from *SIAM News*, Volume 36, Number 9, November 2003

Very Early Days of Matrix Computations

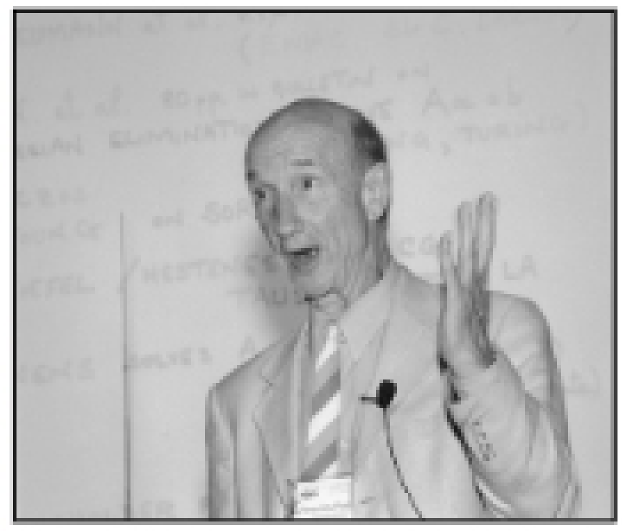
By Beresford Parlett

The following essay is based on one of four retrospective talks given in July 2002 at the SIAM 50th Anniversary Meeting in Philadelphia. The talk was prepared for a general SIAM audience and not directed only to numerical analysts and historians. Interested readers can find a much fuller picture in* A History of Scientific Computing *(edited by Stephen G. Nash and published by the ACM Press in 1990).

"Perhaps you could talk about the founding of SIAM (1952) and its relation to work on matrix computations." That was my mandate when I agreed to participate in the 50th-anniversary retrospective in Philadelphia. It took little time to find out that the founders of SIAM (the members of the first council) were not overly concerned with any of the three standard matricial computing tasks: solve a system of linear algebraic equations $Ax = b$, hunt for eigenvalues, and find a least-squares solution to an inconsistent linear system. Not one of the council members did research on these topics.

By contrast, SIAM recognized from its inception the central importance of optimization in general, and of linear programming in particular. It is puzzling that optimization and matrix computations took so long to get acquainted. The fruits of cooperation have been of great value to both communities.

The situation with matrix computations changed abruptly five years later, when the *Journal of the Society for Industrial and Applied Mathematics*—there was only one in those days—agreed to publish papers given at a conference on matrix computations organized by Wallace Givens and held at Wayne State University early in September 1957. From then on, SIAM played an active role, through meetings as well as journals, in fostering matrix computations. (I follow the lead of Gene Golub and Charlie van Loan in preferring the descriptive name "matrix computations" to the grander "numerical linear algebra," simply because linear algebra (the invariants under equivalence, congruence, and similarity) is a subset of matrix theory, and not the whole story.)



If the founders of SIAM were not overly concerned with matrix computations, Beresford Parlett pointed out at SIAM's 50th-anniversary celebration, the situation changed dramatically in 1957, with SIAM's agreement to publish papers from an early conference on the subject.

A few comments about important early work that predates 1957 and is independent of SIAM are in order. It is striking how long it took for the true worth of some algorithms to be appreciated. This must be due to the difficulties in assessing the role of finite-precision arithmetic. After all, the pioneers were all steeped in classical mathematics and had to discover for themselves what methods worked. In computational fluid dynamics it is routine practice to ignore roundoff error, on the assumption that it will be negligible in comparison with various approximation errors. In some matrix algorithms, roundoff error is the only error; many of these errors can be ignored, while others may be devastating. In the 1950s and 1960s, the experts were effectively all learning the subject together, particularly at meetings. If you permit only t digits for any "number," you are bound to have uncertainty in any nontrivial calculation.

The Direct Solution of $Ax = b$

During World War II a computer was a human being, usually female; several of them would work together (a foretaste of parallel processing?) to solve systems of difference equations derived from differential equations.

In 1943 Harold Hotelling published an analysis showing that the error in an approximate inverse computed via triangular factorization might grow like 4^{n-1}, where n is the order of the matrix. Alan Turing was making similar analyses informally in England. Not a promising omen for the future of direct methods that factor matrices.

By 1946 the ENIAC was up and running in Philadelphia. In England during the same period, Jim Wilkinson was assisting Turing in the development of a prototype for the ACE (Automatic Computing Engine) at the National Physical Laboratory. Leslie Fox recalled spending two weeks in 1946 with Goodwin, Turing, and Wilkinson, using desk calculators to solve an 18×18 linear system $Ax = b$. This chore was a blessing in disguise for Wilkinson.

