The Matrix Sign Function and Riccati Equations

Prof. Alan J. Laub
Dept. of Electrical and Computer Engineering
University of California
Santa Barbara, CA 93106-9560
laub@ece.ucsb.edu

Abstract

An overview is given of recent research (jointly with Charles Kenney and others) on the solution of large-scale algebraic Riccati equations by means of iterative algorithms for computation of the matrix sign function. Riccati equations, which lie at the heart of control theory, present many challenging computational problems. Riccati algorithms based on the matrix sign function have shown considerable promise for implementation on modern parallel and vector computing machines.

A new family of iterative algorithms, which includes the classical Newton iteration as a special case, is described. Other topics outlined are condition of the matrix sign function, scaling strategies for the iterations, and some error analysis, including a discussion of the possibility of chaotic behavior.

Analysis of the Recursive Least Squares Lattice Algorithm

by
Richard C. LeBorne[1] and James R. Bunch
University of California, San Diego

Fast Recursive Least Squares Lattice (fast RLSL) algorithms have enjoyed increasing popularity in recent years in fields ranging from biomedical engineering and communications to control, radar, sonar, and seismology. To address concerns of divergence with some of these implementations, past analyses have focused on the convergence and stability properties of these fast filters. Direct and Indirect updating as well as the a priori and a posteriori for the normalized and unnormalized versions of the fast RLSL have been studied comparatively for accuracy and stability. Traditionally, such analyses have concentrated on some part of the overall algorithm; for example, the analysis might concentrate on convergence in time for one filter stage. Alternatively, one other analysis might study the statistical properties of error effects for different filter stages for large values of the time index. Here, improved error estimates are obtained by combining the two algorithm recursion directions; time and order. Specifically, recursions describing the effects of roundoff errors are given from which insight regarding filter error propagation is gained.

Conditions are given which guarantee that the computed forward and backward prediction residuals at the updated stage are the exact solutions to a perturbed problem near the original one. This is of central importance since the sequence of backward prediction residuals defines the orthogonalizing process inherent to the adaptive lattice filters. The relative error defined with the infinity norm is shown to be bounded by the condition number of a 2x2 matrix of forward and backward reflection coefficients as well as the perturbations in the prediction residuals and reflection coefficients.

[1] Speaker

Title and Abstract

Author: Steven L. Lee
Email: slee@cs.uiuc.edu
Title: A new and sharp upper bound for departure from normality

The nonnormality of a matrix adversely affects the convergence behavior of Krylov subspace methods and the accuracy of eigenvalue estimation methods. The departure from normality of large matrices is impractical to compute if the eigenvalues are unknown.
A new and practical formula for computing sharp upper bounds for departure from normality is presented.

On Eigenvalues of Rayleigh Quotient Matrix Pencils of a Definite Pencil

Ren-Cang Li
Department of Mathematics
University of California at Berkeley
Berkeley, California 94720

June 14, 1992

Extended Abstract

Let $A - \lambda B$ be a definite matrix pencil of order n, i.e., both A and B are $n \times n$ Hermitian and

$$c(A,B) \stackrel{\text{def}}{=} \min_{x \in \mathbf{C}^n, \|x\|_2 = 1} |x^H(A + iB)x| > 0.$$

Suppose Y is an $n \times \ell$ matrix with full column rank whose column vectors span an approximate invariant subspace for $A - \lambda B$. This paper investigates the relation between eigenvalues of $A - \lambda B$ and those of $Y^H AY - \lambda Y^H BY$, which is termed the ***Rayleigh Quotient Matrix Pencil*** of $A - \lambda B$ with respect to Y. Our result for the spectral norm improves Sun's (*Linear Algebra Applic.*, 139:253-267 (1990)). We also present a bound in Frobenius norm which is new.

For two nonzero number pairs (α, β) and $(\widetilde{\alpha}, \widetilde{\beta})$, the *chordal distance* will be used throughout:

$$\rho((\alpha,\beta),(\widetilde{\alpha},\widetilde{\beta})) \stackrel{\text{def}}{=} \frac{|\alpha\widetilde{\beta} - \beta\widetilde{\alpha}|}{\sqrt{|\alpha|^2 + |\beta|^2}\sqrt{|\widetilde{\alpha}|^2 + |\widetilde{\beta}|^2}}.$$

For two ℓ-dimensional subspace $\mathcal{X}_1$ and $\widetilde{\mathcal{X}}_1$ spanned by the column vectors of X_1 and $\widetilde{X}_1$, both having full column rank, respectively, the distance between them will be measured by $\|\sin\Theta(X_1, \widetilde{X}_1)\|$ where $\|\cdot\|$ is a matrix norm and

$$\Theta(X_1, \widetilde{X}_1) \stackrel{\text{def}}{=} \arccos(\widetilde{X}_{10}^H X_{10} X_{10}^H \widetilde{X}_{10})^{1/2} \geq 0,$$

where $X_{10} = X_1(X_1^H X_1)^{-1/2}$, $\widetilde{X}_{10} = \widetilde{X}_1(\widetilde{X}_1^H \widetilde{X}_1)^{-1/2}$. It has been proved that if $(X_1, X_2)^{-1} = \begin{pmatrix} W_1^H \\ W_2^H \end{pmatrix}$, where X_2, $W_2 \in \mathbf{C}^{n \times (n-\ell)}$ and $W_1 \in \mathbf{C}^{n \times \ell}$, then

$$\rho_p(\mathcal{X}_1, \widetilde{\mathcal{X}}_1) \stackrel{\text{def}}{=} \|\sin\Theta(X_1, \widetilde{X}_1)\|_p = \|(W_2^H W_2)^{-1/2} W_2^H \widetilde{X}_{10}^H\|_p \qquad (1)$$

for $p = 2, F$.

Our main results are the following:

Theorem 1 *Let $A - \lambda B \in \mathbf{D}(n)$ with the generalized eigenvalues set $\{(\alpha_j, \beta_j),\ j = 1, \cdots, n\}$, and let $\mathcal{X}_1$ be the eigenspace of $A - \lambda B$, spanned by the column vectors of X_{10}, associated with $\{(\alpha_j, \beta_j),\ j = 1, \cdots, \ell\}$, where $X_{10} \in \mathbf{C}^{n \times \ell}$ and $X_{10}^H X_{10} = I$. Assume $\widetilde{\mathcal{X}}_1$ is an approximate eigenspace of $A - \lambda B$ spanned by the column vectors of $\widetilde{X}_{10}$ such that*

$$\eta \stackrel{\text{def}}{=} \frac{\|(A,B)\|_2}{c(A,B)} \rho_2(\mathcal{X}_1, \widetilde{\mathcal{X}}_1) < 1, \qquad (2)$$

where $\widetilde{X}_{10} \in \mathbf{C}^{n \times \ell}$ and $\widetilde{X}_{10}^H \widetilde{X}_{10} = I$. Let $(\widetilde{\alpha}_j, \widetilde{\beta}_j)$, $j = 1, \cdots, \ell$ be the generalized eigenvalues of $\widetilde{X}_{10}^H A \widetilde{X}_{10} i - \lambda \widetilde{X}_{10}^H B \widetilde{X}_{10}$. Then there is a permutation τ of $\{1, \cdots, \ell\}$ such that

$$\max_{1 \leq j \leq \ell} \rho((\alpha_j, \beta_j), (\widetilde{\alpha}_{\tau(j)}, \widetilde{\beta}_{\tau(j)})) \leq \max_{\substack{1 \leq i \leq \ell \\ \ell+1 \leq j \leq n}} \rho((\alpha_i, \beta_i), (\alpha_j, \beta_j)) \cdot \eta^2. \qquad (3)$$

Theorem 1 improves Sun's theorem (*Linear Algebra Applic.*, 139:253-267 (1990)) in two aspects:

- Instead of (2), Sun assumes

$$\frac{\max\{\sqrt{\|(A_1, B_1)\|_2}, 1\}}{\min\{\sqrt{\lambda_{\min}(A_1^2 + B_1^2)}, 1\}} \cdot \frac{\|(A,B)\|_2}{c(A,B)} \rho_2(\mathcal{X}_1, \widetilde{\mathcal{X}}_1) < 1,$$

  which is stronger than our assumption (2) and where $\lambda_{\min}(\cdot)$ denotes the smallest eigenvalue of a Hermitian matrix; $A_1 = X_{10}^H A X_{10}$ and $B_1 = X_{10}^H B X_{10}$.

- Our inequality (3) improves Sun's by a factor $\frac{\|(A,B)\|_2}{c(A,B)}$.

Theorem 2 *Under the conditions of Theorem 1. There is a permutation ω of $\{1,\cdots,\ell\}$ such that*

$$\sqrt{\sum_{j=1}^{\ell} \rho\big((\alpha_j,\beta_j),(\widetilde{\alpha}_{\omega(j)},\widetilde{\beta}_{\omega(j)})\big)} \leq \max_{\substack{1\leq i\leq \ell\\ \ell+1\leq j\leq n}} \rho\big((\alpha_i,\beta_i),(\alpha_j,\beta_j)\big)\frac{\|(A,B)\|_2}{c(A,B)}\cdot\frac{\delta}{\sqrt{1-\eta^2}}, \qquad (4)$$

where $\delta = \left(\frac{\|(A,B)\|_2}{c(A,B)}\right)^2 \rho_2(\mathcal{X}_1,\widetilde{\mathcal{X}}_1)\rho_F(\mathcal{X}_1,\widetilde{\mathcal{X}}_1)$, and η is define by (2).

An Easily-Updatable
Approximate Generalized Singular Value Decomposition

FRANKLIN T. LUK

Department of Computer Science
Rensselaer Polytechnic Institute
Troy, New York 12180, USA

Abstract

A recurring matrix problem in signal processing concerns generalized eigenvalues:

$$A^H Ax = \lambda B^H Bx,$$

where the matrix B has full column rank. Often, the generalized eigenvalues, call them d_j^2's, satisfy this property:

$$d_1^2 \geq d_2^2 \geq \cdots \geq d_{p-k}^2 >> d_{p-k+1}^2 \approx \cdots \approx d_p^2 . \qquad (1)$$

The k-dimensional subspace spanned by the eigenvectors corresponding to the k smallest generalized eigenvalues is called the noise subspace. We are interested in the problem of *computing an orthonormal basis for the noise subspace.*

This problem has a known solution for the *special* case where $B = I_p$, where I_p denotes a $p \times p$ identity matrix. Compute a singular value decomposition (SVD) of A:

$$A = UD_AV^H,$$

where U is $n \times p$ and orthonormal, i.e., $U^H U = I_p$, V is $p \times p$ and unitary, D_A is diagonal and $D_A = \text{diag}(d_1, \ldots, d_p)$. From (1) we get that that the desired orthonormal basis is given by the last k columns of V. However, the SVD is not amenable to efficient updating when a new row is added to A. A clever procedure was devised by Stewart in the form of the ULV decomposition (ULVD):

$$A = UL_AV^H,$$

where U is orthonormal and V unitary as in the SVD, but the middle matrix L_A is lower triangular and essentially block diagonal. In particular,

$$L_A = \begin{pmatrix} \check{L}_A & 0 \\ E & K \end{pmatrix}, \qquad (2)$$

where

(i) $\check{L}_A$ and K are lower triangular and $\check{L}_A$ is $(p-k) \times (p-k)$;

(ii) $\sigma_{\min}(\check{L}_A) \approx d_{p-k}$ and $\|E\|_F^2 + \|K\|_F^2 \approx d_{p-k+1}^2 + \cdots + d_p^2$.

Essentially, Stewart showed that to separate out the noise subspace from the signal subspace, it suffices to reduce A to the 2×2 block lower triangular form L_A, where both E and K are very small in norm. The last k columns of V then provide an orthonormal basis for the noise subspace.

In this talk we consider the noise subspace problem for the *general* case where $B \neq I_p$. First, the problem may be solved via the generalized SVD (GSVD):

$$A = U_A D_A L V^H \quad \text{and} \quad B = U_B L V^H,$$

where U_A is $n \times p$ and orthonormal, U_B is $m \times p$ and orthonormal, V is $p \times p$ and unitary, L is $p \times p$ and lower triangular, and $D_A = \text{diag}(d_1, \ldots, d_p)$. If the generalized singular values d_j's satisfy (1), then the last k columns of V provide a basis for the noise subspace. We propose here a *generalized ULVD* (ULLVD):

$$A = U_A L_A L V^H \quad \text{and} \quad B = U_B L V^H, \qquad (3)$$

where U_A, U_B, V and L are just as in the GSVD. The new middle matrix L_A has the same form as in (2) and the desired orthonormal basis is given by the last k columns of V.

An important advantage of our new decomposition lies in updating. As in signal processing, we incorporate a forgetting factor β, where $0 < \beta \leq 1$. Assume that we are given an $n \times p$ matrix $A^{(n)}$, an $m \times p$ matrix $B^{(m)}$, and an ULLVD of these two matrices. Define

$$A^{(n+1)} = \begin{pmatrix} \beta A^{(n)} \\ x_{n+1}^T \end{pmatrix} \quad \text{and} \quad B^{(m+1)} = \begin{pmatrix} \beta B^{(m)} \\ y_{m+1}^T \end{pmatrix},$$

where x_{n+1}^T denotes a new row for A, and y_{m+1}^T a new row for B. We will show how to update the ULLVD in only $O(p^2)$ operations when a new row is added to either $A^{(n)}$ or $B^{(m)}$, i.e., how to quickly compute an ULLVD of either the pair $A^{(n+1)}$ and $B^{(m)}$ or the pair $A^{(n)}$ and $B^{(m+1)}$.

This work is joint with Sanzheng Qiao of McMaster University, Canada.

Eigenvalue Perturbation Theory and Stability of Hamiltonian Systems

John H. Maddocks and Michael L. Overton

Extended Abstract for Householder Symposium, 1993

An autonomous Hamiltonian system of ordinary differential equations is of the form

$$\dot{x} = J\,\nabla H(x). \tag{1}$$

Here ∇H denotes the gradient of the Hamiltonian $H(x)$ with respect to the variable x, the matrix J is skew-symmetric, and solutions $x(t)$ of (1) are curves in phase space. In the classical setting

$$J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}, \quad \textit{and} \quad x(t) \in \Re^{2n}, \tag{2}$$

i.e. J is the standard skew matrix and the phase space is $\Re^{2n}$, with n degrees of freedom. Because J is nonsingular, the equilibrium solutions of (1), i.e. trajectories satisfying $\dot{x}_e(t) = 0$, are precisely the critical points of the Hamiltonian, i.e. those points in phase space satisfying

$$\nabla H(x_e) = 0. \tag{3}$$

At an equilibrium point x_e there are two eigenvalue problems bearing upon the stability of the dynamical system. The first is a nonsymmetric problem obtained by linearizing the dynamics (1) and separating out time:

$$JLu = \lambda u, \tag{4}$$

where $L = \nabla^2 H(x_e)$. Because of the special structure of the matrix JL, which is the product of a skew-symmetric and a symmetric matrix, it is easy to show that the eigenvalues of (4) have four-fold symmetry in the complex plane, i.e. they are symmetric with respect to both the real and imaginary axes. If they are all imaginary, the system is said to be linearly stable; otherwise, the system is unstable. The second eigenvalue problem is a real symmetric problem associated with the second variation of the Hamiltonian at x_e:

$$Lv = \mu v. \tag{5}$$

The question of interest, which was investigated by Krein and others as early as the 1950's, is whether an analysis of (5) can provide the information required for determining (linearized dynamic) stability, i.e. whether or not there is an eigenvalue of (4) in the right half-plane. It is easy to show that when (5) has only positive eigenvalues, (4) has only pure imaginary eigenvalues. However it is possible that (4) has only pure imaginary eigenvalues while (5) has negative eigenvalues, which allows the possibility that linearized stability may occur even at critical points of the Hamiltonian which are not minima. Thus, there is not a sharp correspondence between the two problems.

We show that a sharper correspondence is possible if we modify the original problem to include some damping. One way to do this is to consider dynamics of the form

$$\dot{x} = (J - \epsilon D)\nabla H(x), \tag{6}$$

where D is a positive definite matrix and $\epsilon > 0$. Then the linearized dynamic eigenvalue problem becomes

$$(J - \epsilon D)Lu = \lambda u. \tag{7}$$

We show that, for ϵ sufficiently small, *positive (negative) eigenvalues of (5) correspond to eigenvalues of (7) in the left (right) half-plane.* Thus, *stable* equilibria are *minima* of the Hamiltonian. Furthermore, the following results describe the limiting process as $\epsilon \to 0$. Suppose that all eigenvalues of (4) are imaginary and none are zero. Let λ be an eigenvalue of multiplicity m, and let Z be an $n \times m$ matrix whose columns span the corresponding invariant subspace.

