arnoldi, Lanczos, CG, and other methods to be recognized.

Lecture 34. How Arnoldi Locates Eigenvalues. The convergence of the Lanczos iteration is reasonably well understood; some of the key papers are by Kaniel [Kan66], Paige [Pai71], and Saad [Saa80]. The convergence of the more general Arnoldi iteration, however, is not fully understood. For some of the results that are available, see [Saa92].

Our discussion in terms of lemniscates is nonstandard. The connection with polynomial approximation, including the notions of ideal Arnoldi and GMRES polynomials, is developed in [GrTr94]. An algorithm for computing these polynomials based on semidefinite programming is presented in [ToTr98], together with examples relating lemniscates to pseudospectra. The idea of estimating pseudospectra via the Arnoldi iteration comes from [ToTr96].

Concerning the "Note of Caution," see [TTRD93], [Tre91], and [Tre97].

Lecture 35. GMRES. The GMRES algorithm was proposed surprisingly recently, by Saad and Schultz in 1986 [SaSc86], though various related algorithms had appeared earlier.

Lecture 36. The Lanczos Iteration. The Lanczos iteration dates to 1950 [Lan50]. Though closely related to conjugate gradients, it was conceived independently. The Lanczos iteration was "rediscovered" in the 1970s, as tractable matrix problems grew to the size where it became competitive with other methods [Pai71]. A two-volume treatment was given in 1985 by Cullum and Willoughby [CuWi85].

The connection of Krylov subspace iterations with potential theory (electric charges) via polynomial approximation is well established. For a detailed analysis of what can and cannot be inferred about convergence from potential theory, see [DTT98].

Lecture 37. From Lanczos to Gauss Quadrature. Since 1969 it has been appreciated that the right way to compute Gauss quadrature nodes and weights is via tridiagonal matrix eigenvalue problems [GoWe69]. The brief presentation here describes the connection in full except for one omitted point: the relation of the weights to the first components of the eigenvectors, which can be derived from the Christoffel–Darboux formula. For information on this and other matters related to orthogonal polynomials, the classic reference is the book by Szegő [Sze75].

On p. 289 it is remarked that nth-order Newton–Cotes formulas have coefficients of order 2^n for large n. As Newton–Cotes formulas can be derived by interpolation, this is essentially the same factor 2^n mentioned in connection with Lebesgue constants in the notes on Lecture 12, above.

Lecture 38. Conjugate Gradients. The conjugate gradient iteration originated with Hestenes and Stiefel independently, but communication between the two men was established early enough (August 1951) for the original major paper on the subject, one of the great classics of numerical analysis, to be a joint effort [HeSt52]. Like the Lanczos iteration, CG was "rediscovered" in the 1970s, and soon became a mainstay of scientific computing. For the closely intertwined history of the CG and Lanczos iterations, see [GoOL89].

Much of what is known about the behavior of the CG iteration in floating point arithmetic is due to Greenbaum and her coauthors; see [Gre97].

Lecture 39. Biorthogonalization Methods. The biconjugate gradient iteration originated with Lanczos in 1952 [Lan52] and was revived (and christened) by Fletcher in 1976 [Fle76]. The other methods mentioned in the text are look-ahead Lanczos [PTL85], CGS [Son89], QMR [FrNa91], Bi-CGSTAB [vdV92], and TFQMR [Fre93]. For a survey as of 1991, see [FGN92], and for a description of the deep connections of these algorithms with orthogonal polynomials, continued fractions, Padé approximation, and other topics, see [Gut92]. Another valuable reference is [Saa96].

For comparisons of the matrix properties that determine convergence of the various types of nonsymmetric matrix iterations, see [NRT92], where Exercises 39.1 and 39.2 are also addressed. For specific discussions of the relationships between BCG and QMR, see [FrNa91] and [CuGr96], where it is pointed out that spikes in the BCG convergence curve correspond in a precise way to flat (slow-progress) portions of the QMR convergence curve.

Lecture 40. Preconditioning. The word "preconditioning" originated with Turing in 1948, and some of the early contributions in the context of matrix iterations were due to Hestenes, Engeli, Wachspress, Evans, and Axelsson. The idea became famous in the 1970s with the introduction of incomplete factorization by Meijerink and van der Vorst [Meva77], and another influential paper of that decade was [CGO76]. For summaries of the current state of the art we recommend [Axe94] and [Saa96]. Domain decomposition is discussed in [SBG96], and the use of an unstable direct method as a preconditioner is considered in [Ske80]. The idea of circulant preconditioners for Toeplitz matrices originated with Strang [Str86] and has been widely generalized since then.

What about speeding up an iteration by changing the preconditioner adaptively at each step, just as the Rayleigh quotient shift speeds up inverse iteration from linear to cubic convergence? This idea is a promising one, and has recently been getting some attention; see [Saa96].

Preconditioners for eigenvalue problems have come into their own in the 1990s, though Davidson's original paper dates to 1975 [Dav75]; a good place to begin with these methods is [Saa92]. Polynomial acceleration devices have been developed by Chatelin [Cha93], Saad, Scott, Lehoucq and Sorensen [LeSo96], and others. Shift-and-invert Arnoldi methods have been developed by Saad and Spence, and rational Krylov iterations by Ruhe; for a recent survey see [MeRo96]. The Jacobi–Davidson algorithm was introduced by Sleijpen and van der Vorst [Slvd96].

Bibliography

[AHU74] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

[And95] E. Anderson et al., *LAPACK Users' Guide*, 2nd ed., SIAM, Philadelphia, 1995.

[Arn51] W. E. Arnoldi, *The principle of minimized iteration in the solution of the matrix eigenvalue problem*, Quart. Appl. Math. 9 (1951), 17–29.

[Axe94] O. Axelsson, *Iterative Solution Methods*, Cambridge U. Press, Cambridge, UK, 1994.

[Axl95] S. Axler, *Down with determinants*, Amer. Math. Monthly 102 (1995), 139–154.

[Bar94] R. Barrett et al., *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, 1994.