In the United States, anticipating the advent of rapid, non-human computing machines, John von Neumann had already turned his attention to both the eigenvalue problem and the solution of $Ax = b$ by direct methods as part of his larger campaign to harness these computers for fluid dynamics computations. In 1946, von Neumann, F. Murray, and H.H. Goldstine produced a paper in which they rediscovered Jacobi's method (1846) for reducing a real symmetric matrix to diagonal form by repeatedly annihilating the largest off-diagonal pair of entries. One year later Goldstine and von Neumann published a 78-page paper, "Numerical Inverting of Matrices of High Order" (*Bulletin of the American Mathematical Society*), that corrected to some extent Hotelling's gloomy analysis of elimination.

In the view of some scholars, the appearance of this landmark paper marks the birth of modern numerical analysis. Here we find explicit use of matrix norms to measure error and the introduction of the condition number of an invertible matrix. Some nice use of "backward" error analysis occurs in the paper, but it is incidental. There is a subtle difficulty here. The success of Wilkinson's approach (described below) does not apply to inverses computed from a factorization $A = LU$, although it does apply to the solution of $Ax = b$ computed from that same factorization. If only the paper had focused on the computed solution of $Ax = b$ rather than on the computed inverse of A! The authors did show that the dreaded element growth cannot occur in the Cholesky factorization of a symmetric positive-definite matrix A. This prompted them to recommend the use of Cholesky on the normal equations $A'Ax = A'b$ when n exceeds 15. This was bad, although sensible, advice. Recall that it is unlikely that either of these formidable mathematicians had ever solved a general 5×5 system of linear equations himself.

How could someone as brilliant as von Neumann think hard about a subject as mundane as triangular factorization of an invertible matrix and not perceive that, with suitable pivoting, the results are impressively good?

Jim Wilkinson's 1946 chore at NPL with the desk calculator, along with examples computed on the ACE, and then from 1951 onward on its successor DEUCE, convinced him that Gaussian elimination with pivoting was an excellent algorithm even on unsymmetric matrices. He gradually came to recognize the salient issues, although the first edition of NPL's *Modern Computing Methods* (1957) contains no hint of backward analysis. That should give us food for thought.

The paper that established Wilkinson's reputation and corrected the misjudgments of Hotelling, Turing, and von Neumann did not appear until 1961, in the *Journal of the ACM*, under the title "Error Analysis of Direct Methods of Matrix Inversion." That late date puts the paper beyond the scope of this essay.

(Briefly, Wilkinson proved that if element growth is modest when A is factored into LU, then the computed solution z to $Ax = b$ exactly solves a system very close to $Ax = b$. This result was a dramatic triumph for a backward error analysis, and Wilkinson went on to exploit that approach with great success for the rest of his career.

One of the vital contributions of the 1961 paper was to make error analysis comprehensible, whereas earlier efforts made such work seem both difficult and boring. Wilkinson's analyses used the familiar context of exact arithmetic to exhibit exact relations among the computed quantities; in this way he avoided the horrible pseudo-operations that some earlier scholars felt obliged to introduce.)

Krylov Subspace Methods

In Russia in 1931, Krylov used the relation between the vectors $b, Ab, A^2b, \ldots$ to compute the coefficients of A's characteristic polynomial. We do not do that today, but his name now connotes any method that uses the space spanned by vectors generated by the power method with some initial vector b.

I am grateful to Augustin Dubrulle for alerting me to the 1941 dissertation of Karl Hessenberg on the unsymmetric eigenvalue problem; in it, he introduced a sequential way to build up an almost triangular matrix that is similar to a given full matrix. "Almost" here means that the next-to-diagonal entries will not, in general, be zero. Today, we call this type of matrix the Hessenberg form and refer to the procedure as a Krylov subspace method. Hessenberg was German and worked as an engineer during World War II.

Other seminal ideas and algorithms appeared in the early 1950s. Their ultimate value was not appreciated initially. As with Gaussian elimination, the effect of roundoff error is subtle, and comprehension comes only after varied experience and failures. It is the twists and turns in judgment over the years that reveal this area as a challenging one in which to work.

In 1950 Cornelius Lanczos published the algorithm that bears his name. For a symmetric matrix A the procedure builds up an orthogonal matrix V that reduces A to tridiagonal form $T = V'AV$ in a modest number of steps. The compact form of T can be exploited in the calculation of its eigenvalues and eigenvectors, which are easily related to those of T. Lanczos also described an extension of the algorithm to unsymmetric matrices and reported that the significant effects of roundoff error could be mitigated to some extent by a process called full reorthogonalization.

At about the same time, Magnus Hestenes in Los Angeles and Eduard Stiefel at ETH, Zurich, were developing the conjugate gradient algorithm for solving symmetric positive-definite matrices. This rival to the Cholesky factorization had the attraction of not modifying the given matrix A. These three men (L, H, and S) met in 1951 at the Institute for Numerical Analysis in Los Angeles, at a small meeting organized by Olga Taussky. What an encounter that must have been!

It did not take these pioneers long to see the connections between the conjugate gradient method and the Lanczos algorithm in exact arithmetic. The famous paper on conjugate gradients was not published until 1952, the year of SIAM's birth. That meeting (sometimes called Gatlinburg$_{-1}$) saw the birth of what are today called Krylov subspace methods. The story of the ups and downs of these methods is too long for this short essay. Very briefly: They flopped as direct methods and were brought back to life as effective iterative procedures only when coupled with some form of preconditioning.