Theorem 1. *Suppose λ is semisimple, so that the columns of Z are all eigenvectors. Then Z can be chosen so that*

$$Z^* LZ = K = \begin{bmatrix} I & 0 \\ 0 & -I \end{bmatrix},$$

(the identity blocks need not be the same size), with the eigenvalues of (7) associated with λ given by

$$\lambda + \xi_k\, \epsilon + o(\epsilon), \tag{8}$$

where ξ_k, $k = 1, \ldots, m$, are the eigenvalues of $M = -KZ^*LDLZ$.

Noting the positive definite factor in M, we conclude that the derivatives of the eigenvalues (8) with respect to ϵ are real, with signs determined by the sign pattern in K, which also defines the inertia of Z^*LZ.

Theorem 2. *Suppose λ is nonderogatory, so that only one vector in the invariant subspace, say z, is an eigenvector. Then Z can be chosen so that its first column is z and*

$$Z^* LZ = \kappa \begin{bmatrix} & & & \sigma \\ & & -\sigma & \\ & \sigma & & \\ \cdot & & & \end{bmatrix}$$

where

$$\sigma = \left\{ \begin{array}{ll} 1 & m \bmod 4 = 1 \\ i & m \bmod 4 = 2 \\ -1 & m \bmod 4 = 3 \\ -i & m \bmod 4 = 0 \end{array} \right\}, \quad i = \sqrt{-1}$$

with the eigenvalues of (7) associated with λ given by

$$\lambda + \psi_k \, \epsilon^{1/m} + o(\epsilon^{1/m}), \tag{9}$$

where

$$\psi_k = \rho^{1/m} \exp \frac{(\theta + 2(k-1)\pi)i}{m}, \quad k = 1, \dots, m$$

and $\rho e^{i\theta}$ is the polar form for $-\kappa\overline{\sigma} z^* LDLz$.

In this case, we see that the inertia of $Z^* LZ$ is highly restricted. If m is even, half of its eigenvalues are positive and half are negative; if m is odd, the sign of the extra eigenvalue is determined by σ and κ. Furthermore, the eigenvalues (9) have a nonlipschitz dependence on ϵ, splitting from λ at angles of $2\pi/m$, with half of them in the right half-plane and half in the left if m is even. If m is odd, the behavior of the extra eigenvalue is again determined by σ and κ.

If λ is simple, i.e. $m = 1$, both Theorems 1 and 2 reduce to precisely the following: the corresponding eigenvalue of (7) is

$$\lambda - \kappa z^* LDLz \, \epsilon + o(\epsilon)$$

where z, the eigenvector of (7) corresponding to λ, is normalized so that $z^* Lz = \kappa = \pm 1$. The sign κ is sometimes known as the Krein signature of the eigenvalue.

The proof of these results uses two powerful branches of matrix theory: analytic perturbation theory for eigenvalues, and the theory of indefinite inner products. Extension to the general derogatory, defective case is under study.

Reducing the number of floating point operations in the Jacobi method

Walter F. Mascarenhas
State University of Campinas
Campinas S.P., Brazil

February 8, 1993

We present a simple new strategy for reducing the number of floating point operations (flops) required by the classical Jacobi method for finding eigenvalues of symmetric matrices. Similar ideas can be applied to one sided Jacobi methods. We assume that the reader is familiar with the Jacobi method. Our strategy is based on the fact that, for matrices of the same size, one sweep of the Jacobi method requires twice as many flops as matrix multiplication. In order to describe our strategy we need the concept of *principal submatrix*. We say that a $k \times k$ matrix B is principal submatrix of a $n \times n$ matrix A if B can be obtained by choosing a set $D \in \{1, \ldots, n\}$ with $n - k$ indices and deleting the rows and columns of A with index in D. Our strategy consists in roughly halfing the operation count for the Jacobi method by decomposing the matrix in principal submatrices, accumulating the rotations with pivots in these submatrices and using matrix multiplication to apply these rotations to the rest of the matrix. By analogy with the traditional block Jacobi methods, we call such strategy a *submatrix Jacobi method*. We discuss how to implement submatrix Jacobi methods in serial and vector machines and present experimental results about their performance and accuracy.

The Stability of Parallel Prefix Matrix Multiplication

—

With Applications to Tridiagonal Matrices

Roy Mathias*

Many algorithms can be reduced to the problem of computing the partial products

$$M_1,\ M_1M_2, \ldots,\ M_1M_2\cdots M_n \tag{1}$$

where the $M_i, i = 1, \ldots, n$ are 2×2 matrices. We will define

$$M_{i:j} = M_iM_{i+1}\cdots M_j, \quad i \le j.$$

The n products in (1) can be computed by parallel prefix in time $O(\log_2 n)$ in parallel using $O(n)$ processors. Although this technique has been known for some time there has been little work done on its stability. In general one is interested in the components of $M_{i:j}$ rather than the matrix itself so we require component-wise bounds.

Let ϵ denote the arithmetic precision. We do not assume the use of a guard digit, nor would it's use allow us to strengthen the results. Let the eigenvalues of a symmetric matrix be ordered in decreasing order; i.e., $\lambda_i(A) \ge \lambda_{i+1}(A)$.

1 General bounds

Let $\hat{M}_{i:j}$ denote the value of $M_{i:j}$ computed by parallel prefix in floating point arithmetic. We show that there are indices $i_j < j < k_k, j = 1, \ldots, n-1$ such that

$$|\hat{M}_{1:n} - M_{1:n}| \le 2\epsilon \sum_{j=1}^{n-1} |M_{1:i_j}|\,|M_{i_j+1:j}|\,|M_{j+1:k_j}|\,|M_{k_j+1:n}| \ + O(\epsilon^2) \tag{2}$$

and for $r = 1, \ldots, n-1$

$$|\hat{M}_{1:r} - M_{1:r}| \le 2\epsilon \sum_{j=1}^{r-1} |M_{1:i_j}|\,|M_{i_j+1:j}|\,|M_{j+1:k_j^{(r)}}|\,|M_{k_j^{(r)}+1:r}| \ + O(\epsilon^2) \tag{3}$$

where $k_j^{(r)} = \min\{r, k_j\}$. The additional cost of computing the bound (2) is about the same as that of computing the $M_{1:j}, j = 1, \ldots, n$. But if we want to compute the bounds in (3) for *all* $r = 1, \ldots, n-1$ using $O(n)$ processors it takes longer than $O(\log_2 n)$ in general. Never-the-less there are some real situations where the bound (2) gives a bound on the error in $M_{1:r}, r = 1, \ldots, n-1$ at no extra cost and we present these in the paper.

2 Leading Minors of Symmetric Tridiagonal Matrices

Let T be a symmetric tridiagonal matrix, let $d_0 = 1$ and let $d_i, i = 1, \ldots, n$ be the leading principal minors of T. It is well known that the number of sign changes in the sequence $d_0, d_1, \ldots, d_n$ is equal to the number of negative eigenvalues of T. By In computing the eigenvalues of T by bisection one computes the d_i's associated with $T - \lambda I$ for various values of λ. So parallel prefix allows one to parallelize this bisection algorithm.

It can be shown that the d_i can be computed in time $O(\log_2 n)$ in parallel by parallel prefix. Determining the accuracy of the d_i's computed by parallel prefix is important in determining the accuracy of the eigenvalues computed by bisection with parallel prefix. This open problem was raised by Demmel in [1].

Let $\hat{d}_i$ denote the computed value of d_i. We show that if T is positive definite and if we use parallel prefix then

$$\left|\frac{d_r - \hat{d}_r}{d_r}\right| \ \le \ \frac{128(r-1)\epsilon}{\lambda_n(A)\lambda_{n-1}(A)\lambda_{n-2}(A)} + O(\epsilon^2), \quad r = 1, \ldots, n. \tag{4}$$

where $A = DTD$ and D is a diagonal matrix chosen so that A has main diagonal entries equal to 1. The corresponding bound if the $\hat{d}_i$'s are computed serially is

$$\left|\frac{d_r - \hat{d}_r}{d_r}\right| \ \le \ \frac{4r\epsilon}{\lambda_n(A)} + O(\epsilon^2). \tag{5}$$

We present examples for which the bound (4) is attained. Thus one can see that parallel prefix can be considerably less accurate than serial computation of the d_i's. This means that the eigenvalues computed by bisection with parallel prefix can be rather inaccurate. For example, we present a 16×16 positive definite matrix T with $\kappa(T) \approx 10^6$ for which bisection with parallel prefix computes a negative eigenvalue when the computations are done with arithmetic precision $\epsilon \approx 2 \times 10^{-16}$. That is, the relative error in one of the computed eigenvalues is greater than 1. Any norm-wise backward stable method (e.g., QR, serial bisection, Jacobi) would compute the eigenvalues to a relative accuracy of approximately $\epsilon \cdot \kappa(T) \approx 2 \times 10^{-10}$, which is much better.

We also give a bound that is stronger than (4) and is applicable to possibly indefinite tridiagonal matrices. This bound is easily computable in the positive definite case, but unfortunately, is rather expensive to compute in the general indefinite case.

Another approach is to consider the backward error in the $\hat{d}_i$'s rather than the forward error. We show that the backward error in the $\hat{d}_i$'s is $\eta = \max\{\eta_i\}$ where

$$\eta_i = \frac{|\hat{d}_i - D_i|}{|a_i\hat{d}_{i-1}| + 2|b_{i-1}^2\hat{d}_{i-2}|}, \quad \text{and} \quad D_i = a_i\hat{d}_{i-1} - b_{i-1}^2\hat{d}_{i-2}. \tag{6}$$

Here $a_i, i = 1, \ldots, n$ are the main diagonal entries of T, $b_i, i = 1, \ldots, n-1$ are the off diagonal entries of T and $b_0 = 0$. Note that each η_i can be computed independently, and that we are nt assuming the T is positive definite. Comparing η with ϵ, the arithmetic precision, one can see how reliable the computed d_i's are. If $\eta_i, i = 1, 2, \ldots, k$ are small then it follows that the $\hat{d}_i, i = 1, 2, \ldots, k$ are reliable, regardless of what the $\hat{d}_i, i = k+1, \ldots, n$ are. So if we use parallel prefix to compute the $\hat{d}_i$ and then use the η_i to check their backward error, even if the whole sequence $\hat{d}_i, i = 1, \ldots, n$ is not reliable we need only recompute $\hat{d}_i, i = k+1, \ldots, n$. This can result in considerable computational savings. We also discuss how to correct large backward errors cheaply.

This technique also enables one to determine, in less time than computing the d_i's, the reliability of an estimate of an eigenvalue of a symmetric tridiagonal matrix. This partially answer to another question raised in [1].

3 Other Applications

The parallel prefix operation can be used to parallelize many other algorithms in numerical linear algebra. We use the techniques of Sections 1 and 2 to analyze the accuracy of parallel prefix when used to evaluate a linear fractional recurrence, to compute the Cholesky, LDL^T, LU, and QR factorizations of a tridiagonal matrix, and to implement the differential qd algorithms of [2].

*Institute for Mathematics and its Applications, 514 Vincent Hall, 206 Church St. SE, University of Minnesota, Minneapolis, MN 55455. On leave from Department of Mathematics, College of William & Mary, Williamsburg, VA 23187. e-mail: na.mathias@na-net.ornl.gov. Research supported in part by National Science Foundation grant DMS-9201586.Additional support was provided by from a contract between the Army Research Office and the University of Minnesota for the Army High Performance Computing Research Center.

References

[1] J. Demmel. Open problems in numerical linear algebra. Technical Report 961, Institute for Mathematics and its Applications, University of Minnesota, April 1992.

[2] K. V. Fernando and B. Parlett. Accurate singular values and differential qd algorithms. Technical Report PAM-544, Center for Pure and Applied Mathematics, University of Calfornia, Berkeley, 1992.

ABSTRACT

AN ERROR ANALYSIS OF THE AGGREGATION/DISAGGREGATION PROCESS

Carl D. Meyer
North Carolina State University

Aggregation/Disaggregation is a numerical procedure designed to provide approximations to the solution of linear systems or to approximate eigenvectors of large-scale problems which are often intractable by other means. A/D schemes have been proposed as accelerators to standard iterative methods, and there are iterative A/D algorithms based on successive applications of A/D procedures.

The most successful application of A/D techniques has been to problems which have a nearly uncoupled structure—i.e., problems for which the states or variables can be grouped into clusters in which there exist relatively strong interactions within each cluster, but the clusters themselves are only weakly connected to each other. In particular, aggregation/disaggregation has become a popular method by which to solve problems from applications which can be placed in the context of large-scale nearly uncoupled Markov chains. Problems ranging from economic modeling to the analysis of large-scale queueing networks are known to fall into this category. In the Markov chain setting, A/D begins with an initial approximation $\mathbf{s}^T$ to the normalized left-hand eigenvector $\boldsymbol{\pi}^T$ associated with the unit eigenvalue of an irreducible stochastic matrix $\mathbf{P}$. If $\mathbf{P}_{n\times n}$ and $\mathbf{s}^T$ are partitioned as

$$\mathbf{P} = \begin{pmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} & \cdots & \mathbf{P}_{1k} \\ \mathbf{P}_{21} & \mathbf{P}_{22} & \cdots & \mathbf{P}_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{k1} & \mathbf{P}_{k2} & \cdots & \mathbf{P}_{kk} \end{pmatrix} \quad \text{and} \quad \mathbf{s}^T = (\mathbf{s}_1^T, \mathbf{s}_2^T, \cdots, \mathbf{s}_k^T),$$

then A/D returns a new approximation to $\boldsymbol{\pi}^T$ in the form

$$\tilde{\mathbf{s}}^T = (\xi_1\mathbf{s}_1^T, \xi_2\mathbf{s}_2^T, \cdots, \xi_k\mathbf{s}_k^T)$$

in which the ξ_i's are components of the dominant left-hand eigenvector of a smaller matrix $\mathbf{C}_{k\times k} = \mathbf{L}_{k\times n}\mathbf{P}\mathbf{R}_{n\times k}$ where $\mathbf{L}\mathbf{R} = \mathbf{I}_k$. Different choices of $\mathbf{L}$ and $\mathbf{R}$ lead to different algorithms—the most standard choice is to take

$$\mathbf{L} = \begin{pmatrix} \mathbf{s}_1^T/\|\mathbf{s}_1\|_1 & 0 & \cdots & 0 \\ 0 & \mathbf{s}_2^T/\|\mathbf{s}_2\|_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{s}_k^T/\|\mathbf{s}_k\|_1 \end{pmatrix} \quad \text{and} \quad \mathbf{R} = \begin{pmatrix} \mathbf{e}^{(1)} & 0 & \cdots & 0 \\ 0 & \mathbf{e}^{(2)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{e}^{(k)} \end{pmatrix}$$

where $\mathbf{e}^{(j)}$ is an appropriate sized column of 1's.

Several variations on this basic scheme have been proposed to create iterative A/D algorithms, and error analyses for some of these variations have been given by P.J. Courtois, G.W. Stewart, F. Chatlin and W.L. Miranker, and M. Haviv. Notwithstanding the past attention, there are a variety of detailed issues involved, and bringing all facets together to produce a clear picture seems to be less than straightforward. The purpose of this work is to take another look at A/D error from a different point of view.

Past researchers have relied more on spectral perturbation theory or else standard perturbation techniques from the theory of linear systems in order to produce norm-based error estimates. Furthermore, past work does not completely capitalize on the special properties enjoyed by stochastic matrices. Our approach differs from that of the past in the sense that we bring to bear tools which are specific to the analysis of Markov chain problems. In particular, we show how to use results concerning standard coefficients of ergodicity in conjunction with perturbation results specific to Markov chains to move away from traditional norm-based inequalities in order to produce relatively simple error estimates for the standard A/D procedure.

Matrix Algebra and Mappings of Signal Processing Algorithms on Array Processors

George J. Miel
Department of Mathematical Sciences
University of Nevada
Las Vegas, NV 89154

Summary

Array processors, made possible by very large scale integration (VLSI) and wafer scale integration (WSI) technology, provide effective parallel architectures for computationally intensive applications such as signal and image processing [3]. In this type of parallel computers, the cells of the array operate in Single Instruction Multiple Data (SIMD) mode and algorithms are executed in systolic or cellular fashion. Many linear algebraic procedures have properties of locality, recursiveness, and regularity that match well the local connectivity and fine-grain parallelism of array processors. It is well-known, for example, that many real-time signal processing tasks (such as adaptive filtering, beamforming, cross-ambiguity calculations, data compression, etc.) map well onto arrays because they may be reduced to a common set of linear algebraic operations [9]. There is considerable ongoing research aimed at further applications of linear algebraic techniques in array processing, see, e.g., Miel [4].