[Bau94] D. Bau, *Faster SVD for matrices with small m/n*, TR94-1414, Computer Science Dept., Cornell U., 1994.

[Bjö96] Å. Björck, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, 1996.

[BjPa92] Å. Björck and C. C. Paige, *Loss and recapture of orthogonality in the modified Gram–Schmidt algorithm*, SIAM J. Matrix Anal. Appl. 13 (1992), 176–190.

[Böt95] A. Böttcher, *Infinite matrices and projection methods*, in P. Lancaster, ed., Lectures on Operator Theory and Its Applications, Amer. Math. Soc., Providence, RI, 1995.

[BrMe94] R. A. Brualdi and S. Mellendorf, *Regions in the complex plane containing the eigenvalues of a matrix*, Amer. Math. Monthly 101 (1994), 975–985.

[BrRy91] R. A. Brualdi and H. J. Ryser, *Combinatorial Matrix Theory*, Cambridge U. Press, Cambridge, UK, 1991.

[Bru95] A. M. Bruaset, *A Survey of Preconditioned Iterative Methods*, Addison-Wesley Longman, Harlow, Essex, UK, 1992.

[CHQZ88] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics*, Springer-Verlag, New York, 1988.

[ChIp94] S. Chandrasekaran and I. C. F. Ipsen, *On rank-revealing factorisations*, SIAM J. Matrix Anal. Appl. 15 (1994), 592–622.

[Cha93] F. Chatelin, *Eigenvalues of Matrices*, Wiley, New York, 1993.

[Cia89] P. G. Ciarlet, *Introduction to Numerical Linear Algebra and Optimisation*, Cambridge U. Press, Cambridge, UK, 1989.

[CGO76] P. Concus, G. H. Golub, and D. P. O'Leary, *A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations*, in J. R. Bunch and D. J. Rose, eds., Sparse Matrix Computations, Academic Press, New York, 1976.

[Code80] S. D. Conte and C. de Boor, *Elementary Numerical Analysis: An Algorithmic Approach*, 3rd ed., McGraw-Hill, New York, 1980.

[CoWi90] D. Coppersmith and S. Winograd, *Matrix multiplication via arithmetic progressions*, J. Symbolic Comput. 9 (1990), 251–280.

[CuGr96] J. Cullum and A. Greenbaum, *Relations between Galerkin and norm-minimizing iterative methods for solving linear systems*, SIAM J. Matrix Anal. Appl. 17 (1996), 223–247.

[CuWi85] J. K. Cullum and R. A. Willoughby, *Lanczos Algorithms for Large Symmetric Eigenvalue Computations, v.* 1 *and* 2, Birkhäuser, Boston, 1985.

[Cup81] J. J. M. Cuppen, *A divide and conquer method for the symmetric tridiagonal eigenproblem*, Numer. Math. 36 (1981), 177–195.

[Dan71] J. W. Daniel, *The Approximate Minimization of Functionals*, Prentice Hall, Englewood Cliffs, NJ, 1971.

[Dat95] B. N. Datta, *Numerical Linear Algebra and Applications*, Brooks/Cole, Pacific Grove, CA, 1995.

[Dav75] E. R. Davidson, *The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real symmetric matrices*, J. Comp. Phys. 17 (1975), 87–94.

[deJ77] L. S. de Jong, *Towards a formal definition of numerical stability*, Numer. Math. 28 (1977), 211–219.

[Dem87] J. W. Demmel, *On condition numbers and the distance to the nearest ill-posed problem*, Numer. Math. 51 (1987), 251–289.

[Dem97] J. W. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

[DeVe92] J. Demmel and K. Veselić, *Jacobi's method is more accurate than QR*, SIAM J. Matrix Anal. Appl. 13 (1992), 1204–1245.

[DBMS79] J. J. Dongarra, J. R. Bunch, C. B. Moler, and G. W. Stewart, *LINPACK Users' Guide*, SIAM, Philadelphia, 1979.

[DDDH90] J. J. Dongarra, J. J. Du Croz, I. S. Duff, and S. J. Hammarling, *Algorithm* 679. *A set of level* 3 *basic linear algebra subprograms: Model implementation and test programs*, ACM Trans. Math. Software 16 (1990), 18–28.

[DoSo88] J. J. Dongarra and D. C. Sorensen, *A fully parallel algorithm for the symmetric eigenvalue problem*, SIAM J. Sci. Stat. Comput. 8 (1987), s139–s154.

[DTT98] T. A. Driscoll, K.-C. Toh, and L. N. Trefethen, *From potential theory to matrix iterations in six steps*, SIAM Review 40 (1998), 547–578.

[DER86] I. S. Duff, A. M. Erisman, and J. K. Reid, *Direct Methods for Sparse Matrices*, Clarendon Press, Oxford, UK, 1986.

[Dwy44] P. S. Dwyer, *A matrix presentation of least squares and correlation theory with matrix justification of improved methods of solutions*, Ann. Math. Stat. 15 (1944), 82–89.

[Ede88] A. Edelman, *Eigenvalues and condition numbers of random matrices*, SIAM J. Matrix Anal. Appl. 9 (1988), 543–560.

[Ede94] A. Edelman, *Large numerical linear algebra in* 1994*: The continuing influence of parallel computing*, Proc. 1994 Scalable High Performance Computing Conf., IEEE Computer Soc. Press, Los Alamitos, CA, 1994, 781–787.

[EdMu95] A. Edelman and H. Murakami, *Polynomial roots from companion matrix eigenvalues*, Math. Comp. 64 (1995), 763–776.

[Fey85] R. P. Feynman, *Surely You're Joking, Mr. Feynman! Adventures of a Curious Character*, Norton, New York, 1985.

[Fis96] B. Fischer, *Polynomial Based Iteration Methods for Symmetric Linear Systems*, Wiley-Teubner, Chichester, UK, 1996.

[Fle76] R. Fletcher, *Conjugate gradient methods for indefinite systems*, in G. A. Watson, ed., Numerical Analysis Dundee 1975, Lec. Notes in Math. v. 506, Springer-Verlag, Berlin, 1976, 73–89.