Other Early Landmarks

A survey of the topic of iterative methods for solving $Ax = b$ cannot fail to mention that in 1950 David Young was just finishing his dissertation at Harvard under Garrett Birkhoff. Young gave a rigorous general analysis of the convergence of a much used class of methods that change one unknown at a time and do not alter the matrix at all. In fact, the matrix may be only implicit in a set of difference equations derived from some elliptic differential equation. Unlike direct methods, these relaxation methods work only on special equations. For SOR (successive over-relaxation), Young characterized the optimal relaxation parameter for certain relevant matrix classes. The published account did not appear until 1954.

It was in 1954 that Wallace Givens produced the first proven "backward" stable solution to the symmetric eigenvalue problem. Step 1 reduces the given matrix A to tridiagonal form T, using a sequence of "plane rotations." A plane rotation changes only two rows and columns in the matrix while the eigenvalues remain unchanged. Step 2 computes the eigenvalues of T by evaluating the sequence of leading principal minors of $T - xI$ for various values of the shift x. The sequence tells us the number of eigenvalues greater than x. The shifts are chosen by the bisection technique applied to any interval that contains one or more eigenvalues.

Givens performed an explicit backward error analysis, in fixed-point arithmetic, of these two phases. He exhibited a matrix close to the given matrix A for which the computed eigenvalues are the exact eigenvalues. In other words, his method computes the eigenvalues of $A + E$, but the backward error E is very small. For symmetric matrices the error in a computed eigenvalue cannot exceed the norm of E. Finally, Givens's step 3 computes eigenvectors by inverse iteration, using the computed eigenvalue as the shift for T; this is the only unsatisfactory part of the procedure. The technical report was never published. Nevertheless, this new method eclipsed the Lanczos algorithm until the latter was seen in a new light as a way to compute a small number of eigenpairs of a large sparse symmetric matrix.

On another front, 1954 saw IBM's introduction of the programming language Fortran I. This exciting development ushered in the new area of mathematical software and the possibility of sharing programs among different computers and thus avoiding much duplication of effort. The much better language ALGOL, developed in about 1957 in Switzerland, found little support outside academia. Perhaps academics were too much in control to be effective in the marketplace?

Inevitably, the impact of mathematical software was strong in many areas of applied mathematics, and especially so in matrix computations. The experts, led by Wilkinson and Reinsch, collaborated to produce and maintain libraries of programs that encode the best available methods; this activity continues to this day. To everyone's surprise, it turned out to be really difficult to test and properly document codes that would work well, not just correctly, on all the important computer architectures. That is a long story. Through a journal and prizes, such as the one offered by Gordon Bell and the Wilkinson Prize for Numerical Software (see "Hooked on Meshing, Researcher Creates Award-Winning Triangulation Program" for details on the 2003 Wilkinson Prize recipient), some effort has been made to recognize excellence in mathematical software. Until the 1990s there was no academic reward for fine work in this field.

Depth versus Subtlety

How could someone as brilliant as von Neumann think hard about a subject as mundane as triangular factorization of an invertible matrix and not perceive that, with suitable pivoting, the results are impressively good? Partial answers can be suggested—lack of hands-on experience, concentration on the inverse rather than on the solution of $Ax = b$—but I do not find them adequate. Why did Wilkinson keep the QR algorithm as a backup to a Laguerre-based method for the unsymmetric eigenproblem for at least two years after the appearance of QR? Why did more than 20 years pass before the properties of the Lanczos algorithm were understood? I believe that the explanation must involve the impediments to comprehension of the effects of finite-precision arithmetic.

My suggestion is that the difficulties in matrix computations may not be *deep* but they are *subtle*. Let me illustrate my use of these two adjectives:

The fact that, for a smooth function f, $\lim(h \to 0)[f(x + h) - f(x)]/h = f'(x)$ is deep, but the fact that, for small enough h, the computed divided difference $[f(x + h) - f(x)]/h$ vanishes is subtle.

The fundamental theorem of algebra is deep, but the "region of confusion" (all values x such that the error in evaluating polynomial p at x exceeds $|p(x)|$) that surrounds each zero of p is a subtle concept.

The implications of the failure of the associative law for addition in the computer are pervasive and subtle. The failure of that law for multiplication is less serious, except for the occurrence of unnecessary over/underflows.

Lest some readers regard my suggestion of subtlety versus depth as damning the field of matrix computations with faint praise, I conclude by citing two great thinkers who held subtlety in high regard.

Albert Einstein (1879–1955): "Subtle is the Lord, but not malicious."

Francis Bacon (1561–1626): "Histories make men wise; poets witty; the mathematics subtile; natural philosophy deep; moral grave; logic and rhetoric able to contend."

Beresford Parlett is a professor emeritus of mathematics at the University of California, Berkeley.