Our purpose here is to explore linear algebraic structures, not of the objects of algorithms, but in the actual process of mapping certain algorithms onto array processors. Two case studies are investigated. First, it is shown that a factorization of the perfect shuffle permutation specifies a mapping of such permutations onto rectangular array processors. Secondly, factorizations of unitary matrices are derived, based on Kronecker products and direct sums, that result in sparse factors with useful patterns and redundancies. Such factorizations effectively define the mapping of certain fast transforms by specifying both the concurrent execution of specialized matrix-vector products as well as the data movement among the processing cells of the array.

Our results can be summarized as the following theorems, in which S_n denotes the perfect shuffle permutation of order n and R_P denotes a general type of P-point unitary transform. The family of transforms for which our analysis is applicable is engendered by a single recursion formula, formulated in terms of a generalized Kronecker product of matrices, presented by Regalia and Mitra [8].

Theorem 1. *Let P=2MNs where M,N,s are powers of 2 with nonnegative integer exponents. Then*

$$S_P = C_1 C_2 C_0 \tag{1}$$

where

$$C_0 = \bigoplus_{j=0}^{Ms-1} S_{2N}, \quad C_1 = C_0 \overline{C}_1 C_0^t, \quad C_2 = C_0 \overline{C}_2 C_0^t,$$

$$\overline{C}_1 = \bigoplus_{m=0}^{M-1} (I_N \otimes S_{2s}), \quad \overline{C}_2 = I_{Ns} \otimes S_{2M}.$$

Theorem 2. *Let $P = 2^\rho$, where ρ is a positive integer. Then*

$$R_P = \overline{A}^{(\rho-1)} C^{(\rho-1)} \dots \overline{A}^{(1)} C^{(1)} \overline{A}^{(0)} C^{(0)}, \tag{2}$$

where

$$\overline{A}^{(k)} = \bigoplus_{i=0}^{2^k-1} \bigoplus_{\mu=0}^{2^{\rho-k-1}-1} B_i^{(k)},$$

$$C^{(0)} = S_{2^\rho}, \quad C^{(k)} = \left(\bigoplus_{i=0}^{2^k-1} S_{2^{\rho-k}} \right) \left(\bigoplus_{i=0}^{2^{k-1}-1} S_{2^{\rho-k+1}}^{-1} \right).$$

Theorem 3. *Let $P = 2^\rho$, where ρ is a positive integer. Then*

$$R_P = A^{(\rho-1)} C \cdots A^{(1)} C A^{(0)} C, \tag{3}$$

where

$$A^{(k)} = \bigoplus_{\mu=0}^{2^{\rho-k-1}-1} \bigoplus_{i=0}^{2^k-1} B_i^{(k)}, \quad C = S_P.$$

The matrix factorization (1) in Theorem 1 engenders a parallel SIMD algorithm for executing a perfect shuffle permutation on an $M \times N$ rectangular array processor. The resulting data movement is realized in parallel as relatively small perfect shuffles along each row and column and inside each local memory of the array processor, represented by the matrices C_0, C_2, C_1 respectively, without requiring that the complete array itself have the shuffle-exchange network. The algorithm engendered by matrix factorization (1) is the corner stone to a parallel algorithm recently presented by Miel [6] for executing a P-point constant geometry Fast Fourier Transform (FFT) on a rectangular $M \times N$ array processor. Theorem 3 above can be used to extend this result to the class of unitary transforms of type R_P.

The shuffle-exchange network has wide applicability in parallel processing, including bitonic sorting, polynomial evaluation, matrix transposition, and linear transformations [10]. Hence, implementation of the perfect shuffle represented by matrix factorizations (1) can be put to good use for such applications on rectangular array processors. Moreover, assuming bidirectional links in the shuffle connections, we also have access to the inverse perfect shuffle for executing on such arrays algorithms based on recursive doubling [2].

Matrix factorizations (2) and (3) in Theorems 2 and 3 represent parallel algorithms for the class of discrete unitary transforms engendered by the Regalia-Mitra recursion formula. Matrices of this type possess properties that match requirements for a wide range of applications in signal processing. Examples include the FFT used in filtering and frequency domain analysis, the discrete Cosine transform in data compression, the Haar and Hadamard transforms in dyadic-invariant signal processing, the Slant transform in image coding, and these and others such as the BIFORE transform in generalized spectral analysis, see [8].

In matrix factorization (2), each $C^{(k)}$ is a $P \times P$ permutation matrix and each $A^{(k)}$ is a block-diagonal matrix

$$A^{(k)} = \bigoplus_{i=0}^{P/2-1} B_i^{(k)}, \qquad (4)$$

in which each of the $P/2$ blocks $B_i^{(k)}$ is a 2×2 complex matrix. Each permutation matrix $C^{(k)}$ causes a rearrangement of the data and each 2×2 matrix $B_i^{(k)}$ denotes a butterfly operation. From the point of view of mapping the transform onto an array processor, the matrices $C^{(k)}$ represent communication operations that move data among the cells of the array, while the matrices $A^{(k)}$ represent arithmetic operations for SIMD execution of butterfly operations. Thus an algorithm for computing in accordance to matrix factorization (2) the transform $R_P z$ of an input P-vector z proceeds as follows:

```
for  $k = 0, 1, \cdots, \rho - 1$ do:
     $z := A^{(k)} C^{(k)} z$
     end for
$z := C^{(\rho)} z$
```

The symbol ":=" denotes a vector overwrite operation. Analogously to a radix 2 FFT each sweep of the for-loop represents one of $\log_2 P$ stages in the computation, with each stage consisting of a rearrangement of the data followed by $P/2$ butterfly operations.

The effectiveness of a mapping of a matrix factorization (2) onto an array processor depends primarily on two items. The first item concerns the data movement represented by the permutation matrices $C^{(k)}$, specifically, the efficiency with which the interconnection network realizes the data transfers required by the permutations. The second item involves a

divide-and-conquer strategy for the SIMD evaluation of specialized matrix-vector products. Suppose that a product Az, where $A = A^{(k)}$ is a block-diagonal matrix of type (4) and z is a P-vector, is to be computed on an array processor with $P/2$ cells. The vector is first divided into $P/2$ ordered pairs

$$z = (\bar{z}_0, \bar{z}_1, \cdots \bar{z}_{P/2-1})^t, \quad \bar{z}_i = (z_{2i}, z_{2i+1}),$$

each cell computes in parallel a butterfly operation $B_i \bar{z}_i$, and the subvectors are then concatenated to get the result. Whereas the first item deals with the communication complexity of the mapping, the second item pertains to its parallel arithmetic complexity.

Matrix factorization (3) in Theorem 3 deserves special consideration. It yields a so-called constant geometry algorithm because its communication pattern is kept the same from stage to stage. This means that the addressing of operands for the butterfly operations is identical for every stage. The general factorization contains as a special case the factorization obtained by Pease [7], in his modification of the Cooley-Tukey procedure, for parallelizing the FFT on linear arrays. As for the particular case, the most natural mapping of the general P-points factorization is onto a linear array architecture with $P/2$ cells and a shuffle-exchange interconnection network [10]. However, matrix factorizations (1) and (3) in Theorems 1 and 3 can be combined to yield parallel algorithms on a rectangular $M \times N$ array processor, for Pease's constant geometry FFT in specific [6], and in general for unitary transforms R_P that satisfy the Regalia-Mitra recursion. In Theorem 3, we in effect used the matrix approach pioneered by Pease in his parallelization of the Cooley-Tukey procedure as a means to characterize a wide class of constant geometry algorithms.

Miel and Yfantis [5] used matrix factorizations (2) and (3) in Theorems 1 and 2, combined with abstract constructs that link linear algebraic concepts to a high-level model of a given array processor, to obtain the mapping formalism of an interactive software tool that helps the user map discrete unitary transforms onto the array.

References

[1] P. Kogge and H. Stone, A parallel algorithm for the efficient solution of a general class of recurrence equations, *IEEE Trans. Comput.*, vol. 22, 1973, pp. 786-793.

[2] S.Y. Kung, *VLSI Array Processors*, Prentice Hall, Englewood Cliffs, NJ, 1988.

[3] G. Miel, Trends in systolic and cellular computation, *J. Computational and Applied Mathematics*, vol. 38, 1991, pp. 1-25.

[4] G. Miel and E. Yfantis, A software tool for cellular mapping of discrete unitary transforms, in *Proc. 6th International Parallel Processing Symposium, 23-26 March 1992, Beverly Hills, CA*, IEEE Computer Society Press, Los Alamitos, CA, 1992, pp. 298-304.

[5] G. Miel, Constant geometry Fast Fourier Transforms on array processors,*IEEE Trans. Comput.*, vol. 42, no. 2, February 1993, pp. 1-5.

[6] M.C. Pease, An adaptation of the Fast Fourier Transform for parallel processing, *J. Assoc. Comput. Mach.*, vol. 15, April 1968, pp. 252-264.

[7] P.A. Regalia and S.K. Mitra, Kronecker products, unitary matrices and signal processing applications, *SIAM Review*, vol. 31, Dec. 1989, pp. 586-613.

[8] J.M. Speiser and H. Whitehouse, A review of signal processing with systolic arrays, *SPIE Real-time Signal Processing VI*, August 1983, pp. 2-6.

[9] H.S. Stone, *High-Performance Computer Architecture*, Addition-Wesley Publishing, Reading, MA, 1987.

Abstract for Householder 93

A QR-like Approach in Eigenvalue Assignment Problems

by

George Miminis
Department of Computer Science,
Memorial University of Newfoundland, St. John's,
NFLD, CANADA, A1C 5S7,
e-mail:george@cs.mun.ca

A problem of vital importance in Control is the stabilization of **Dynamic Systems** (mathematical models that describe time changing physical phenomena). A popular method that stabilizes **Linear** dynamic systems is **Eigenvalue Assignment** (**Pole Placement** is a better known terminology amongst control engineers). The problem of stabilizing linear dynamic systems may be briefly described as follows. Consider the **continuous time system**

$$E\dot{x}(t) = Ax(t) + Bu(t) \tag{1}$$

$$y(t) = Cx(t) \tag{2}$$

where $E \in I\!R^{n\times n}, A \in I\!R^{n\times n}, B \in I\!R^{n\times m}$ and $C \in I\!R^{p\times n}$. Also $x(t) \in I\!R^{n}, y(t) \in I\!R^{p}$ are the **state** and the **output** of the system respectively, and $u(t) \in I\!R^{m}$ is the **input** or **control** to the system at time t. If we choose the input as $u(t) = -Fy(t)$, with $F \in I\!R^{m\times p}$ we have the so called **feedback** approach, for determining the input to the system. In this case (1) because of (2) gives the following equation

$$E\dot{x}(t) = (A - BFC)x(t) \tag{3}$$

It may be shown that, if E is nonsingular (3) is stable when the eigenvalues of the pencil $[A-BFC, E]$ have negative real parts. The **Eigenvalue Assignment Problem** may then be described as follows.

Problem: *Given matrices E, A, B and C and a self conjugate set of numbers Λ, compute matrix F such that the set of the eigenvalues of $[A - BFC, E]$ is equal to Λ.*

The above problem has a number of versions according to the following:

$$E\left\{\begin{array}{l}\neq I, \text{Descriptor systems}\\ = I, \text{Nondescriptor systems}\end{array}\right., C\left\{\begin{array}{l}\neq I, \text{Output feedback}\\ = I, \text{State feedback}\end{array}\right.$$

Although each of the above versions has its own peculiarities which may justly classify each version as a research problem of its own; there is definitely some **common ground** among them. For a number of years control engineers have been producing numerically unstable algorithms for the above problems. These algorithms were also totally unrelated of one another. The approach we have employed has been, to apply techniques from the QR or the QZ algorithms on the above Eigenvalue Assignment problems. We call this a QR-like approach. As a result we have managed to apply numerically powerful techniques like **Deflation**, so that, as the solution progresses we work with smaller problems, **Double Steps** so that only real arithmetic is used, and **Orthogonal Transformations** to facilitate **Numerical Stability**. We have therefore produced **Numerically Efficient** algorithms for each of the above Eigenvalue Assignment Problems. The **Numerical Stability** of most of these algorithms has been shown. We have also shown that each of these algorithms may been derived by appropriately "fine tuning" a **Generic algorithm**. The existence of such a generic algorithm proves our "common ground" claim among the different versions of Eigenvalue Assignment Problems.

To give some insight to the above approach we present an application of the **Generic** algorithm on the $[A - BF, E]$ $(C = I)$ **Eigenvalue Assignment Problem** when E is not singular.

1. Compute an eigenvector x_1 of $[(A - BF), E]$ corresponding to, say $\lambda_1 \in \Lambda$, with $\|x_1\|_2 = 1$.
2. Compute a unitary matrix $Q = (x_1, \tilde{Q})$.
3. Fx_1 is computed so that the eigenpair (λ_1, x_1) is assigned.
4. A unitary matrix $U = (y_1, \tilde{U})$ with $y_1 = \frac{Ex_1}{\|Ex_1\|_2}$ is computed.
5. Perform the transformation

$$U^H[(A - BF), E]\left(\begin{array}{c|c} Q & \\ \hline & Q\end{array}\right) = \left[\left(\begin{array}{c|c} \lambda_1\alpha & y_1^H(A - BF)\tilde{Q} \\ \hline & \tilde{U}^H(A - BF)\tilde{Q}\end{array}\right), \left(\begin{array}{c|c} \alpha & y_1^H E\tilde{Q} \\ \hline & \tilde{U}^H E\tilde{Q}\end{array}\right)\right]$$

   where $\alpha = y_1^H Ex_1 \neq 0$.

6. The assignment continues with the pencil $[\tilde{U}^H(A - BF)\tilde{Q}, \tilde{U}^H E\tilde{Q}]$.

In this paper we will present the double step version of the **Generic** algorithm on a specific **Eigenvalue Assignment Problem**. We will also show how we may "fine tune" it to produce a **Numerically Efficient** algorithm. From this, it will become apparent how to use the generic algorithm on the other eigenvalue assignment problems.

A package of MATLAB programs, named **PolePack** has been produced, with the implementations of our QR-like algorithms on all the above Eigenvalue Assignment Problems. The package will be briefly presented and it will become available to all those who may be interested.

What is a good Ω?

Jan Modersitzki
University of Hamburg
Bundesstraße 55
2000 Hamburg 13, FRG

Abstract. Consider the linear system of equations $Ax = b$, where A is a regular and normal matrix. We are concerned with polynomial iteration methods. These are methods where the iterates x_n are implicitly definded by

$$r_n = b - Ax_n = p_n(A) \cdot r_0, \quad p_n \in \Pi_n,\ p_n(0) = 1.$$

To estimate the convergence rate we introduce the following constraint approximation problem

$$\|r_n\|_2 = \min_{\substack{p \in \Pi_n \\ p(0)=1}} \|p(A)r_0\|_2 \leq \min_{\substack{p \in \Pi_n \\ p(0)=1}} \max_{\lambda \in \Omega} |p(\lambda)| \cdot \|r_0\|_2.$$

Here, Ω is a set containing all eigenvalues, but not the origin. It is a realistic bound, if the eigenvalues are fairly uniformly distributed in Ω.

Now, one may ask, what sets Ω lead to small convergence bounds? In the talk we will discuss a variety of possible sets. For instance, we will demonstrate that it is not advisable, despite popular believe, to precondition an indefinite system such that the resulting system has a symmetric spectrum with respect to the origin. The most delicate situation for nonsymmetric systems occurs, if the origin is contained in the convex hull of the eigenvalues. Here, some sets Ω produce a reasonable convergence rate, while others won't. We will present examples for both situations.

To solve the resulting approximation problems we use COCA, a MATLAB program package for solving COmplex Chebyshev Approximation problems, recently developed by Fischer and Modersitzki. This COCA package is publically available via the netlib facility. The program is very easy to handle and the interested audience may try their favourate Ω after the talk.

The COCA program package is also capable of handling weight functions. So, if time permitting, we will also discuss weigthed approximation problems,

$$\min_{\substack{p \in \Pi_n \\ p(0)=1}} \max_{\lambda \in \Omega} |\ w(\lambda) \cdot p(\lambda)\ |,$$

where the weight function w is designed to reflect the eigenvalue distribution.