[FoHe60] G. E Forsythe and P. Henrici, *The cyclic Jacobi method for computing the principal values of a complex matrix*, Trans. Amer. Math. Soc. 94 (1960), 1–23.

[FoMo67] G. E. Forsythe and C. B. Moler, *Computer Solution of Linear Algebraic Systems*, Prentice Hall, Englewood Cliffs, NJ, 1967.

[Fos94] L. V. Foster, *Gaussian elimination with partial pivoting can fail in practice*, SIAM J. Matrix Anal. Appl. 15 (1994), 1354–1362.

[Fra61] J. G. F. Francis, *The QR transformation: A unitary analogue to the LR transformation, parts* I *and* II, Computer J. 4 (1961), 256–72 and 332–45.

[Fre93] R. W. Freund, *A transpose-free quasi-minimal residual algorithm for non-hermitian linear systems*, SIAM J. Sci. Stat. Comput. 13 (1992), 425–448.

[FGN92] R. W. Freund, G. H. Golub, and N. M. Nachtigal, *Iterative solution of linear systems*, Acta Numerica 1 (1992), 57–100.

[FrNa91] R. W. Freund and N. M. Nachtigal, *QMR: A quasi-minimal residual method for non-Hermitian linear systems*, Numer. Math. 60 (1991), 315–339.

[GeLi81] A. George and J. W.-H. Liu, *Computer Solution of Large Sparse Positive Definite Systems*, Prentice Hall, Englewood Cliffs, NJ, 1981.

[Geu82] A. J. Geurts, *A contribution to the theory of condition*, Numer. Math. 39 (1982), 85–96.

[GMW91] P. E. Gill, W. Murray, and M. H. Wright, *Numerical Linear Algebra and Optimization*, Addison-Wesley, Redwood City, CA, 1991.

[Gir90] V. L. Girko, *Theory of Random Determinants*, Kluwer, Dordrecht, the Netherlands, 1990.

[Gol91] D. Goldberg, *What every computer scientist should know about floating-point arithmetic*, ACM Computing Surveys 23 (1991), 5–48.

[GoKa65] G. Golub and W. Kahan, *Calculating the singular values and pseudo-inverse of a matrix*, SIAM J. Numer. Anal. 2 (1965), 205–224.

[GoOL89] G. H. Golub and D. P. O'Leary, *Some history of the conjugate gradient and Lanczos methods*, SIAM Review 31 (1989), 50–100.

[GoPe73] G. H. Golub and V. Pereyra, *The differentiation of pseudoinverses and nonlinear least squares problems whose variables separate*, SIAM J. Numer. Anal. 10 (1973), 413–432.

[GoVa96] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins U. Press, Baltimore, 1996.

[GoWe69] G. H. Golub and J. H. Welsch, *Calculation of Gauss quadrature rules*, Math. Comp. 23 (1969), 221–230.

[GoWi76] G. H. Golub and J. H. Wilkinson, *Ill-conditioned eigensystems and the computation of the Jordan canonical form*, SIAM Review 18 (1976), 578–619.

[Gra96] S. Gratton, *On the condition number of linear least squares problems in a weighted Frobenius norm*, BIT 36 (1996), 523–530.

[Gre97] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1997.

[GrTr94] A. Greenbaum and L. N. Trefethen, *GMRES/CR and Arnoldi/Lanczos as matrix approximation problems*, SIAM J. Sci. Comput. 15 (1994), 359–368.

[GuEi95] M. Gu and S. C. Eisenstat, *A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem*, SIAM J. Matrix Anal. Appl. 16 (1995), 172–191.

[Gut92] M. H. Gutknecht, *A completed theory of the unsymmetric Lanczos process and related algorithms, part* I, SIAM J. Matrix Anal. Appl. 13 (1992), 594–639.

[Hac94] W. Hackbusch, *Iterative Solution of Large Sparse Linear Systems of Equations*, Springer-Verlag, Berlin, 1994.

[Hag88] W. Hager, *Applied Numerical Linear Algebra*, Prentice Hall, Englewood Cliffs, NJ, 1988.

[HYG88] S. M. Hammel, J. A. Yorke, and C. Grebogi, *Numerical orbits of chaotic processes represent true orbits*, Bull. Amer. Math. Soc. 19 (1988), 465–469.

[HeSt52] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Stand. 49 (1952), 409–436.

[Hig96] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, 1996.

[HoJo85] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge U. Press, Cambridge, UK, 1985.

[HoJo91] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge U. Press, Cambridge, UK, 1991.

[Hou58] A. S. Householder, *Unitary triangularization of a nonsymmetric matrix*, J. Assoc. Comput. Mach. 5 (1958), 339–342.

[Hou64] A. S. Householder, *The Theory of Matrices in Numerical Analysis*, Blaisdell, New York, 1964.

[Ips97] I. C. F. Ipsen, *A history of inverse iteration*, in B. Huppert and H. Schneider, eds., Helmut Wielandt, Mathematische Werke, Mathematical Works, v. 2, Walter de Gruyter, Berlin, 1996, 453–463.

[IpMe95] I. C. F. Ipsen and C. D. Meyer, *The angle between complementary subspaces*, Amer. Math. Monthly 102 (1995), 904–911.

[Jac46] C. G. J. Jacobi, *Über ein leichtes Verfahren die in der Theorie der Säcularstörungen vorkommenden Gleichungen numerisch aufzulösen*, J. Reine Angew. Math. 30 (1846), 51–94.

[JeTr70] M. A. Jenkins and J. F. Traub, *A three-stage variable-shift iteration for polynomial zeros and its relation to generalized Rayleigh iteration*, Numer. Math. 14 (1970), 252–263.

[Kah72] W. M. Kahan, *Conserving confluence curbs ill-condition*, unpublished manuscript, 1972.

[Kan66] S. Kaniel, *Estimates for some computational techniques in linear algebra*, Math. Comp. 20 (1966), 369–378.

[Kat76] T. Kato, *Perturbation Theory for Linear Operators*, 2nd ed., Springer-Verlag, New York, 1976.

[Kel95] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia, 1995.