On the group theoretic properties of Fast Poisson Solvers

Hans Munthe-Kaas
Dept. of Informatics, University of Bergen, N-5020 Bergen Norway
Email: hans@ii.uib.no, Tel.: +47-5-544179, Fax : +47-5-544199

February 1, 1993

1 Abstract for the Householder meeting 1993

Fast Poisson solvers is a family of direct algorithms for solving certain classes of constant coefficient elliptic PDE's. These algorithms were originally introduced by G. Golub and R. Hockney in 1965 and further developed by several authors, among them O. Buneman (1969), P.N. Swarztrauber (1974-1977) and U. Trottenberg/ J. Schröder (1973-1976). The main techniques being used in Fast Poisson Solvers are *cyclic reductions*, *total reductions* and *fast Fourier transforms (FFTs)*.

In this talk we will show that the classical theory of fast Poisson solvers may be formulated within the framework of group theory, in particular *finitely generated Abel groups* and *crystallographic symmetry groups*. The Abel groups arise because of local commutativity of the differential operators involved (due to constant coefficients) and the crystallographic groups enter the theory because of the boundary conditions. The theory permits a systematic study of different reduction schemes, where cyclic reductions and total reductions are special cases of more general reductions. The FFTs appear naturally as a dual approach to cyclic/total reductions.

Classical fast Poisson solvers are restricted to very simple geometries and boundary conditions, such as circles and rectangles. By investigating the crystallographic symmetry groups, we show that the techniques may be extended to other geometries, such as e.g. isosceles and equilateral triangles in 2-D and tetrahedrons in 3-D.

Finally, we will discuss the relevance of this theory in the design of iterative preconditioners for more general elliptic problems. Our theory provides us with exact expressions for the spectra of the reduced operators, extending well known results for the (1-way) classical cyclic reduction to more general (2-way) total reduction schemes. Whereas the condition number of the cyclically reduced operator is known to be increasing exponentially (in the explicit version), the total reduction schemes are known to be stable. A better understanding of these phenomena is of importance in the design of iterative preconditioners.

An Implementation of the QMR Method Based on Coupled Two-Term Recurrences

Noël M. Nachtigal

Recently, we proposed a new Krylov subspace iteration, the quasi-minimal residual (QMR) algorithm, for solving general nonsingular non-Hermitian linear systems. The original QMR algorithm relies on the three-term look-ahead Lanczos process to generate the basis vectors for the underlying Krylov subspace. It then constructs iterates defined by a quasi-minimization property, which leads to a smooth and nearly monotone convergence behavior. The quasi-minimization property is also strong enough to enable one to obtain theoretical results describing the convergence of the method.

However, empirical observations indicate that, in finite precision arithmetic, three-term vector recurrences are less robust than the mathematically equivalent coupled two-term recurrences. We therefore propose an implementation of the look-ahead Lanczos algorithm using coupled two-term recurrences. We then derive a new implementation of the QMR algorithm, and present some of its properties, as well as numerical examples.

The main idea behind the coupled two-term variant of the look-ahead Lanczos process is to construct a factorization of the Lanczos block tridiagonal matrix and to use the factors to define two auxiliary sequences of vectors. The basic recurrences constructed by the Lanczos algorithm can be written as

$$\begin{aligned} AV_n &= V_{n+1}H_n, \\ A^TW_n &= W_{n+1}H_n, \\ W_n^TV_n &= D_n, \end{aligned}$$

where V_n and W_n are the matrices containing the left and right Lanczos vectors after n steps, respectively, H_n is the $(n+1) \times n$ block tridiagonal

How dense is sparse orthogonal factorization?[1]

Esmond Ng
Mathematical Sciences Section
Oak Ridge National Laboratory

Let A be an $m \times n$ matrix with $m \geq n$, and assume that A has full column rank. We consider the orthogonal factorization of A when it is large and sparse.

Denote the orthogonal factorization of A by

$$A = QR,$$

where Q is $m \times n$ with orthonormal columns and R is $n \times n$ upper triangular. If the computation is organized so that the diagonal elements of R are positive, then the factorization is unique. The decomposition can be obtained by applying either Householder transformations or Givens rotations to annihilate the nonzeros in the strictly lower trapezoidal part of A.

There are two ways to represent the orthogonal factor Q. It can be represented *implicitly* as a product of Householder transformations or Givens rotations. For definiteness, assume that Householder transformations are used. Then all we need to store are basically the nonzeros in the vectors that are used to form the Householder transformations. These vectors can be conveniently represented by an $m \times n$ lower trapezoidal matrix H, which will be referred to as the Householder matrix. Alternatively, the orthogonal factor can be formed and represented *explicitly* as an $m \times n$ matrix.

When A is sparse, fill occurs in the QR factorization; that is, some of the elements that are zero in A becomes nonzero during the computation. George and Ng (1987) have provided a way to *bound* the sparsity structure of the Householder matrix H, which relies on the so-called *elimination tree* associated with the upper triangular factor R. In this talk, we extend the results due to George and Ng. By examining the sparsity structures of R and H, we construct a *generalized elimination tree*. Using the generalized elimination tree, we are able to *bound* the sparsity structure of the explicit orthogonal factor Q. By making use of a recent result which is due to Hare, Johnson, Olesky, and Driessche, we prove that the bounds on the structures of H and Q are tight when A is a *strong Hall matrix*. Moreover, we show that the *lower trapezoidal* parts of Q and H have *identical* sparsity structures. When A is a *Hall matrix*, similar results can be obtained if A has been permuted into block upper triangular form, where the diagonal blocks are strong Hall.

[1] Joint work with Barry Peyton.

Variable Block CG Algorithms for Solving Large Sparse Symmetric Positive Definite Linear Systems on Parallel Computers

A.A. Nikishin
Russian Academy of Sciences
and
Elegant Mathematics, Inc. (USA)
E-mail: nikishin@sms.ccas.msk.su

The talk describes a new approach to construction of efficient parallel solution methods of large sparse SPD linear systems. The source of parallelism is mostly related to the basic iterative scheme and not to the preconditioning strategy. This approach is based on the so called Variable Block CG methods, a generalization of the standard Block CG method [1], where it is possible to reduce the iteration block size adaptively (at any iteration) by construction of an A-orthogonal projector without restarts and without algebraic convergence of residual vectors. The general iterative scheme can be written as follows:

Given an initial guess x^0 and an initial block size $s(0)$. Construct a right hand side matrix $B \in \mathcal{R}^{n\times s(0)}$ and an initial guess matrix $X^0 \in \mathcal{R}^{n\times s(0)}$ whose first columns coincide with b and x^0, respectively, while other columns are chosen arbitrarily to produce a full-rank matrix $R^0 = B - AX^0$.

Initial stage : Set $R^0 = B - AX^0$ and $P^0 = MR^0$, where M is a SPD preconditioner.

For $k = 0, 1, \ldots$ *iterate* :

1. $\tilde{R}^{k+1} = R^k - AP^k\alpha_k$ and $\tilde{X}^{k+1} = X^k + P^k\alpha_k$, where $\alpha_k \in \mathcal{R}^{s(k)\times s(k)}$ are determined so that $\tilde{R}^{k+1T}P^k = 0$.

2. Choose with respect to some criterion a positive integer $s(k+1) \le s(k)$ and a matrix $\epsilon_k \in \mathcal{R}^{s(k)\times s(k+1)}$ so that $rank(\tilde{R}^{k+1}\epsilon_k) = s(k+1$..

3. Set $R^{k+1} = \tilde{R}^{k+1}\epsilon_k$, $X^{k+1} = \tilde{X}^{k+1}\epsilon_k$, and $\tilde{P}^k = P^k\alpha_k\epsilon_k$.

4. If $d(k) = s(k) - s(k+1) > 0$ then choose $H^k \in \mathcal{R}^{n\times d(k)}$ so that

$$\begin{aligned} &(i) \quad span(P^k) = span(H^k) \oplus span(\tilde{P}^k), \\ &(ii) \quad \tilde{P}^{kT}AH^k = 0 \quad \text{and} \quad H^{kT}AH^k = I. \end{aligned}$$

5. Update the block direction P^{k+1}

$$P^{k+1} = MR^{k+1} + \tilde{P}^k\beta_k + \sum_{i=0}^{k} H^i\gamma_k^i,$$

where the coefficients β_k and γ_k^i are computed so that $P^{k+1T}A\tilde{P}^k = 0$ and $P^{k+1T}AH^i = 0$ for $i \le k$.

The following theorem establishes VBPCG properties similar to the orthogonal and conjugate properties of the BPCG iterates.

Theorem 1 *For $j < k$ the VBPCG method iterates satisfy the following conditions*

$$\begin{aligned} R^{kT}MR^j &= 0, \\ P^{kT}AP^j &= 0, \\ R^{kT}P^j &= 0. \end{aligned}$$

The minimization properties of the VBPCG algorithm are stated in the following theorem.

Theorem 2 *Let $\bar{Q}_k$ denote the conjugate projector with respect to A on $\mathcal{H}_k = span\{H^0, H^1, \ldots, H^k\}$ and $Q_k = I - \bar{Q}_k$.*

Then X^{k+1} from the VBPCG method minimizes $tr[(X - X^*)^TA(X - X^*)]$ over all $X \in \mathcal{R}^{n\times s(k+1)}$ such that

$$X - X^* \in span\{Q_kMR^0, Q_kMAQ_kMR^0, \ldots, (Q_kMA)^kQ_kMR^0\} \oplus \mathcal{H}_k,$$

where $X^* = A^{-1}B\epsilon_0\epsilon_1\ldots\epsilon_k$.

The convergence analysis enables one to find the constructive compromise between the required resource of parallelism, the resulting convergence rate, and the serial arithmetic costs of one block iteration to minimize the total parallel solution time. The results of numerical experiments with large FE systems coming from h- and p-approximations of 3D equilibrium equations for linear elastic orthotropic materials show that the convergence rate of the Variable Block PCG method is comparable with that of the standard Block PCG method even when utilizing a large block size, while the total serial arithmetic costs of the Variable Block PCG method are comparable or even smaller than those of the corresponding point PCG method. Moreover, the Variable Block PCG method has been proven to more reliable than the corresponding point PCG method.

References

[1] D.P. O'Leary. The block conjugate gradient algorithm and related methods. *Linear Algebra Appl.*, 29:293–322, 1980.

Recent results on MIC strategies [1]

Yvan Notay [2]
Service de Métrologie Nucléaire
Université Libre de Bruxelles (C.P. 165)
50, Av. F.D. Roosevelt, B-1050 Brussels, Belgium.
email : ynotay@ulb.ac.be

The purpose of our talk is to present recent developments of strategies to be used in the implementation of MIC preconditiongs.

MIC preconditioning is often presented, discussed or even criticized in the literature on the basis of an erroneous presentation of the philosophy behind (the successful use of the) method, such as : using extensively the unperturbed method even in cases where it does not lead to a positive definite preconditioner, or using uniform perturbations where they should be modulated, considering arbitrarily as negligible the influence of strong local perturbations.

We feel therefore necessary to first summarize the basic principles of the analysis of MIC methods, stressing their potential sensivity to various features such as ordering and (modulated) perturbations.

We next present recent results obtained in this field.

First, we discuss anisotropic problems. The original conditioning is then potentially so bad that one may obtain disastrous results when applying straightforwardly some perturbation strategies proved robust for isotropic problems. We suggest the use of a recently developed technique, which reproduces the behaviour of the former methods in isotropic cases, but which compensates their weakness in the anisotropic cases by taking benefit of the small size of the fill entries.

Besides, we analyse how these neglected entries may be further reduced by allowing more fill-in. This depends heavily on the ordering and we prove that reverse Cuthill-Mc Kee orderings are efficient in this respect; for instance, considering the 5 point finite difference approximation of the constant coefficient PDE

$$-K_x\partial^2_{xx}u \,-K_y\partial^2_{yy}u \;=\; f \;,$$

we show that all discarded fills are reduced to $\mathcal{O}(\varepsilon^2)$, where $\varepsilon = \min(K_x/K_y, K_y/K_x)$, while the total number of nonzero offdiagonal entries in the triangular factors is kept less than $4n$, where n is the number of unknowns. Using the above mentioned method, the approximate factorization turns then out to be nearly exact in strongly anisotropic regions, and, as these strategy choices are also worthwhile in isotropic regions, we have there a robust method which may be proposed as reference for MIC preconditioning.

The next point under consideration is the application of incomplete factorization preconditioning to positive definite matrices with offdiagonal entries of arbitrary sign. Indeed, the main limitation of the MIC theory is that it requires the input matrix to have nonpositive offdiagonal entries (i.e. to be a Stieltjes matrix). We discuss the reduction techniques that allow to deduce a Stieltjes approximation of the system matrix, and especially the "diagonal compensation method", according to which the positive offdiagonal entries are discarded and added to the diagonal so as to preserve the row sum. A combined use with a more direct approach is also considered, and a connection made with the methods based on hierarchical bases.

Considering the finite element context, we prove that these techniques are reliable if and only if the elementary submatrices satisfy some given spectral properties. This is mostly the case for second order elliptic problems, but not for e.g. coupled systems of second order PDEs such as those encountered in structural mechanic. In the latter case, it is better to first deduce a block diagonal approximation based on the decoupling of the different PDEs, and use the diagonal compensation method only as complementary technique. We present some application results obtained in this field for unstructured grid problems, in collaboration with P. St-GEORGES within the framework of the development of the code PCGELFIN.

Finally, we discuss the ordering problem for approximate factorization preconditioners in connection with their parallel implementation. As is well known, parallelism may be increased by reordering, but this may imply a deterioration of the convergence properties. Returning to the conditioning analysis, we examine how ordering requirements are expressed in the latter and to what extent they are compatible with parallel orderings. Further, we deduce the cheapest way to introduce more parallelism and estimate the corresponding increase of the number of iterations. This results in a family of orderings with increasing parallelism and decreasing convergence properties, with at one end standard natural orderings and at the opposite end the red/black ordering.

Acknowledgements. We thanks Prof. R. Beauwens for useful comments and P. Saint-Georges for its kind help in the preparation of some numerical tests.

[1] The present work was supported by the "Programme d'impulsion en Technologie de l'Information", financed by Belgian State, under contract No. IT/IF/14.

[2] Supported by the "Fonds National de la Recherche Scientifique", Chargé de recherches.

History and Generality of the CS Decomposition *

C. C. Paige†

March 3, 1993

Abstract

It is nearly a quarter of a century since Chandler Davis and William Kahan introduced the basics of what Stewart later developed into the CS decomposition of a partitioned unitary matrix. Since then many users of the CSD have recognized it as one of the major tools of matrix analysis. This talk outlines some germane points in the history of the CSD, pointing out the contributations of Davis and Kahan,and Stewart, and the relationship of the CSD to Davis' "direct rotation". The talk next suggests a motivation for the CSD which emphasizes how generally useful it is. It provides an easy to memorize constructive proof of the CSD and reviews some of its diverse uses. It then points out some useful but previously unnoticed nullity properties one form of the CSD trivially reveals, and so extends its area of application. Finally it shows how via the QR factorization, the CSD can be used to obtain interesting and nontrivial results for partitioned nonsingular matrices, thus emphasizing yet again the power of this decomposition.

Key words: CS decomposition, unitary matrices, direct rotation, angles between subspaces, nonsingular matrices.

AMS Subject Classifications: 65F25, 15A23, 15A57

[*] This research was partially supported by NSERC of Canada Grant No. A9236.

[†] Computer Science, McGill University, Montreal, Quebec, Canada, H3A 2A7, (chris@cs.mcgill.ca).

Recent Advances in Rank–Revealing QR Factorization

C.–T. Pan
Department of Mathematical Sciences
Northern Illinois University
DeKalb, IL 60115

January 16, 1993

Abstract

In many numerical linear algebra applications, estimating the gaps between two consecutive singular values of a given matrix without actually computing them is of essential interests. For example, this estimation can be used to replace the SVD in solving the rank determination problems, column selection problems, and matrix approximation problems.

A recent work by Hong and Pan[1] shows that the rank–revealing QR factorization, or pivoted QR factorization provides us with a tight estimation on the gap between any two a priori–chosen consecutive singular values. This is a first rigorous proof for the existence of the rank-revealing factorizations. However, since the proof utilizes a set of right singular vectors to decide the column permutation needed, it does not serve directly for the purpose of estimating the gap without knowing the SVD of the given matrix.

It turns out that such a pivoting strategy totally depending on the norms of the columns or partial columns of the given matrix, like the well-known column pivoting strategy proposed by Golub in 1965, does exist, and it identifies the column permutation needed for obtaining the bounds on the gaps of two consecutive singular values. Furthermore, the bounds so obtained is exactly the same bounds obtained earlier in our existence proof.