[KlKo65] V. V. Klyuyev and N. I. Kokovkin-Shcherbak, *On the minimization of the number of arithmetic operations for the solution of linear algebraic systems of equations*, Zh. Vychisl. Mat. i Mat. Fiz. 5 (1965), 21–33; translated from the Russian by G. J. Tee, Tech. Rep. CS24, Computer Science Dept., Stanford University, 1965.

[Koz92] D. C. Kozen, *The Design and Analysis of Algorithms*, Springer-Verlag, New York, 1992.

[Kry31] A. N. Krylov, *On the numerical solution of equations which in technical questions are determined by the frequency of small vibrations of material systems*, Izv. Akad. Nauk. S. S. S. R. Otd Mat. Estest. 1 (1931), 491–539.

[Kub61] V. N. Kublanovskaya, *On some algorithms for the solution of the complete eigenvalue problem*, USSR Comp. Math. Phys. 3 (1961), 637–657.

[Lan50] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Stand. 45 (1950), 255–282.

[Lan52] C. Lanczos, *Solution of systems of linear equations by minimized iterations*, J. Res. Nat. Bur. Stand. 49 (1952), 33–53.

[LaHa95] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*, SIAM, Philadelphia, 1995 (reprinting with corrections and a new appendix of a 1974 Prentice Hall text).

[LeSo96] R. B. Lehoucq and D. C. Sorensen, *Deflation techniques for an implicitly restarted Arnoldi iteration*, SIAM J. Matrix Anal. Appl. 17 (1996), 789–821.

[Mac95] N. Mackey, *Hamilton and Jacobi meet again: Quaternions and the eigenvalue problem*, SIAM J. Matrix Anal. Appl. 16 (1995), 421–435.

[MeRo96] K. Meerbergen and D. Roose, *Matrix transformations for computing rightmost eigenvalues of large sparse non-symmetric eigenvalue problems*, IMA J. Numer. Anal. 16 (1996), 297–346.

[Meh91] M. L. Mehta, *Random Matrices*, 2nd ed., Academic Press, San Diego, 1991.

[Meva77] J. Meijerink and H. van der Vorst, *An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix*, Math. Comp. 31 (1977), 148–162.

[NRT92] N. M. Nachtigal, S. C. Reddy, and L. N. Trefethen, *How fast are nonsymmetric matrix iterations?*, SIAM J. Matrix Anal. Appl. 13 (1992), 778–795.

[Nev93] O. Nevanlinna, *Convergence of Iterations for Linear Equations*, Birkhäuser, Basel, 1993.

[Ost59] A. M. Ostrowski, *On the convergence of the Rayleigh quotient iteration for the computation of characteristic roots and vectors,* IV*. Generalized Rayleigh quotient for nonlinear elementary divisors*, Arch. Rational Mech. Anal. 3 (1959), 341–347.

[Pai71] C. C. Paige, *The Computation of Eigenvalues and Eigenvectors of Very Large Sparse Matrices*, PhD diss., U. of London, 1971.

[Pan84] V. Pan, *How to Multiply Matrices Faster*, Lec. Notes in Comp. Sci., v. 179, Springer-Verlag, Berlin, 1984.

[Par80] B. N. Parlett, *The Symmetric Eigenvalue Problem*, Prentice Hall, Englewood Cliffs, NJ, 1980.

[PTL85] B. N. Parlett, D. R. Taylor, and Z. A. Liu, *A look-ahead Lanczos algorithm for unsymmetric matrices*, Math. Comp. 44 (1985), 105–124.

[PeWi79] G. Peters and J. H. Wilkinson, *Inverse iteration, ill-conditioned equations and Newton's method*, SIAM Review 21 (1979), 339–360.

[Ric66] J. R. Rice, *A theory of condition*, SIAM J. Numer. Anal. 3 (1966), 287–310.

[Saa80] Y. Saad, *On the rates of convergence of the Lanczos and the block Lanczos methods*, SIAM J. Numer. Anal. 17 (1980), 687–706.

[Saa92] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Manchester U. Press, Manchester, UK, 1992.

[Saa96] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing, Boston, 1996.

[SaSc86] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. 7 (1986), 856–869.

[Ske79] R. D. Skeel, *Scaling for numerical stability in Gaussian elimination*, J. Assoc. Comput. Mach. 26 (1979), 494–526.

[Ske80] R. D. Skeel, *Iterative refinement implies numerical stability for Gaussian elimination*, Math. Comp. 35 (1980), 817–832.

[Slvd96] G. L. G. Sleijpen and H. A. van der Vorst, *A Jacobi–Davidson iteration method for linear eigenvalue problems*, SIAM J. Matrix Anal. Appl. 17 (1996), 401–425.

[Smi76] B. T. Smith et al., *Matrix Eigensystem Routines—EISPACK Guide*, Springer-Verlag, Berlin, 1976.

[SBG96] B. Smith, P. Bjørstad, and W. Gropp, *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge U. Press, Cambridge, UK, 1996.

[Smi70] F. Smithies, *Integral Equations*, Cambridge U. Press, Cambridge, UK, 1970.

[Son89] P. Sonneveld, *CGS, a fast Lanczos-type solver for nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. 10 (1989), 36–52.

[Ste73] G. W. Stewart, *Introduction to Matrix Computations*, Academic Press, New York, 1973.

[Ste77] G. W. Stewart, *On the perturbation of pseudo-inverses, projections, and linear least squares problems*, SIAM Review 19 (1977), 634–662.

[Ste93] G. W. Stewart, *On the early history of the singular value decomposition*, SIAM Review 35 (1993), 551–566.

[StSu90] G. W. Stewart and J. Sun, *Matrix Perturbation Theory*, Academic Press, Boston, 1990.

[Sti86] S. M. Stigler, *The History of Statistics*, Harvard U. Press, Cambridge, MA, 1986.

[Str86] G. Strang, *A proposal for Toeplitz matrix calculations*, Stud. Appl. Math. 74 (1986), 171–176.

[Str88] G. Strang, *Linear Algebra and Its Applications*, 3rd ed., Harcourt, Brace, and Jovanovich, San Diego, 1988.