In more detail, a recent report of Pan and Tang[2] introduces a pair of dual concepts, **pivoted blocks** and **reverse pivoted blocks**. These blocks are the natural generalization of the Golub's column pivoted magnitudes and the Stewart's reverse column pivoted magnitudes respectively.

We define the **pivoted magnitude** of a given matrix A, $\eta(A)$, as the largest column norm among all the columns of A. We define the **reverse pivoted magnitude** of a given matrix A, $\tau(A)$, as follows:

$$\tau(A)\overset{\mathrm{def}}{=}\min\left\{|r_{nn}^{(l)}| : A\Pi_{l,n} = Q^{(l)}R^{(l)}, l = 1,2,\ldots,n\right\}$$

where $\Pi_{i,j}$ is the permutation such that $A\Pi_{i,j}$ interchanges columns i and j of A and $Q^{(l)}R^{(l)}$ is the QR factorization of $A\Pi_{l,n}$.

Now we define what we mean by a **pivoted block**.

Let $\Pi_{l,k}$, $l = 1,2,\ldots,k$ be the column permutation matrices just defined. Consider all the QR factorizations,

$$A\Pi_{l,k} = Q^{(l)}R^{(l)} = Q^{(l)}\begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \begin{pmatrix} R_{11}^{(l)} & R_{12}^{(l)} \\ 0 & R_{22}^{(l)} \end{pmatrix} \end{array}.$$

We call R_{11} a pivoted block of A, if

$$|r_{kk}^{(l)}| = \eta(\bar{R}_{22}^{(l)}) \qquad \text{for } l = 1,2,\ldots,k,$$

where $\bar{R}_{22}$ is the $(n-k+1)$-by-$(n-k+1)$ trailing principal submatrix of R.

It is proved in the report that if R_{11} is the pivoted block, then the same R_{11} and the corresponding R_{22} are the blocks which guarantee a RRQR factorization and satisfy the inequalities:

$$\|R_{22}\|_2 \le \sqrt{k(n-k)+\min(k,n-k)}\,\sigma_{k+1}(A),$$

and

$$\sigma_{\min}(R_{11}) \ge \frac{1}{\sqrt{k(n-k)+\min(k,n-k)}}\sigma_k(A).$$

For defining the reverse pivoted block, consider the column permutations $\Pi_{l,k+1}$, $l = k+1, k+2,\ldots,n$, and

$$A\Pi_{l,k+1} = Q^{(l)}R^{(l)} = Q^{(l)}\begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \begin{pmatrix} R_{11}^{(l)} & R_{12}^{(l)} \\ 0 & R_{22}^{(l)} \end{pmatrix} \end{array}.$$

We call R_{22} a reverse pivoted block, if

$$\tau(\bar{R}_{11}^{(l)}) = |r_{k+1,k+1}^{(l)}| \qquad \text{for } l = k+1, k+2,\ldots,n,$$

where $\bar{R}_{11}$ is the $(k+1)$-by-$(k+1)$ leading principal submatrix of R.

It is shown in the report that R_{11} is a pivoted block of R if and only if R_{22} is a reverse pivoted block of A.

The following algorithm does the column pivoting (we call it cyclic pivoting) and finds a pivoted block at the end. We use the notation $\mathcal{R}(M)$ to denote the R factor of a matrix M.

Algorithm. Given an integer k, $1 \le k < n$, this algorithm produces a column permutation Π such that $A\Pi = QR$ and R_{11} is a pivoted block.

```
Step 0. Initialization: $R := \mathcal{R}(A\Pi)$ with (Golub) column pivoting, where $\Pi$ is the
        column permutation. Set $i := k - 1$.

Step 1. Iteration; cyclic pivoting

    Step 1.1. If $i = 0$, exit algorithm.
    Step 1.2. Set $R := \mathcal{R}(R \cdot \Pi_{i,k})$; $\Pi := \Pi \cdot \Pi_{i,k}$.
    Step 1.3. If $|r_{kk}| = \eta(\bar{R}_{22})$, then set $i := i - 1$. Otherwise, perform exchange
    as follows: Find an $l$, $k+1 \le l \le n$, such that
        $\eta(\bar{R}_{22}) = \|[r_{k,l}, r_{k+1,l}, \ldots, r_{n,l}]^T\|_2$
    Set $R := \mathcal{R}(R\Pi_{k,l})$, $\Pi := \Pi \cdot \Pi_{k,l}$, and $i := k - 1$.
    Step 1.4. Go back to Step 1.1.
```

The iteration will terminate, because whenever an exchange takes place in Step 1.3, the value $|\det(R_{11})|$ strictly increases. Therefore, this exchange can happen only a finite number of times. Then, at most $k-1$ iterations can take place after the final exchange. Clearly, at termination, R_{11} is a pivoted block by our definition.

We implemented this algorithm in MATLAB 3.5i, on a SUN SPARC IPC at NIU. The numerical experiments confirm the tight estimates that our theory asserts.

[1] Y. Hong and C.–T. Pan, "Rank-revealing QR factorizations and the Singular Value Decomposition," *Math. Comp.* 58 (1992), 213–232.

[2] C.–T. Pan and P.T.T. Tang, "Bounds on singular values revealed by QR factorizations," *Tech. Report, MCS Argonne National Lab.*, MCS–P332–1092.

SUPEREFFECTIVE SLOW-DOWN OF PARALLEL COMPUTATIONS[*]

Victor Y. Pan

Department of Mathematics
Lehman College
CUNY, Bronx, NY 10468

Franco P. Preparata

Computer Science Department
Brown University
Providence, RI 02912

Abstract

Brent's scheduling principle provides a general simulation scheme when fewer processors are available than specified by the fastest parallel algorithm. Such a scheme preserves the actual number of *executed* operations, and when applicable, it provides a processor balancing technique that significantly reduces the *work*, expressed as the number of *executable* operators. In this paper we discuss a new technique, called ***supereffective slow-down***, that yields quite fast an algorithm with work significantly smaller than that of the fastest algorithm for the same problem. This technique can be viewed as a work-preserving acceleration of an existing recursive sequential algorithm for the considered problem. The presented examples include the computation of path algebras in graphs and digraphs and various computations in linear albegra. Some of the new algorithms may have practical value; for instance, we substantially improve the performance of the known parallel algorithms for triangular linear systems of equations.

Key words: Parallel algorithms, processor efficiency, linear system of equations, path algebras, paths in graphs.

1991 Math. Subject Classification: 68Q22, 68Q25, 68R10

1 Introduction

The primary objective of parallel computation, which more sharply contrasts it against sequential computation, has traditionally been the minimization of computation time t, i.e., of the number of parallel steps required to solve a given problem; a secondary, but still very important, criterion is the size of the equipment, expressed as the number of processors used in the computation. When t is the performance criterion, frequently the resulting algorithms involve very large numbers of processors (all assumed to be identical and capable of executing one arithmetic operation in unit time). It is reasonable to assume that very rarely will the number of usable processors match the requirements of the fastest algorithm for a given problem instance; typically, instead, there will be situations where the number of available processors is fixed, and its choice is dictated by economic as well as engineering reasons.

Thus the typical situation is one where we have for use far fewer processors than are necessary to achieve the minimum computation time; this situation is dealt with by means of the so-called *Brent's scheduling principle* [Br], [KR], which embodies a general simulation scheme of a system with sufficiently many (m) processors by one with a fixed number p such that

$$1 \leq p \leq m$$

of such processors. Specifically, if q is the total number of operations executed by the former system in time t, then the latter system can accomplish the same task in time slow-down, the computation is much faster than that of the best known sequential algorithm. This technique has been implicitly used in [BPa] for polynomial division and in [BPb] for computing modulo x^n the square root (and similarly the m-th root for any integer $m \geq 2$) of a polynomial $p(x)$.

A complementary viewpoint is to consider supereffective slow-down of parallel computation as an "effective" parallel acceleration of an efficient (not necessarily optimal) sequential algorithm for the given problem, i.e., a parallelization that preserves (or slightly increases) the total potential work w; typically, however, the acceleration will not achieve the best known parallel time. Problems that lend themselves to this treatment — as our examples suggest — are those solvable by a sequential recursive algorithm, whose task is to reduce an instance of size n to an instance of size $n-1$. If such an algorithm exists, then the trick is to replace the sequential nonrecursive portion of the algorithm acting on a subproblem of size 1, with a parallel subroutine acting on size $s \geq 1$, and to seek the largest s for which w is maintained constant.

Our objective is to extend this approach to a large class of computations, in particular, to some fundamental computations with matrices and graphs. Our study shows that the supereffective slow-down is possible for numerous parallel computations that may fast extend the solution of a problem of size s to one of size ks for any positive integers s and k.

We only demonstrate our techniques for few computational problems, in particular, for the inversion and quasi-inversion of matrices and for solving structured systems of linear equations, but these problems are fundamental and have numerous applications to linear algebra computations (matrix inversion), to path algebras in graphs and digraphs (quasi-inversion), and to various areas of symbolic computations (structured linear systems).

We believe that some of our algorithms have practical value. In particular, for computations in numerical linear algebra, such as solving triangular linear system of equations, these algorithms run faster than the known customary algorithms, even when the number of processors is reasonably bounded. Furthermore, our algorithms intensively use block matrix computations, which can be effectively implemented on loosely coupled multiprocessors. Finally, we refer the reader to [UY], [S] and [S,a] for some alternate nonalgebraic techniques on supereffective slow-down of graph algorithms.

We will organize our paper as follows: After some definitions and preliminaries in Sections 2 and 3, we show how to apply a supereffective slow-down to quasi-inversion of matrices over the semirings and to their inversion over the fields. In Section 5, we treat the inversion of triangular matrices and solving triangular linear systems of equations. In Section 6 we consider the same computations in the case of Toeplitz-like input matrices, having further extension to polynomial computations.

[*]This work was supported in part by NSF Grants CCR 90-20690, CCR-91-96152 and PSC CUNY Awards 661340 and 662478

Downdating the Rank-Revealing URV Decomposition

Haesun Park [1] and Lars Elden

Computer Science Department
University of Minnesota
Minneapolis, MN 55455, U.S.A.

We present an accurate algorithm for downdating a row in the rank-revealing URV decomposition, which was recently introduced by Stewart [5]. By downdating the full rank part and the noise part in two separate steps, the new algorithm can produce accurate results even for ill-conditioned problems. Other possible generalizations of existing QR decomposition downdating algorithms for the rank-revealing URV downdating are discussed.

The singular value decomposition (SVD) is of great theoretical and practical importance [3]. However, the SVD has the drawback that it is computationally expensive. Especially when the problem is of recursive nature, the SVD requires $\mathcal{O}(n^3)$ flops for a matrix of order n even for a simple update such as adding a new row. Thus, efficient algorithms that utilize the existing results for incorporating changes in data are desired. The two-sided orthogonal decompositions, called the rank-revealing URV and ULV decompositions (RR URVD and RR ULVD) due to Stewart [4,5], have been shown to effectively exhibit the rank and the basis for the null space, and can be updated in $\mathcal{O}(n^2)$ flops. They are compromises between the SVD and a QR decomposition with some of the virtues of both.

For recursive problems, there are two common ways for incorporating changes in data, which are the sliding rectangular window method and the exponential window method. For phasing out the old data, one or more rows are deleted explicitly in the sliding rectangular window method, and a forgetting factor is multiplied to existing rows to damp out the effect of the old information in the exponential window method. After an update in the exponential window method, the numerical rank can increase, decrease, or stay the same. The sliding window method can track the change in the information statistics more accurately than the forgetting factor method when there is an abrupt change in data such as when signals are turned on and off, or outliers are removed [1,6].

Our new algorithm for modifying the RR URVD uses the sliding window method with downdating instead of the exponential window method that uses a forgetting factor [5]. An advantage of the sliding window method is the a priori information on the rank after the modification: mathematically, after adding a row, the rank can only stay the same or increase by one, and, after deleting a row, the rank can only stay the same or decrease by one. Thus, the indefinite steps of deflation in using the forgetting factor that results from not having any similar a priori information on the rank of the modified matrix can be eliminated.

The new algorithms are partly based on the algorithms for downdating the QR decomposition described in [2], the LINPACK algorithm, the corrected semi-normal equation (CSNE) method, and a hybrid method between the LINPACK and the CSNE downdating algorithms. It is necessary to use accurate algorithms, since the downdates in rank deficient cases can be very ill-conditioned. A two-step procedure, where the downdating of the signal and noise parts are performed separately, enables us to obtain accurate results using the LINPACK/CSNE hybrid algorithm for the ill-conditioned downdates, which occur when the numerical rank is decreased. Also, we show how the noise part can be downdated in a robust way, even in the case when the noise part is numerically singular.

The numerical tests show that the two-step algorithm based on the hybrid algorithm for the signal part downdating combined with the new downdating algorithm for singular noise block produces particularly accurate results in computing the basis for the null space and the numerical rank even for ill-conditioned downdating.

1. M. G. Bellanger. The Family of Fast Least Squares Algorithms for Adaptive Filtering. In Mathematics in Signal Processing, ed. J. G. McWhirter, Clarendon Press. Oxford, 1990, 415–434.

2. Å. Björck, H. Park, and L. Eldén. Accurate Downdating of Least Squares Solutions. *SIAM J. Matrix Anal. Appl.*, to appear.

3. G. H. Golub and C. F. Van Loan. *Matrix Computations. 2nd ed.* Johns Hopkins Press, Baltimore, MD., 1989.

4. G.W. Stewart. Updating a rank-revealing ULV decomposition. Tech. report, Dept. of Computer Science, Univ. of Maryland, CS-TR-2627, 1991.

5. G.W. Stewart. An updating algorithm for subspace tracking. *IEEE Trans. Signal Proc.* 40(1992), 1535–1541.

6. G. Xu, H. Zha, G. Golub, and T. Kailath. Fast and Robust Algorithms for Updating Signal Subspaces. Submitted to *IEEE Trans. Circuits Systems.*

[1] This work was supported in part by the National Science Foundation grant CCR-9209726 and also by a contract DAAL02-89-C-0038 between the Army Research Office and the University of Minnesota for the Army High Performance Computing Research Center.
This is a joint work with Lars Eldén, Department of Mathematics, Linköping University, Linköping, Sweden.

PRECONDITIONERS FOR LEAST SQUARES ITERATIONS

Robert J. Plemmons

Least squares problems occur frequently in science and engineering. In many cases the systems are large and dense, in which case iterative methods of solution are of prime importance. Often the characteristics of the underlying physical problem will induce some structure on the least squares system. For example the generating function f for the matrix A of coefficients may be periodic in nature, as in image restoration problems. In other situations it is known that the entries of the matrix are generated by a "smooth" function f on a regular grid, as in problems in potential theory and in signal compression.

For such important situations our purpose is to describe the use of transform-based preconditioned conjugate gradient iterative methods of solution for these large dense least squares problems.

A variety of preconditioners are considered for the problems described above. The methods include the use of preconditioners based on the fast Fourier transform for problems with periodic-type generating functions, and the use of fast wavelet transforms for problems with generally smooth generating functions. These decompositions and subsequent iterations can often be done in O(M log N) or even O(M) operations for M-by-N least squares problems.

HIGHLY PARALLEL SPARSE TRIANGULAR SOLUTION [*]

ALEX POTHEN[†]

We consider some recent developments in the solution of sparse triangular linear systems of equations on highly parallel computers such as the Connection Machine CM-2. For concreteness, we consider a lower triangular system in the following discussion, but the results apply to upper triangular systems as well. On highly parallel machines it is advantageous to compute the solution to a lower triangular system $L\underline{x} = \underline{b}$ by matrix-vector multiplication $\underline{x} := L^{-1}\underline{b}$ when there are several systems (not all available at the same time) involving the matrix L to be solved. This is because there is much more parallelism to be exploited in the multiplication approach than in the conventional substitution algorithm. If we can find a factorization $L = \Pi_{i=1}^{t} P_i$, where each factor P_i has the property that P_i and P_i^{-1} have the same nonzero structure, then $L^{-1} = \Pi_{i=t}^{1} P_i^{-1}$ can be represented in a space-efficient manner, storing the t factors P_i^{-1} in the space required for L. Furthermore, the vector $\underline{x}$ can be computed by a sequence of t matrix-vector multiplication steps, exploiting parallelism fully within each step.

The number of factors t in the factorization of L is an important measure since it is proportional to the number of expensive router communication steps required by the parallel algorithm based on this approach; hence it is a good predictor of the running time of triangular solution on machines like the Connection Machine CM-2.