[Str69] V. Strassen, *Gaussian elimination is not optimal*, Numer. Math. 13 (1969), 354–356.

[Sze75] G. Szegő, *Orthogonal Polynomials*, 4th ed., Amer. Math. Soc., Providence, RI, 1975.

[ToTr94] K.-C. Toh and L. N. Trefethen, *Pseudozeros of polynomials and pseudospectra of companion matrices*, Numer. Math. 68 (1994), 403–425.

[ToTr96] K.-C. Toh and L. N. Trefethen, *Computation of pseudospectra by the Arnoldi iteration*, SIAM J. Sci. Comput. 17 (1996), 1–15.

[ToTr98] K.-C. Toh and L. N. Trefethen, *The Chebyshev polynomials of a matrix*, SIAM J. Matrix Anal. Appl. 20 (1998), 400–419.

[Tre91] L. N. Trefethen, *Pseudospectra of matrices*, in D. F. Griffiths and G. A. Watson, eds., Numerical Analysis 1991, Longman Scientific and Technical, Harlow, Essex, UK, 1992, 234–266.

[Tre97] L. N. Trefethen, *Pseudospectra of linear operators*, SIAM Review 39 (1997), 383–406.

[TrSc90] L. N. Trefethen and R. S. Schreiber, *Average-case stability of Gaussian elimination*, SIAM J. Matrix Anal. Appl. 11 (1990), 335–360.

[TTRD93] L. N. Trefethen, A. E. Trefethen, S. C. Reddy, and T. A. Driscoll, *Hydrodynamic stability without eigenvalues*, Science 261 (1993), 578–584.

[TrVi98] L. N. Trefethen and D. Viswanath, *Condition numbers of random triangular matrices*, SIAM J. Matrix Anal. Appl. 19 (1998), 565–581.

[TrWe91] L. N. Trefethen and J. A. C. Weideman, *Two results on polynomial interpolation in equally spaced points*, J. Approx. Theory 65 (1991), 247–260.

[Tur48] A. M. Turing, *Rounding-off errors in matrix processes*, Quart. J. Mech. Appl. Math. 1 (1948), 287–308.

[vdS75] A. van der Sluis, *Stability of the solutions of linear least squares problems*, Numer. Math. 23 (1975), 241–254.

[vdV92] H. A. van der Vorst, *Bi-CGSTAB: A fast and smoothly convergent variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. 13 (1992), 631–644.

[Var62] R. S. Varga, *Matrix Iterative Analysis*, Prentice Hall, Englewood Cliffs, NJ, 1962.

[Wat82] D. S. Watkins, *Understanding the QR algorithm*, SIAM Review 24 (1982), 427–440.

[Wat91] D. S. Watkins, *Fundamentals of Matrix Computations*, Wiley, New York, 1991.

[Wed73] P.-Å. Wedin, *Perturbation theory for pseudo-inverses*, BIT 13 (1973), 217–232.

[Wei96] R. Weiss, *Parameter-Free Iterative Linear Solvers*, Akademie Verlag, Berlin, 1996.

[Wil61] J. H. Wilkinson, *Error analysis of direct methods of matrix inversion*, J. Assoc. Comput. Mach. 8 (1961), 281–330.

[Wil65] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, UK, 1965.

[Wri93] S. J. Wright, *A collection of problems for which Gaussian elimination with partial pivoting is unstable*, SIAM J. Sci. Comput. 14 (1993), 231–238.

Index

Afterword: Major Developments in Numerical Linear Algebra since 1997

Yuji Nakatsukasa

It cannot be stressed enough how prevalent and influential the foundational parts of classical numerical linear algebra (NLA) still are in 2022: Gaussian elimination with partial pivoting is the algorithm used for most linear systems, the QR algorithm is used to solve virtually all mid-scale eigenvalue problems, and Krylov methods are still a very popular class of algorithms. Nonetheless, since the publication of this book in 1997, a number of branches of NLA have emerged or made significant progress. Here is an attempt to give an overview of some exciting developments, along with some key references. Admittedly any such attempt is subjective and cannot cover all important contributions.

- **Randomized algorithms.** Classical algorithms in NLA are mostly *deterministic*,[1] wherein running the algorithm twice results in the exact same outcome. *Randomized algorithms*, by contrast, explicitly introduce randomness in the algorithm. This turns out to lead to a number of benefits. Among the key insights are the following:

  1. By introducing randomness, with high probability one can avoid “pathological cases” that traditional worst-case analyses need to deal with; this can result in near-optimal solutions with significantly improved speed with exponentially small failure probability [31].
  2. Random *sketching* can often reduce the problem size with minimal impact (or no impact when used to find a preconditioner) on the quality of the solution, e.g., for least-squares problems [6, 50], linear systems, and eigenvalue problems [48].
  3. Monte-Carlo type estimation is possible in many matrix approximation tasks, including matrix multiplication [21] and estimating the trace [7] or diagonals [13] of a matrix.

[1] A notable exception is the initial vector(s) in Krylov methods, especially in eigenvalue computations, which traditionally has been chosen randomly. However, the theoretical justifications and analysis employing random matrix theory have been developed more recently, which led to, e.g., the appreciation of the power of Krylov methods with large initial dimension and small step size [46, 55].

4. By introducing a random perturbation to the original problem, one can avoid pathological cases (e.g., multiple eigenvalues or eigenvalues in an unstable region) [10].

Despite being an emerging field, randomized NLA has been blessed with a number of excellent survey papers and monographs, such as those by Martinsson and Tropp [44], Mahoney [42], Kannan and Vempala [37], and Woodruff [58].

- **Continuous linear algebra.** When a matrix represents a discretization of a continuous object (e.g., a function or linear operator) in the continuous limit as the matrix dimension(s) tend to infinity, one obtains a quasimatrix (or a bivariate function). These limits are what is known as a quasimatrix or cmatrix [52] (their origins can be found in Lecture 7 of this book), and most matrix decompositions (e.g., QR, SVD and LU, Cholesky) can be shown to have a quasimatrix analogue. For many problems in scientific computing that originate from a continuous setting (e.g., function approximation and differential equation), it is a good idea to stay in the continuous space as long as possible, delaying or eliminating discretization.