It has been recognized that the triangular matrix can be symmetrically permuted before factorization to reduce the number of factors t, and we consider the problem of minimizing this number over several appropriate classes of permutations. We have also considered the numerical stability of this approach. A survey of this approach, with computational results on the CM-2, is provided in [1].

Partitioning problems. A matrix X is *invertible in place* if and only if $(X^{-1})_{ij} \neq 0 \rightarrow x_{ij} \neq 0$. Since the elementary lower triangular matrices are invertible in place, there is always at least one partition of L with factors that invert in place. A partition in which the factors P_i are invertible in place is called a *no-fill partition*. A no-fill partition of L with the fewest factors is a *best no-fill partition*.

An *admissible permutation* Q of L is a symmetric permutation of the rows and columns of L such that the permuted matrix QLQ^T is lower triangular. A *best reordered partition* of L is a best no-fill partition of QLQ^T with the fewest factors over all admissible permutations Q of L.

Alvarado and Schreiber [2] have designed an algorithm requiring $O(n\tau)$ time and space to compute a best reordered partition of L. (Here n is the order of the matrix, and τ is the number of nonzeros in L.) We have designed an extremely efficient $O(n)$ time and space algorithm for computing the best reordered partition of a Cholesky factor L [6]. This algorithm makes use of the *elimination tree* data structure.

[*] Jan. 1993.

[†] Department of Computer Science, University of Waterloo, Waterloo, Ontario, N2L 3G1 Canada (apothen@narnia.uwaterloo.ca, na.pothen@na-net.ornl.gov).

For Cholesky factorization, recently we have designed two algorithms that compute the best reordered partition over all permutations Q that preserve the structure of $L + L^T$ [4, 5]. The first of these algorithms makes use of the usual representation of the structure of L, and takes time and space proportional to $O(n + \tau)$ to compute the partition. The second makes use of the more compact *clique tree* data structure, and requires time $O(n + q)$, where q is the size of the clique tree. For many practical problems, $q \ll \tau$.

All the partitioning problems are solved by considering graph models of these problems. These lead to problems of partitioning directed acyclic graphs (DAGs) and chordal graphs into the fewest transitively closed subgraphs, while satisfying a certain precedence relationship, over appropriate orderings of the graphs. While the graph partitioning problems are quite challenging, they have solutions in terms of 'greedy' algorithms. This work has led to new, interesting results about chordless paths in chordal graphs and the structure of vertex separators in these graphs.

We have implemented the partitioning algorithms for computing the best reordered partitions of L. We have also implemented a parallel triangular solution algorithm on the CM-2 based on this approach. Our results confirm the usefulness of these ideas in practice: the matrix-vector multiplication approach outperforms a conventional triangular solution algorithm by a wide margin on a Connection Machine CM-2. For the model problem ($n \times n$ square grid, optimal nested dissection ordering), the complexity of parallel triangular solution using the multiplication approach is $2\log_2 n$, while that of the substitution approach is $3n$.

Stability issues. We have also performed a stability analysis of this approach to triangular solution [3]. We have shown that this approach is normwise forward and backward stable when a certain scalar, that depends on the matrix L and the partition of L, is small; this scalar is guaranteed to be small when L is well-conditioned. (This scalar can be loosely described as a growth factor, since it is a measure of how elements grow in L^{-1}.) Moreover, when the factors of L are invertible in place (as we have chosen), then the backward error matrix has the same sparsity structure as L.

REFERENCES

[1] F. L. Alvarado, A. Pothen, and R. S. Schreiber, *Highly parallel sparse triangular solution*. Tech. Report CS-92-51, Computer Science, University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1, Oct. 1992. To appear in Sparse Matrix Computations: Graph Theory Issues and Algorithms, J. A. George, J. R. Gilbert and J. W. H. Liu (eds.), Springer Verlag. (IMA volumes in Mathematics and its Applications).

[2] F. L. Alvarado and R. S. Schreiber, *Optimal parallel solution of sparse triangular systems*. SIAM J. Sci. Comput., to appear, 1993.

[3] N. J. Higham and A. Pothen, *The stability of the partitioned inverse approach to parallel sparse triangular solution*. Tech. Report CS-92-52, Computer Science, University of Waterloo, Oct. 1992. Submitted to *SIAM J. Sci. Comput.*.

[4] B. W. Peyton, A. Pothen, and X. Yuan, *Partitioning a chordal graph into transitive subgraphs and parallel sparse triangular solution*. Tech. Report CS-92-55, Computer Science, University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1, Dec. 1992. Submitted to *Linear Algebra and its Applications*.

[5] B. W. Peyton, A. Pothen, and X. Yuan, *A clique tree algorithm for partitioning a chordal graph into transitive subgraphs*. Work in preparation, Jan. 1993.

[6] A. Pothen and F. L. Alvarado, *A fast reordering algorithm for parallel sparse triangular solution*. SIAM J. Sci. Stat. Comput., 13 (1992), pp. 645–653.

ITERATIVE SOLUTION TECHNIQUES FOR THE NAVIER-STOKES EQUATIONS

ALISON RAMAGE* AND ANDY WATHEN †

The efficient solution of large systems of linear (or linearised) equations is of great practical importance in computational fluid dynamics as it is in many other areas of scientific computing. A particular example frequently encountered in industrial applications is the case of the Navier-Stokes and continuity equations for incompressible viscous fluid flow. Applying the usual mixed finite element method and a standard nonlinear solver (for example the Newton-Raphson method) reduces this problem to a series of nonsymmetric linear systems. These could be tackled with a non-symmetric iteration: here, however, we adopt a different approach and take advantage of the symmetry which arises naturally in a Lagrange-Galerkin finite element discretisation.

The Lagrange-Galerkin method is a numerical technique introduced as an accurate discretisation method for advection-dominated diffusion problems (see for example [1]). When applied directly to the time-dependent Navier-Stokes equations, it removes the nonlinearity introduced by the advection term in such a way that the resulting linear system is of the form

$$(1) \qquad \begin{bmatrix} A & B \\ B^T & 0 \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix},$$

that is, it has the character of the Stokes problem for creeping flow. Here u and p are the vectors of velocity and pressure unknowns respectively. The matrix A is symmetric and positive definite and, under the assumption that the finite elements chosen satisfy the Babuska-Brezzi stability condition, B is of full rank.

Various iterative methods which have been proposed for Stokes problems can be applied here and in this paper we compare the performance of two different types. Firstly, we consider a traditional pressure correction approach. This is equivalent to applying block elimination to decouple the pressure and velocity equations, forming two nested symmetric positive definite systems which can each be solved using the preconditioned conjugate gradient method. This two-level pressure correction technique is compared with solution of the original fully-coupled indefinite problem by means of the preconditioned conjugate residual (MINRES) method. The question of inner/outer convergence crtiteria which has only been answered for simpler iterative methods ([3]) is avoided in this one-level iterative approach.

* Department of Mathematics, University of Strathclyde, Glasgow G1 1XH, United Kingdom.

† School of Mathematics, University of Bristol, Bristol BS8 1TW, United Kingdom.

The relative performance of these algorithms will be examined both in theory and in practice. Using eigenvalue estimates found by way of polynomial approximation problems on the eigenvalue spectrum [2], asymptotic estimates for the amount of work involved in implementing each method with various preconditioners can be calculated: here we will present results for preconditioners based on diagonal scaling and modified incomplete Cholesky factorisation. These estimates strongly favour the use of the MINRES single level iteration method.

For 'optimal' preconditioners which can be employed in the pressure-correction iterative approach, the corresponding preconditioned MINRES iteration also has a convergence rate which is optimal, i.e. independent of any discretisation parameter.

In addition to theoretical estimates, we will present the results of practical numerical computations carried out within the framework of an industrial fluid flow simulation program (Nuclear Electric plc.'s code FEAT). These experiments confirm the superiority of the MINRES method.

REFERENCES

[1] Morton, K.W. and Priestley, A. 'On Characteristic and Lagrange-Galerkin Methods', *Pitman Research Notes in Mathematics Series*, D.F. Griffiths and G.A. Watson, eds, Longman Scientific and Technical, 1986.

[2] Wathen, A.J.,Fischer, B. and Silvester, D.J. 'On Convergence of Conjugate Residuals for the Stokes Problem', *in preparation*.

[3] Golub, G.H. and Overton, M. 'The Convergence of Inexact Chebyshev and Richardson Iterative Methods for Solving Linear Systems', *Numer. Math.*, 53, 571-594 (1988).

Generalized ADI Iteration
Lothar Reichel

February 1, 1993

Abstract

We consider the solution of Sylvester's equation $AX - XB = C$, where $A \in R^{n\times n}$, $B \in R^{m\times m}$, and $C \in R^{n\times m}$ are given matrices, and $X \in R^{n\times m}$ is the solution matrix to be determined. The ADI iterative method for the solution of Sylvester's equation proceeds by strictly alternating between the solution of the two equations

$$\begin{aligned}(A-\delta_k I)X_{k+1} &= X_k(B-\delta_k I)+C,\\ X_{k+2}(B-\tau_k I) &= (A-\tau_k I)X_{k+1}-C,\end{aligned}$$

for $k = 0, 1, 2, \ldots$. Here X_0 is a given initial approximate solution, and the δ_k and τ_k are real or complex parameters chosen so that the computed approximate solutions X_k converge rapidly to the solution of the Sylvester equation as k increases. We will discuss the possibility of solving one of the equations in the ADI iterative method more often than the other one, i.e., we relax the strict alternation requirement, in order to achieve a higher rate of convergence. Our analysis based on potential theory shows that this generalization of the ADI iteration method can give faster convergence than when strict alternation is required. We will pay particular attention to a special Sylvester's equation that arises when applying a Wiener filter to reduce noise in images. The talk presents joint work with Daniela Calvetti and Norm Levenberg.

ITERATIVE SOLUTION OF RECTANGULAR SYSTEMS

Michael A. SAUNDERS
Department of Operations Research
Stanford University
Stanford, CA 94305-4022, USA

Background

LSQR is a conjugate-gradient-like method for solving the least-squares problem

$$\min \|Ax - b\|, \tag{1}$$

while Craig's method finds the minimum-length solution to under-determined (but compatible) systems:

$$\min \|x\| \quad \text{subject to} \quad Ax = b. \tag{2}$$

Both methods use the Golub-Kahan bidiagonalization of A, with b as the starting vector (see Paige and Saunders, ACM TOMS, 1982). When A is square and nonsingular, problems (1) and (2) are the same. Craig's method is then slightly simpler and more efficient. Otherwise the two methods solve separate problems and cannot be compared.

Damping or Regularization

LSQR also solves the damped least-squares problem

$$\min \|Ax - b\|^2 + \|\delta x\|^2 \quad \equiv \quad \min \left\| \begin{pmatrix} A \\ \delta I \end{pmatrix} x - \begin{pmatrix} b \\ 0 \end{pmatrix} \right\|, \tag{3}$$

where δ is typically a small scalar parameter that regularizes the problem if A is singular or ill-conditioned. Almost no additional work or storage are needed to incorporate damping.

In order to extend Craig's method to incompatible systems, we have studied the problem

$$\begin{aligned} &\min \|x\|^2 + \|s\|^2 \quad \text{subject to} \quad Ax + \delta s = b, \\ \equiv\ &\min \left\| \begin{pmatrix} x \\ s \end{pmatrix} \right\| \quad \text{subject to} \quad \begin{pmatrix} A & \delta I \end{pmatrix} \begin{pmatrix} x \\ s \end{pmatrix} = b. \end{aligned} \tag{4}$$

Some additional work and storage are required, but the extended method is straightforward and stable.

A Surprise Equivalence

Problems (3) and (4) are both well-defined for any A as long as δ is nonzero. Indeed, (3) and (4) are unexpectedly the *same problem* when $\delta \neq 0$.

The extended LSQR and Craig algorithms may therefore be compared. We explore the differences from this viewpoint.

"BiCGstab(ell): an efficient and stable solver for equations involving matrices with complex spectrum"

Gerald L.G. Sleijpen

(joint work with D. Fokkema and H.A. van der Vorst
supported in part by a NCF Cray Research University
Grant, CRG 92.03)

ABSTRACT

For a number of linear matrix-vector equations stemming from realistic problems, the BiCGstab algorithm of van der Vorst [4] to solve these equations is an efficient one. Unfortunately, specifically in case of discretized advection dominated PDE's, BiCGstab stagnates due to the fact that for this type of equations the matrix has (almost) pure imaginary eigenvalues. Here, we generalize the BiCGstab algorithm and get rid of the stagnation. The BiCGstab(ell) algorithm in [3] generalizes BiCGSTAB and gets rid of the stagnation.

In exact arithmetic, if no breakdown occurs, our BiCGstab(2) algorithm is equivalent to the BiCGStab2 algorithm of Gutknecht [1]. However, our version is more flexible, more efficient, less sensitive to evaluation errors and is less likely to suffer from breakdown.

Schemes of BiCGstab(ell) type combines Bi-CG and ell steps of some minimal residual method as ORTHODIR or GMRES [3]. These method profit from both of its components.

[1] M.H. Gutknecht.
Variants of BiCGStab for matrices with complex Spectrum.
IPS Research Report, No 91-14, 1991.

[2] G.L.G. Sleijpen and D.R. Fokkema.
Bi-CGstab(ell) for linear equations involving matrices
with complex spectrum.
Preprint 772, Dep. Math, Utecht University,
Uterecht, 1993.

[3] G.L.G. Sleijpen, D.R. Fokkema and H.A. van der Vorst.
Bi-CGstab(pol); a class of efficient solvers of large
systems of linear equations.
Preprint, 1993.

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

DERIVING GENERAL SPARSE SYMMETRIC AND QUASI POSITIVE DEFINITE QUASI-NEWTON UPDATES VIA THE ABS APPROACH

Emilio Spedicato, University of Bergamo

ABS algorithms have been introduced in 1984 by Abaffy, Broyden and Spedicato for solving determined or underdetermined linear equations. They have been then extended to the solution of linear least squares, nonlinear equations and work is in progress for their application to optimization and eigenvalue problems, the number of papers in this area approaching now two hundred. An interesting application of the ABS technique is the determination of the general solution of the secant or Quasi-Newton equation $B'd=y$ appearing in the formulation of Quasi-Newton methods for nonlinear equations or nonlinear optimization, coupled possibly with the additional conditions of sparsity, symmetry and positive definiteness. The given conditions constitute an underdetermined linear systems in the unkown B' whose general solution can be expressed in a closed and simple form, even when sparsity and symmetry are considered, using the ABS representation of the linear variety containing all solutions of an underdetermined linear system. The resulting general formula, from which all previously considered formulas are easily obtainable, leads naturally to consider some new formulas, e.g. a continuous "dogleg" update. The analysis of the general formula when sparsity and symmetry are present shows that, while positive definiteness cannot generally be forced, under a very mild condition on the sparsity pattern it is always possible to construct, in a computationally simple way, formulas which are quasi positive definite, in the sense that the first $n-1$ principal submatrices in B' are positive definite. A new formulation is also given of the Marwil-Toint formula.

Hybrid and Adaptive Polynomial Iterative Methods for Preconditioned Nonsymmetric Systems of Linear Equations

Gerhard Starke

Institut für Praktische Mathematik
Universität Karlsruhe
Englerstrasse 2
7500 Karlsruhe 1
Germany

In recent years, a lot of progess has been made in the field of polynomial iterative methods for large nonsymmetric systems of linear equations,

$$A\mathbf{x} = \mathbf{b}\,.$$

Most of the current research deals with extensions of the conjugate gradient method to matrices which are not positive definite [1]. There are, however, also reasons for studying Chebyshev-type algorithms, sometimes also called semi-iterative methods. While CG-type methods are parameter-free, Chebyshev-like methods require the computation of the iteration polynomial beforehand based on some information on the underlying linear system.

One of the motivations to study Chebyshev-type methods comes from the fact that inner products — which are an essential part of any CG-type algorithm — constitute a potential bottleneck on certain parallel computer architectures. Furthermore, CG-type methods for nonsymmetric systems, like (restarted) GMRES and QMR, have been noted to be susceptible to stagnation of the convergence behavior. The goal is to avoid this with a Chebyshev-type method using a properly chosen iteration polynomial. In practice, Chebyshev-type methods are implemented as hybrid schemes which consist of a beginning phase where information about the matrix A is acqüired, and a second phase where a polynomial iteration designed with respect to this information is carried out. Closely related, and sometimes advantageous, is the approach to use the iteration polynomial as a polynomial preconditioner for a CG-type method.

We present new techniques for both phases of a hybrid method for nonsymmetric linear systems. For Phase I, eigenvalue estimates constructed from approximations to the fields of values of A and A^{-1} turn out to be promising. The fields of values are approximated in a low-dimensional Krylov subspace associated with A using the orthonormal basis constructed by the Arnoldi process. We prove that, in any case, the Arnoldi Ritz values are contained in the approximate field of values corresponding to A, $W_n(A)$. Similarly, the roots of the GMRES residual polynomial are contained in $1/W_n(A^{-1})$. The resulting polygonal set in the complex plane is guaranteed to exclude the origin and, in addition, to include the entire spectrum of A if the dimension of the Krylov subspace is large enough. Under the assumption that the complement of this set is simply connected, i.e., the origin is not enclosed by eigenvalue estimates, this ensures the existence of a convergent polynomial iteration, e.g. one based on Chebyshev or Faber polynomials, for Phase II [3, 2].

For practical problems, it is essential that iterative schemes are combined with preconditioning in order to achieve faster convergence. The development of efficient preconditioning strategies can benefit heavily from the underlying physical model. In order to be more specific, we investigate preconditioning of finite element discretizations of second order elliptic boundary value problems by domain decomposition. In particular, we report on some computational experiments with substructuring algorithms based on non-overlapping subdomains in connection with Chebyshev-like iterations.

References

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

[2] T. A. Manteuffel and G. Starke. On hybrid iterative methods for nonsymmetric systems of linear equations. 1993. In preparation.

[3] G. Starke and R. S. Varga. A hybrid Arnoldi-Faber method for nonsymmetric systems of linear equations. *Numer. Math.*, 1993. To appear.

Optimal Backward Perturbation Bounds for Certain Matrix Computation Problems

Ji-guang Sun
Institute of Information Processing
University of Umeå
S-901 87 Umeå, Sweden

ABSTRACT

It is known that backward perturbation analysis is a very fruitful method in matrix perturbation theory, and many problems in this field remain to be solved (ref. [1], [2], [4]-[6], [10]). This talk will present a certain number of new results related to the linear least squares problem, the eigenvalue problem, the generalized eigenvalue problem, and the singular value problem.

1. The linear least squares problem

Let A be an $n \times n$ matrix, b be an m-vector, and $\tilde{x}$ be a computed solution to the problem of minimizing $\|b - Ax\|_2$. We consider the following open problem [2],[5]: to find the smallest perturbation E of A such that the vector $\tilde{x}$ exactly minimizes $\|b - (A + E)\tilde{x}\|_2$. This problem is completely solved when E is measured in the Frobenius norm [7], [9]:

Let $\tilde{x}$ be given, and let $\hat{r} = b - A\tilde{x}$. Let λ_* be the smallest eigenvalue of

$$H \equiv AA^{\mathrm{H}} - \frac{\hat{r}\hat{r}^{\mathrm{H}}}{\|\tilde{x}\|_2^2},$$

and let u_* be a unit eigenvector of the matrix H belonging to λ_*. Then the matrix

$$E_* = \begin{cases} \hat{r}\tilde{x}^{\dagger} & \text{if } \lambda_* \geq 0 \\ (I - u_* u_*^{\mathrm{H}})\hat{r}\tilde{x}^{\dagger} - u_* u_*^{\mathrm{H}} A & \text{if } \lambda_* < 0 \end{cases}$$

is the smallest matrix measured in the Frobenius norm that $\tilde{x}$ exactly minimizes $\|b - (A + E_*)\tilde{x}\|_2$, and

$$\|E_*\|_F = \begin{cases} \frac{\|\hat{r}\|_2}{\|\tilde{x}\|_2} & \text{if } \lambda_* \geq 0 \\ \left[\left(\frac{\|\hat{r}\|_2}{\|\tilde{x}\|_2}\right)^2 + \lambda_*\right]^{1/2} & \text{if } \lambda_* < 0. \end{cases}$$

The above result is an extension of the Rigal-Gaches Theorem on backward perturbations of the system $Ax = b$ (see [4]).

2. Certain Characteristic Subspaces

Let A be an $n \times n$ matrix. Let $\tilde{X}_1$ be an $n \times l$ matrix whose column vectors are orthonormal and span a subspace $\tilde{\mathcal{X}}_1$ which approximately is invariant for A. An important problem is that what is the smallest matrix E measured in any unitarily invariant norm for which $\tilde{\mathcal{X}}_1$ is an invariant subspace of $A + E$ and how small it could be. The solution to this problem is given [8]

In the similar way, we study similar questions for eigenproblems of Hermitian matrices, generalized eigenproblems of a regular matrix pair, and singular value problems of a general matrix [8].

Further, applying our results on backward perturbation analysis for certain characteristic subspaces, we derive residual bounds for certain eigenvalues, generalized eigenvalues, and singular values. For instance, we get the following result [8]:

Let A be an $n \times n$ Hermitian matrix, $\tilde{X}_1$ be an $n \times l$ matrix whose column vectors are orthonormal and span a subspace $\tilde{\mathcal{X}}_1$ which approximately is invariant for A. Let

$$\tilde{A}_1 = \tilde{X}_1^{\mathrm{H}} A \tilde{X}_1, \qquad \hat{R} = \tilde{X}_1 \tilde{A}_1 - A\tilde{X}_1.$$

If the eigenvalues of A are $\lambda_1 \geq \cdots \geq \lambda_n$, and the eigenvalues of $\tilde{A}_1$ are $\tilde{\lambda}_1 \geq \cdots \geq \tilde{\lambda}_l$, then there are integers $i_1 < i_2 < \cdots < i_l$ such that for any unitarily invariant norm $\| \|$,

$$\|\mathrm{diag}(\tilde{\lambda}_1 - \lambda_{i_1}, \ldots, \tilde{\lambda}_l - \lambda_{i_l})\| \leq \left\| \begin{pmatrix} \hat{R} & 0 \\ 0 & \hat{R}^{\mathrm{H}} \end{pmatrix} \right\|. \tag{1}$$

Moreover, let $\tilde{X}_2$ be an $n \times (n - l)$ matrix making $(\tilde{X}_1, \tilde{X}_2)$ unitary, and let $\tilde{A}_2 = \tilde{X}_2^{\mathrm{H}} A \tilde{X}_2$. If

$$\delta_2 \equiv \mathrm{sep}_2(\tilde{A}_1, \tilde{A}_2) > 0, \qquad \eta_2 \equiv \frac{2\|\hat{R}\|_2}{\delta_2} < 1, \tag{2}$$

then there are integers $i_1 < i_2 < \cdots < i_l$ such that for any unitarily invariant norm $\| \|$,

$$\|\mathrm{diag}(\tilde{\lambda}_1 - \lambda_{i_1}, \ldots, \tilde{\lambda}_l - \lambda_{i_l})\| \leq \frac{2}{\sqrt{1 - \eta_2^2}} \frac{\|\hat{R}\|_2 \|\hat{R}\|}{\delta_2}. \tag{3}$$

Kahan [3] proved that the inequality (1) holds for the spectral norm and the Frobenius norm. The inequality (3) shows that taking into account the situation of the spectrum (2), we get a new bound of order $\|\hat{R}\|^2$.

References

[1] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd Edition, The Johns Hopkins University Press, Baltimore, Maryland 1989.

[2] N. J. Higham, *Computing error bounds for regression problems*, Numerical Analysis Report No. 179, Department of Mathematics, University of Manchester, December 1989.

[3] W. M. Kahan, *Inclusion theorems for clusters of eigenvalues of Hermitian matrices*, Technical Report, Computer Science Department, Unversity of Toronto, 1967.

[4] J. L. Rigal and J. Gaches, *On the compatibility of a given solution with the data of a linear system*, J. Assoc. Comput. Mach., 14(1967), 543-548.

[5] G. W. Stewart, *Research, development, and LINPACK*, in *Mathematical Software III*, J. R. Rice, ed., Academic Press, New York, 1977, 1-14.

[6] G. W. Stewart and J. -G. Sun, *Matrix Perturbation Theory*, Academic Press, New York 1990.

[7] J. -G. Sun, *An improved backward perturbation bound for the linear least squares problem* (manuscript), 1991.

[8] J. -G. Sun, *Backward perturbation analysis of certain characteristic subspaces*, submitted to Numer. Math.

[9] B. Waldén, R. Karlson and J. -G. Sun, *Optimal backward perturbation bounds for the linear least squares problem*, Technical Report, LiTH-Mat-R-1992-06, Department of Mathematics, Linköping University, 1992.

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

Inverse Iteration Method With A Complex Parameter

TOSHIO SUZUKI

Department of Mathematics, Yamanashi University

In the paper[1] we introduced into the inverse iteration method for symmetric matrices a new technique, which is simple but effective in practical computaions. Here I would like to talk about our new method of [1] and some propositions which suggest us some applications of it.

Let A be a (n,n) matrix which is symmetric and has n different eigenvalues. Let $\lambda_k, \phi_k, k = 1, 2, \ldots, n$, be pairs of eigenvalues and the corresponding normalized eigenvectors of A. First we describe our method to compute the eigenvector ϕ_j corresponding the eigenvalue λ_j under the following assumption.

Assumption H. Eigenvalues $\lambda_k, k = 1, 2, \ldots, n$ of A are known with the following accuracy: There are three numerical constants c, ε and λ such that $\inf_{k\neq j} |\lambda_j - \lambda_k| > 2c$ *and* $|\lambda_j - \lambda| < \varepsilon$ *and* $0 < 2\varepsilon < c$.

Let ξ be an initial vector and let τ be a real number whose absolute value is smaller than c. Our iteration process proposed in [1] consists of the following three steps where $u^{(m)}$ and $v^{(m)}$ are real vectors.

$$(1) \qquad (A - \lambda^{(m)}I - \sqrt{-1}\tau I)w^{(m)} = z^{(m)} \quad \text{where } z^{(0)} = \xi, \lambda^{(0)} = \lambda$$

$$(2) \qquad z^{(m+1)} = \frac{v^{(m)}}{\|v^{(m)}\|} \quad \text{where } w^{(m)} = u^{(m)} + \sqrt{-1}v^{(m)}$$

$$(3) \qquad \lambda^{(m+1)} = \begin{cases} (Az^{(m+1)}, z^{(m+1)}) & \text{if } \|v^{(m)}\| > \|u^{(m)}\| \\ \lambda^{(m)} & \text{othewise.} \end{cases}$$

The most essential and characteristic feature of our process is the second step(2) where the imaginary part of the solution of the linear equation (1) is taken as an approximating eigenvector. In third step (3), we change the value λ to a better approximating value, obtained by Raileigh-Rits formula where the inequality $|\lambda_j - \lambda^{(m+1)}| < \varepsilon$ also holds as is seen in proposition 3 later. The following theorem guarantees that this iteration process works well.

THEREM 1. *(Theorem 2.1 of [1]) If the assumption H is satisfied, the iteration process (2.1)-(2.3) excites the component of the eigenvector ϕ_j,namely $z^{(m)} \to \pm\phi_j, m \to \infty$,provided $|\tau| < \varepsilon$.*

Before we state the theorem of the error estimates and some propositions, we need some prepararions to simplify the notations. Consider the following equation with $\|z\| = 1$:

$$(4) \qquad (A - \lambda I - \sqrt{-1}\tau I)w = z$$

Let $z = \sum_{k=1}^{n} a_k\phi_k$, then we have

$$w = \sum_{k=1}^{n} \frac{1}{\lambda_k - \lambda - \sqrt{-1}\tau} a_k\phi_k$$
$$= \sum_{k=1}^{n} \frac{\lambda_k - \lambda}{(\lambda_k - \lambda)^2 + \tau^2} a_k\phi_k + \sqrt{-1}\sum_{k=1}^{n} \frac{\tau}{(\lambda_k - \lambda)^2 + \tau^2} a_k\phi_k$$

Put $x_k = \frac{\lambda_k - \lambda}{\lambda_k - \lambda^2 + \tau^2} a_k\phi_k$ and $y_k = \frac{\tau}{\lambda_k - \lambda^2 + \tau^2} a_k\phi_k$. Let $x = \sum_{k=1}^{n} x_k$ and $y = \sum_{k=1}^{n} y_k$.

THEOREM 2. *(Theorem 2.2 of [1]) Put $\delta = \frac{\|x\|}{\|y\|}$. Under the assumption H, the relative error $\|y - y_j\|/\|y\|$ is estimated as*

$$\frac{\|y - y_j\|}{\|y\|} \leq \frac{\tau}{c}\delta.$$

The following proposition shows that in the iteration process (1)-(3) the inequality $|\lambda_j - \lambda| < \varepsilon$ in the assumption H continues to hold after the approximating eigenvalues are replaced in (3) under the more relaxed criterion than that in (3).

PROPOSITION 3. *Let x,y be the real and imaginary part of the solution of the equation(4) with $|\lambda_j - \lambda| < |\tau| \leq \varepsilon$ under the assumption H in which the inequality $|\lambda_j - \lambda| < |\varepsilon|$ is assumed. Put $\tilde{\lambda} = (Ay, y)/\|y\|^2$. If $3\|y\| > 2\|x\|$, then $|\lambda_j - \tilde{\lambda}| < |\tau|$.*

Proposition 3 gives a criterion when the approximating value λ may be replaced by a better approximating one. Moreover, through the following Proposition 4, we have another criterion when the complex parameter τ can be replaced by a smaller one if it is neccessary.

PROPOSITION 4. *Under the same assumption of Proposition 3, if $\|y\| \geq \|x\|$ then $|\lambda_j - \tilde{\lambda}| < \frac{2\tau^2}{c}$.*

Considering Proposition 3 and 4, we have an improvement of the iteration process(1)-(3) by replacing the step(3) with the following (5) and by adding the next process(6).

$$(5) \qquad \lambda^{(m+1)} = \begin{cases} (Az^{(m+1)}, z^{(m+1)}) & \text{if } 3\|v^{(m)}\| > 2\|u^{(m)}\| \\ \lambda^{(m)} & \text{otherwise.} \end{cases}$$

$$(6) \qquad \tau^{(m+1)} = \frac{2(\tau^{(m)})^2}{c} \quad \text{if } \|v^{(m)}\| > \|u(m)\|.$$

Proposition 3 and 4 show that even if we do not have a so accurate value of ε or even if the initial vector is not so well,$\lambda^{(m)}$ in the improved process converges to the aimed eigenvalue efficiently by using better parameters in each iteration. So we can have an application of our method to get a rapid tool for computing eigenpairs combining the bisection method. The idea of it is such that: get rough estimates of eigenvalues by the bisection method,first, then, apply our iteration process. The computing time to improve the accuracy of an eigenvalue by 5 decimal digits with the aid of the bisection method is comparable to that of two times iterations of our method. So, for example, if ,starting from the initial approximating value with the accuracy about 10^{-4}, we could have the eigenvalue with the accuracy 10^{-15} after two times iterations, this method is an improvement of the procedure done by only the bisection method. The test computations of this example and of the others of this kinds have shown satisfactory results. We do not have the optimal of it yet but the above example is at least one of the application of our method to get eigen-pairs more rapidly.

REFERENCES

1. Suzuki T., *Inverse Iteration Method with a Complex Parameter*, Proceedings of THE JAPAN ACADEMY, 68 Ser A No.3 (1992), 68-73.

Two-stage Iterative Methods

Daniel B. Szyld
Department of Mathematics
Temple University

Abstract

A talk is proposed in which two-stage iterative methods for the solution of linear systems of the form $Ax = b$ are discussed. These are iterative methods in which the linear system at each (outer) iteration is solved in turn by an (inner) iterative method. If the number of inner iteration (s) is fixed, it is called a stationary method. If the number of inner iterations changes at each outer step ($s = s(k)$), then it is a non-stationary method. We will review the literature including results by Nichols [5], Golub and Overton [2], [3] among others, as well the work with coauthors [4], [7], [1]. We will also present more recent results including computational experiments illustrating the applicability of the methods.

During the talk, convergence of the non-stationary method will be shown if the number of inner iterations becomes sufficiently large. The R_1-factor (σ) of the two-stage method is related to the spectral radius of the iteration matrix of the outer splitting. In addition the following results holds.