  Chebfun [22] is a MATLAB toolbox that enables a variety of computations with continuous objects, including quasimatrices and cmatrices, and performs computations with them at speed comparable to that with finite-dimensional matrices. Building on the foundation of approximation theory [53], Chebfun lets users perform all sorts of operations with functions including approximation, root finding, differentiation, and integration, and more. Chebfun has grown in its capabilities and is now able to solve differential equations [54], perform multivariate approximation (2d [51] and 3d [33]), and compute with rational functions [47], among other tasks.

- **Tensors.** Tensors are high-dimensional arrays of numbers and a natural generalization of matrices (vectors are order-1 tensors, and matrices order 2). While the study of tensors has existed for many decades, their applications and practical algorithms have surged in the past decade or so. It turns out that many matrix/NLA results have no direct counterpart for tensors. (For example, one can say there are at least three tensor analogues of the matrix SVD: CP, Tucker [19], and tensor train [49] decompositions, each generalizing some, but not all, of the elegant properties of the matrix SVD.) Moreover, most tensor problems are known to be NP-hard [36]. Despite these inconvenient facts, remarkable progress has been made in the theory, algorithms and applications of tensors. Surveys and monographs on tensors include Kolda and Bader [39], Hackbusch [30], and Khoromskij [38].

- **Mixed precision arithmetic.** Modern computing hardware increasingly supports lower-precision and mixed-precision arithmetic. A compelling line of work makes use of two or more precisions to improve speed or accuracy (or even both). In some cases, one can show that by using low-precision arithmetic in some (ideally the most expensive parts) but not all of the computation, one can reduce runtime without impairing the accuracy of the solution. Conversely, by using higher precision in selected parts of the computation, one can sometimes improve accuracy without impairing the runtime. Surveys on this rapidly growing subject are given in [1, 35].

- **Communication-avoiding algorithms.** As the clock speed of individual CPUs stopped improving around 2005, making parallel computing indispensible in high performance large-scale NLA computations, it has been recognized more than ever that *communication* cost is a vital part of computation, in addition to (and sometimes more than) the standard arithmetic cost (flops). Here communication refers to the movement of data between processors and memory. A number of studies have shown that minimizing or reducing communication can lead to significant speedup in NLA computations. Comprehensive surveys and summaries on the topic (through 2014) can be found in [8, 9], and a book is in preparation as of 2022, by Ballard, Carson, Demmel, and Grigori.

- **Functions of matrices $f(A)$.** The matrix exponential has been a topic of significant interest in NLA [45], classically arising in the solution for the differential equation $\dot{x} = Ax$. Other matrix functions, such as the matrix logarithm, square root, and sign functions, have found a number of recent applications in scientific computing. Higham's book [34] gives a comprehensive account of the topic, presenting theory and (usually iterative) algorithms for computing $f(A)$. Most of these algorithms are based on approximating f on the spectrum of A using a polynomial or rational function, which is often in a composite form, that is, $f(A)$ is approximated by $r(A)$, where $f(z) \approx r(z) = r_k(\cdots r_2(r_1(z)))$ (where k is the number of compositions, which is often equal to the iteration count[2]) holds for z in (or near) the spectrum of A.

- **Nonlinear eigenvalue problems.** Generalized eigenvalue problems $Ax = \lambda Bx$ are a generalization of eigenvalue problems that have found a growing number of applications. The QZ algorithm provides a robust and backward stable solution. Further generalization is possible; the most general form is perhaps $N(\lambda)x = 0$, where N is a nonlinear matrix-valued function (e.g., when N is a polynomial, this is a

[2] This way r is a rational function whose degree grows exponentially in k.

polynomial eigenvalue problem). Polynomial eigenvalue problems can be rewritten as linear (usually generalized) eigenvalue problems, by a process called *linearization*, to which a large body of work have been devoted; a landmark contribution is [41]. When N is a non-polynomial nonlinear function, the number of eigenvalues is unknown (and can be infinite); nonetheless, a powerful and practical approach is to approximate it (locally) by a polynomial or rational function and then use linearization combined with a linear eigensolver. See Güttel and Tisseur [28] for a survey.

- **Low-rank and rank-structured matrices.** As the size of the matrices that arise in applications continues to grow at a remarkable rate, exploiting their structures has been increasingly important. Of these structures, low-rank matrices (and tensors) have seen a particular surge in interest and applications. Sometimes the data matrices themselves can be shown to be low rank [12, 56]; more commonly, a wide class of matrices can be shown to possess a hidden low-rank structure (e.g., in the off-diagonal blocks) known as *hierarchical matrices* [29]. When such structure is present, it pays off significantly to explore algorithms that take advantage of it, often leading to fast direct solvers [43]. Such ideas have been applied successfully to tensors [27].

- **Matrix manifolds.** In many problems in optimization, the solution is a matrix that lies on a prescribed, smooth manifold [2, 15, 24]. Typical examples include the Stiefel manifold (matrices with orthonormal columns) and the manifold of low-rank matrices (often with the additional positive semidefinite structure). Using algorithms that respect the structure and find iterates that stay on the manifold, one can often obtain elegant and efficient solutions for challenging problems. Manopt [16] is a popular toolbox currently offered in MATLAB, Python, and Julia that enables users to benefit from the state-of-the-art methods for optimization on manifolds.

- **Fast matrix multiplication.**[3] As mentioned in the Notes section at the end of this book, research is ongoing to reduce the exponent ω in the matrix multiplication complexity $O(n^\omega)$, where $\omega > 2$. At least in theory, the exponent and its numerical stability carry over to a host of NLA operations, including the LU and QR factorization, eigenvalue problems, and the SVD [20]. As of 2022, the best exponent is due to Alman and Williams [4], who show that $\omega < 2.37286$. Unfortunately, significant improvement of ω is impossible with the currently known

[3] This topic is not so much NLA as TCS (theoretical computer science); a compelling attempt to bridge the two fields is made by the CMC seminar series `https://sites.google.com/view/cmc-seminar`.

approaches [3, 18], and most of these "fastest" algorithms are impractical, as the hidden constant is enormous and their implementation is complicated.

- **Applications.** Application areas that crucially rely on NLA techniques continue to grow. Some of the areas that have witnessed the most remarkable advances include model order reduction [14, 17], control [11, 40], inverse problems [26, 32], and network analysis [5, 25].

Another active line of research with a long history is **sparse direct methods** [23], which can bring enormous speedups for sparse matrices over classical methods. Similarly, research on **preconditioning** continues to be a vital part of an efficient solution for many challenging problems [57]. Finally, open-source **software** packages have always been the medium with which the field has benefited scientific computing. In addition to the classical (and still widely used) LAPACK, successful projects in NLA include ScaLAPACK, FLAME, MAGMA, and PLASMA, and more are sure to come.

With these active research areas[4] and new ones to emerge, NLA will undoubtedly continue to be a vital component for the solution of most problems in scientific computing.

References

[1] A. Abdelfattah, H. Anzt, E. G. Boman, E. Carson, T. Cojean, J. Dongarra, A. Fox, M. Gates, N. J. Higham, X. S. Li, et al., *A survey of numerical linear algebra methods utilizing mixed-precision arithmetic*, Int. J. High Perform. Comput. Appl. 35 (2021), 344–369.

[2] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, Princeton, NJ, 2009.

[3] J. Alman, *Limits on the universal method for matrix multiplication*, Theory Comput. 17 (2021), 1–30.

[4] J. Alman and V. V. Williams, *A refined laser method and faster matrix multiplication*, in Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA), SIAM, Philadelphia, 2021, 522–539.

[5] F. Arrigo, P. Grindrod, D. J. Higham, and V. Noferini, *Non-backtracking walk centrality for directed networks*, J. Complex Netw. 6 (2018), 54–78.

[6] H. Avron, P. Maymounkov, and S. Toledo, *Blendenpik: Supercharging LAPACK's least-squares solver*, SIAM J. Sci. Comp. 32 (2010), 1217–1236.

[7] H. Avron and S. Toledo, *Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix*, J. ACM 58 (2011), 8.

[8] G. Ballard, E. Carson, J. Demmel, M. Hoemmen, N. Knight, and O. Schwartz, *Communication lower bounds and optimal algorithms for numerical linear algebra*, Acta Numer. 23 (2014), 1–155.

[4] An excellent collection of presentations on the latest NLA research are available at the E-NLA seminar website `https://sites.google.com/view/e-nla/home`.

[9] G. Ballard, J. Demmel, O. Holtz, and O. Schwartz, *Minimizing communication in numerical linear algebra*, SIAM J. Matrix Anal. Appl. 32 (2011), 866–901.

[10] J. Banks, J. Garza-Vargas, A. Kulkarni, and N. Srivastava, *Pseudospectral shattering, the sign function, and diagonalization in nearly matrix multiplication time*, in 2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 2020, 529–540.

[11] C. Beattie, V. Mehrmann, H. Xu, and H. Zwart, *Linear port-Hamiltonian descriptor systems*, Math. Control Signals Syst. 30 (2018), 1–27.

[12] B. Beckermann and A. Townsend, *On the singular values of matrices with displacement structure*, SIAM J. Matrix Anal. Appl. 38 (2017), 1227–1248.

[13] C. Bekas, E. Kokiopoulou, and Y. Saad, *An estimator for the diagonal of a matrix*, Appl. Numer. Math. 57 (2007), 1214–1229.

[14] P. Benner, S. Gugercin, and K. Willcox, *A survey of projection-based model reduction methods for parametric dynamical systems*, SIAM Rev. 57 (2015), 483–531.

[15] N. Boumal, *An Introduction to Optimization on Smooth Manifolds*, Available online, Aug 2020.

[16] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, *Manopt, a Matlab toolbox for optimization on manifolds*, J. Mach. Learn. Res. 15 (2014), 1455–1459.

[17] S. Chaturantabut and D. C. Sorensen, *Nonlinear model reduction via discrete empirical interpolation*, SIAM J. Sci. Comp. 32 (2010), 2737–2764.

[18] M. Christandl, P. Vrana, and J. Zuiddam, *Barriers for fast matrix multiplication from irreversibility*, Theory Comput. 17 (2021), 1–32.

[19] L. De Lathauwer, B. De Moor, and J. Vandewalle, *A multilinear singular value decomposition*, SIAM J. Matrix Anal. Appl. 21 (2000), 1253–1278.

[20] J. Demmel, I. Dumitriu, and O. Holtz, *Fast linear algebra is stable*, Numer. Math. 108 (2007), 59–91.

[21] P. Drineas, R. Kannan, and M. W. Mahoney. *Fast Monte Carlo algorithms for matrices I: Approximating matrix multiplication*, SIAM J. Comput. 36 (2006), 132–157.

[22] T. A. Driscoll, N. Hale, and L. N. Trefethen, *Chebfun Guide*, Pafnuty Publications, Oxford, 2014.

[23] I. S. Duff, A. M. Erisman, and J. K. Reid, *Direct Methods for Sparse Matrices*, Oxford University Press, Oxford, 2017.

[24] A. Edelman, T. A. Arias, and S. T. Smith, *The geometry of algorithms with orthogonality constraints*, SIAM J. Matrix Anal. Appl. 20 (1998), 303–353.

[25] E. Estrada, N. Hatano, and M. Benzi, *The physics of communicability in complex networks*, Physics Reports 514 (2012), 89–119.

[26] S. Gazzola, P. C. Hansen, and J. G. Nagy, *IR Tools: A MATLAB package of iterative regularization methods and large-scale test problems*, Numer. Algorithms 81 (2019), 773–811.

[27] L. Grasedyck, D. Kressner, and C. Tobler, *A literature survey of low-rank tensor approximation techniques*, GAMM-Mitteilungen 36 (2013), 53–78.

[28] S. Güttel and F. Tisseur, *The nonlinear eigenvalue problem*, Acta Numer. 26 (2017), 1–94.

[29] W. Hackbusch, *Hierarchical Matrices: Algorithms and Analysis*, Springer, Berlin, 2015.

[30] W. Hackbusch, *Tensor Spaces and Numerical Tensor Calculus*, 2nd edition, Springer, Berlin, 2019.

[31] N. Halko, P.-G. Martinsson, and J. A. Tropp, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev. 53 (2011), 217–288.

[32] P. C. Hansen, J. G. Nagy, and D. P. O'Leary, *Deblurring Images: Matrices, Spectra, and Filtering*, SIAM, Philadelphia, 2006.

[33] B. Hashemi and L. N. Trefethen, *Chebfun in three dimensions*, SIAM J. Sci. Comp. 39 (2017), C341–C363.

[34] N. J. Higham, *Functions of Matrices: Theory and Computation*, SIAM, Philadelphia, 2008.

[35] N. J. Higham and T. Mary, *Mixed precision algorithms in numerical linear algebra*, MIMS EPrint 2021.20, UK. To appear in Acta Numer.

[36] C. J. Hillar and L.-H. Lim, *Most tensor problems are NP-hard*, J. ACM 60 (2013), 45.

[37] R. Kannan and S. Vempala, *Randomized algorithms in numerical linear algebra*, Acta Numer. 26 (2017), 95–135,.

[38] B. N. Khoromskij, *Tensor Numerical Methods in Scientific Computing*, De Gruyter, Berlin, 2018.

[39] T. G. Kolda and B. W. Bader. *Tensor decompositions and applications*, SIAM Rev. 51 (2009), 455–500.

[40] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*, SIAM, Philadelphia, 2016.

[41] D. S. Mackey, N. Mackey, C. Mehl, and V. Mehrmann, *Vector spaces of linearizations for matrix polynomials*, SIAM J. Matrix Anal. Appl. 28 (2005), 971–1004.

[42] M. W. Mahoney, *Randomized algorithms for matrices and data*, arXiv:1104.5557, 2011.

[43] P.-G. Martinsson, *Fast Direct Solvers for Elliptic PDEs*, SIAM, 2020.

[44] P.-G. Martinsson and J. A. Tropp, *Randomized numerical linear algebra: Foundations and algorithms*, Acta Numer. 29 (2020), 403—572.

[45] C. Moler and C. Van Loan, *Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later*, SIAM Rev. 45 (2003), 3–49.

[46] C. Musco and C. Musco, Stronger and faster approximate singular value decomposition via the block Lanczos method, in *Advances in Neural Information Processing Systems (NIPS 2015)*, 2014, 14243–14253. Available at http://arXiv.org/abs/1504.05477.

[47] Y. Nakatsukasa, O. Sète, and L. N. Trefethen, *The AAA algorithm for rational approximation*, SIAM J. Sci. Comp 40 (2018), A1494–A1522.

[48] Y. Nakatsukasa and J. A. Tropp, *Fast & accurate randomized algorithms for linear systems and eigenvalue problems*, arXiv 2111.00113.

[49] I. V. Oseledets, *Tensor-train decomposition*, SIAM J. Sci. Comp. 33 (2011), 2295–2317.

[50] V. Rokhlin and M. Tygert, *A fast randomized algorithm for overdetermined linear least-squares regression*, Proc. Natl. Acad. Sci. 105 (2008), 13212–13217.

[51] A. Townsend and L. N. Trefethen, *An extension of Chebfun to two dimensions*, SIAM J. Sci. Comp 35 (2013), C495–C518.

[52] A. Townsend and L. N. Trefethen, *Continuous analogues of matrix factorizations*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 471 (2015), 20140585.

[53] L. N. Trefethen, *Approximation Theory and Approximation Practice*, SIAM, Philadelphia, 2013.

[54] L. N. Trefethen, A. Birkisson, and T. A. Driscoll, *Exploring ODEs*, SIAM, 2018.

[55] J. A. Tropp, *Randomized block Krylov methods for approximating extreme eigenvalues*, Numer. Math., 150 (2022), 217–255.

[56] M. Udell and A. Townsend, *Why are big data matrices approximately low rank?*, SIAM Journal on Mathematics of Data Science 1 (2019), 144–160.

[57] A. J. Wathen, *Preconditioning*, Acta Numer. 24 (2015), 329–376.

[58] D. P. Woodruff, *Sketching as a tool for numerical linear algebra*, Foundations and Trends® in Theoretical Computer Science 10 (2014), 1–157.

"A beautifully written textbook offering a distinctive and original treatment."
Nicholas J. Higham, University of Manchester

"Offers a rarely seen integration of computation and theory, illuminated by judiciously chosen examples."
Ilse Ipsen, North Carolina State University

"Almost the perfect text to introduce graduate students to the subject."
Daniel Szyld, Temple University

"An ideal book for a graduate course in numerical linear algebra."
Suely Oliveira, University of Iowa

Since its original appearance in 1997, *Numerical Linear Algebra* has been a leading textbook in its field, used in universities around the world. It is noted for its 40 lecture-sized short chapters and its clear and inviting style. It is reissued here with a new foreword by James Nagy and a new afterword by Yuji Nakatsukasa about subsequent developments.

Nick Trefethen is Professor of Numerical Analysis and head of the Numerical Analysis Group at Oxford University. An American, he taught previously at MIT and Cornell. He is a Fellow of the Royal Society and a former President of SIAM.

David Bau was a student of Trefethen at Cornell when this book was written and then had a career in the tech industry, notably at Microsoft and Google. Most recently he has joined the computer science faculty at Northeastern University, where he leads a research group in computer vision and machine learning.

For more information about SIAM books, journals,
conferences, memberships, or activities, contact:

siam®

Society for Industrial and Applied Mathematics
3600 Market Street, 6th Floor
Philadelphia, PA 19104-2688 USA
+1-215-382-9800
siam@siam.org • *www.siam.org*

OT181

ISBN: 978-1-611977-15-8
90000
9 781611 977158

SIAM Textbooks