Theorem 1 *Let $A = M - N$ and $M = F - G$ be convergent splittings. Let $\|\cdot\|$ be any operator norm such that $\|M^{-1}N\| < 1$. Let $\bar{s} \in \mathbb{N}$ be such that*

$$\|(F^{-1}G)^s\| \leq q < \frac{1 - \|M^{-1}N\|}{1 + \|M^{-1}N\|} \quad \text{for all } s \geq \bar{s}.$$

Assume that $\liminf_{k\to\infty} s(k) > \bar{s}$ and let $\{x_k\}_{k=0}^{\infty}$ be the sequence generated by a two-stage method with a given initial guess x_0. Then

(i) $\lim_{k\to\infty} x_k = x^* \ (= A^{-1}b)$,

(ii) $\sigma(\{x_k\}_{k=0}^{\infty}) \leq q + (1+q)\|M^{-1}N\| < 1$.

In applications it is not practical to perform large number of inner iterations, i.e., to wait until the assimptotic convergence is attained. The question is then, how does the method converge when a small number of inner iterations take place. Examples will be shown in which for few inner iterations the two-stage method fails to converge.

Conditions of the splittings are given that guarantee convergence for *any* number of inner iterations. If the matrix A is monotone, i.e., $A^{-1} \geq 0$, the appropriate conditions are that the outer splitting is a regular splitting and the inner splitting is a weak regular splitting. Similar conditions are obtained for splittings of H-matrices. These matrices are not necessarily monotone.

Block two-stage methods will also be analyzed, in which the linear system in each diagonal block, e.g. when the outer method is block Jacobi, is solved iteratively. This case is particularly applicable to parallel computations. Some of its convergence properties can be studied using the theory of convergence of the multisplitting method of O'Leary and White [6].

Computational results in sequential and parallel machines will be reported. In particular, the optimal number of inner iterations will be shown for specific examples.

References

[1] Andreas Frommer and Daniel B. Szyld. H-splittings and two-stage iterative methods. *Numerische Mathematik*, 63:345–356, 1992.

[2] Gene H. Golub and Michael L. Overton. Convergence of a two-stage Richardson iterative procedure for solving systems of linear equations. In G.A. Watson, editor, *Numerical Analysis (Proceedings of the Ninth Biennial Conference, Dundee, Scotland, 1981), Lecture Notes in Mathematics 912*, pages 128–139, New York, 1982. Springer Verlag.

[3] Gene H. Golub and Michael L. Overton. The convergence of inexact Chebyshev and Richardson iterative methods for solving linear systems. *Numerische Mathematik*, 53:571–593, 1988.

[4] Paul J. Lanzkron, Donald J. Rose, and Daniel B. Szyld. Convergence of nested classical iterative methods for linear systems. *Numerische Mathematik*, 58:685–702, 1991.

[5] Nancy K. Nichols. On the convergence of two-stage iterative processes for solving linear equations. *SIAM Journal on Numerical Analysis*, 10:460–469, 1973.

[6] Diane P. O'Leary and Robert E. White. Multi-splittings of matrices and parallel solution of linear systems. *SIAM Journal on Algebraic and Discrete Methods*, 6:630–640, 1985.

[7] Daniel B. Szyld and Mark T. Jones. Two-stage and multisplitting methods for the parallel solution of linear systems. *SIAM Journal on Matrix Analysis and Applications*, 13:671–679, 1992.

Parallelizing Linear Algebra on the KSR1

Anne E. Trefethen
Cornell Theory Center

January 31, 1993

Introduction

We present the serial and parallel implementations of a complex matrix-vector multiply on the KSR1. The implementation is in Fortran and we attempt to illustrate the performance of the KSR on such linear algebra kernels and also some of the issues that arise when parallelizing Fortran code on this machine. We have chosen this example as it simple but effective at illustrating some of the features of the architecture and we believe gives a good indication of issues for higher lever linear algebra routines.

We give timings which indicate the scalabilty of the architecture for these types of algorithms. We also discuss problem areas for the development of modular software on this machine.

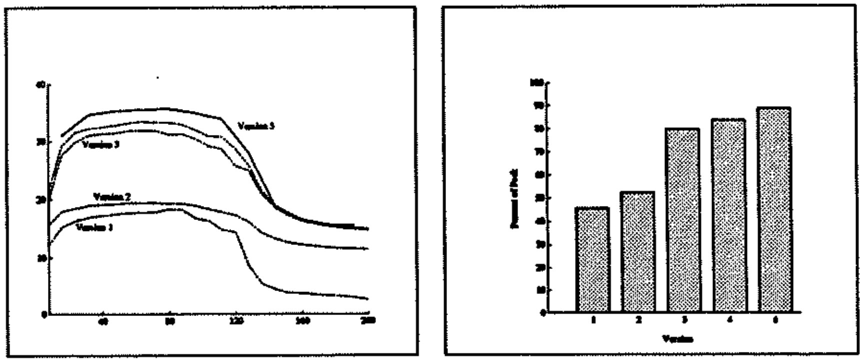
The KSR1

The KSR1 is a virtual, shared-memory parallel machine. It comprises individual processors each with its own memory, but through the "ALLCACHE" memory system it is presented to the user as shared memory. We show that in this shared memory environment it is essential to consider the cache hierarchy in order to obtain good performance and, as might be expected, data locality is very important.

One model for parallelizing a code on the KSR1 is to optimize the serial code on one node and then to parallelize that code. So we first address the question of how to obtain high performance on a single node.

The Serial Implementation

Figure 1a illustrates the Mflop performance for five versions of the code. Versions 1 and 2 are written in terms of complex arithmetic and differ only by loop reordering (as indicated by the KAP preprocessor supplied by KSR); version 3 is version 2 with the arithmetic split into real components, version 4 is a hand unrolled version 3, and version 5 is a hand unrolled version 1 (i.e. ignoring the advice of the preprocessor). The plots are of Mflops versus n, where the matrix and vector are of sizes $n \times n$ and $n \times 1$ respectively.



1a 1b
Serial performance of complex matrix-vector multiply

Figure 1b illustrates the percentage of peak performance obtained by each of the versions.

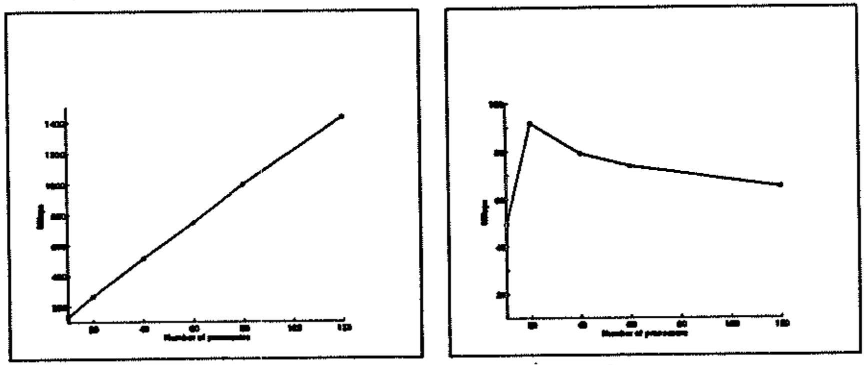
Clearly all of the serial implementations shown have the problem that although they have a relatively good peak performance, this is not sustained beyond the size of the subcache. Even for the best implementation the performance drops to below 50% of the peak performance of the floating point unit. One might think that a better strategy for blocking the data would yield a higher performance. However, this is not the case and we cannot expect to achieve much higher performance than shown, for data sizes beyond the size of the subcache.

The Parallel Implementation

There are four techniques for parallelizing a code on the KSR1: pthreads, parallel regions, parallel sections and tiles. We show how each applies to our algorithm and the performance that might be expected from each. The best parallel performance is gained by using the parallel region construct which allows the execution of a single piece of code on multiple processors. This feature allows the SPMD (Single Program Multiple Data) programming style or concurrent computation on blocks of the data.

In our example we can utilize this form by considering the matrix vector multiply block algorithm. We split the matrix into blocks of rows and also split the vector y into the assosiated sections. The idea is to use the serial code on each processor and have it operate on a block of the matrix.

Figure 2 shows the Mflop rates for the parallelized version 5, using parallel regions. These timings indicate that the locality of the data on calling the matrix vector multiply routine affects the performance enormously. Figure 2a shows the average Mflop rate of 100 calls to the routine for a problem size of (15360,128) when the data is already in the appropriate caches. There is linear speedup.



2a

2b

Parallel performance of complex matrix-vector multiply

Figure 2b shows the timings for the same piece of code with the same data size but the data is not resident in the appropriate subcache. The results shown are the worst case where the data are accessed in a completely orthogonal fashion by the same team of threads. There is far from linear speedup; in fact as the number of processors increase the performance begins to drop off. If we were to shift data around in this manner on any distributed memory machine we would expect to see similar results. However, the problem here is that there is no way of knowing where the data resides and the movement of data is done by the memory manager, not the user. There are prefetch and poststore facilities that may be used to ease the problem.

Conclusions

We show that getting a code to run in parallel fairly well on the KSR1 is reasonably simple using the tools provided. To obtain better performance may require the use of constructs not included in the preprocessing tools and in some cases ignoring advice given by the preprocessing tool.

Our results indicate that:

- at the present time complex arithmetic should be avoided when possible.
- For algorithms with the equivalent potential for data reuse, one can only expect to obtain 80 to 90% of the peak performance of a single node when the data is within the size of the subcache and at most 50% of the single node performance otherwise.
- As with any cache based (or distributed memory) machine, data locality is very important.
- For "nicely" located data our algorithm scales linearly with the number of processors; otherwise the results are quite different. This could mean problems for the design of modular routines.

We comment on how these conclusions relate to other parallel machines, in particular, to the CM5.

Pseudozeros of Polynomials and Pseudospectra of Companion Matrices

Lloyd N. Trefethen
Department of Computer Science
Cornell University
lnt@cs.cornell.edu

Kim-chuan Toh
Field of Applied Mathematics
Cornell University
kc@macomb.tn.cornell.edu

Zeros of polynomials and eigenvalues of nonsymmetric matrices are well-known examples of problems whose answers may be highly sensitive to perturbations. The sensitivity of these two problems was made famous by Wilkinson in the early 1960s and contributed to his development of the notions of stability and conditioning. And, of course, the two problems are related, for the zeros of a polynomial are the same as the eigenvalues of the associated companion matrix.

Despite the classical nature of the subject, the relationship between these two problems has received less study than one might suppose. Polynomial zerofinding has been something of a backwater in numerical analysis, and it is probably fair to say that although all numerical analysts know that one can find zeros via companion matrices in principle, most assume that it isn't a good idea to do so. The goal of our work has been to take a new look at these problems and see whether the use of companion matrices is or is not a good idea.

The approach we have taken is geometric. For a monic polynomial $p(z)$, let $Z(p)$ denote the zero set (= set of zeros) of $p(z)$ in the complex plane, and define the ϵ**-pseudozero set** of $p(z)$ by

$$Z_\epsilon(p) = \{z \in \mathbf{C} \colon z \in Z(\tilde{p}) \text{ for some } \tilde{p}\},$$

where $\tilde{p}$ ranges over polynomials whose coefficients are those of p modified by relative perturbations of size $\leq \epsilon$. (Precise details are omitted from this abstract.) The relevance of such sets to the conditioning of the zerofinding problem has been studied by Mosier. Analogously, for a square matrix A, let $\Lambda(A)$ denote the spectrum of A, and define the ϵ**-pseudospectrum** of A by

$$\begin{aligned}\Lambda_\epsilon(A) &= \{z \in \mathbf{C} \colon z \in \Lambda(A+E) \text{ for some } E \text{ with } \|E\| \leq \epsilon\} \\ &= \{z \in \mathbf{C} \colon \|(zI-A)^{-1}\| \geq \epsilon^{-1}\}.\end{aligned}$$

(The matrix $(zI-A)^{-1}$ is known as the resolvent; if $zI-A$ is singular, we define $\|(zI-A)^{-1}\| = \infty$.) Matrix pseudospectra have been studied by Trefethen, Godunov, and others going back at least to H. J. Landau in 1975.

We have found that for most polynomials, $Z_\epsilon(p)$ and $\Lambda_\epsilon(A)$ are quite close to one another when A is a companion matrix of p that has been "balanced" in the usual EISPACK sense proposed originally by Parlett and Reinsch. It follows that the conditions of the polynomial zerofinding problem and the balanced matrix eigenvalue problem are comparable. Therefore, finding zeros via eigenvalues of companion matrices ought to be a stable algorithm. These results are empirical as applied to arbitrary polynomials, but can be justified precisely in certain limiting cases where the boundaries of both pseudozero sets and pseudospectra reduce to certain generalized lemniscates in the complex plane.

To test this prediction of stability we have compared the companion matrix algorithm, which is the method used by the Matlab ROOTS command, with the Jenkins-Traub code CPOLY and the Madsen-Reid code PA16. Our experiments with a wide variety of polynomials suggest that all three codes are reliable, but that on average, contrary to what one might have expected, it is ROOTS that is the most accurate.

The figures on the next page show two examples of the approximate agreement between pseudozero sets and pseudospectra. The first example is a degree-21 Euler polynomial whose zeros are modestly ill-conditioned ($\kappa = O(10^5)$). The second is a degree-20 Wilkinson polynomial whose zeros $1, 2, 3, \ldots, 20$ are highly ill-conditioned ($\kappa = O(10^{17})$).

The significance of pseudozero sets and pseudospectra is not just a matter of rounding errors and stability. In any mathematical problem that apparently depends on polynomial zeros, it is likely that what really matters is whether $|p(z)|$ is very small, not necessarily exactly zero. Similarly, in a matrix eigenvalue problem what really matters may be whether $\|(zI-A)^{-1}\|$ is very large, not necessarily exactly infinity. Thus the study of pseudozero sets and pseudospectra is a natural one in its own right, having a bearing on the meaning of the zerofinding and eigenvalue problems themselves, not just on associated numerical algorithms.

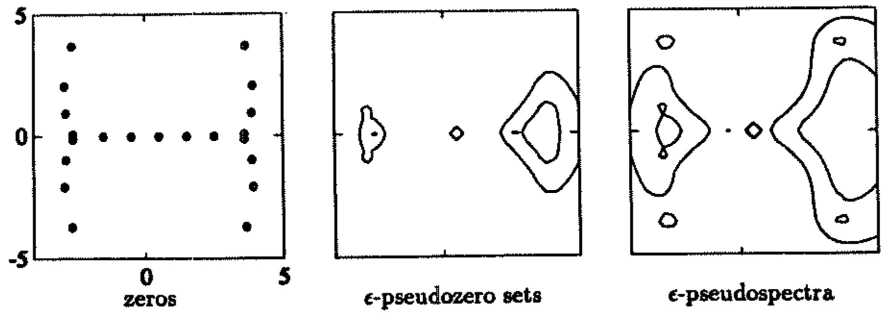


Figure 1. Degree-21 Euler polynomial ($\epsilon = 10^{-3}, 10^{-4}$).

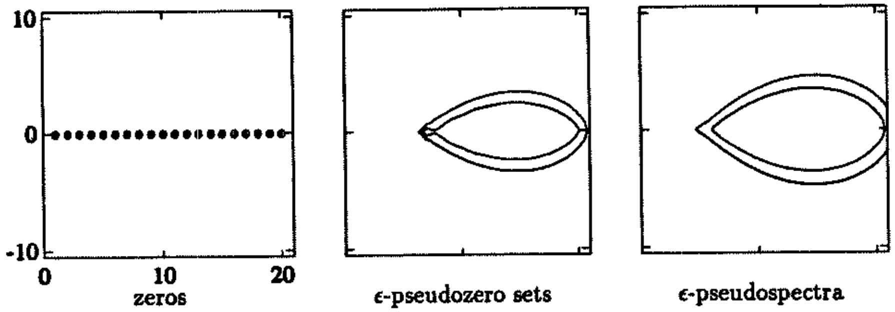


Figure 2. Degree-20 Wilkinson polynomial ($\epsilon = 10^{-12}, 10^{-13}$).

EIGENVALUE AND SINGULAR VALUE DISTRIBUTIONS FOR STRUCTURED MATRICES WITH APPLICATIONS TO CIRCULANT PRECONDITIONING TECHNIQUES

E.E.Tyrtyshnikov
Institute of Numerical Mathematics
Russian Academy of Sciences
Leninskij Prosp. 32-A, Moscow 117334, Russia
E-mail: tee@adonis.ias.msk.su

A unifying approach is proposed to studying the distributions of eigenvalues and singular values of Toeplitz matrices associated with a Fourier series, and multilevel Toeplitz matrices associated with a multidimensional Fourier series. Obtained are the extensions of the Szego and Avram - Parter theorems, where the generating function is now required to belong to L-two and not necessarily to L-infinity. Analogous extensions are given for multilevel Toeplitz matrices. In particular, it is proved that if f(x1,...,xp) belongs to L-two, then the p-level (complex) Toeplitz matrices allied with f have their singular values to be distributed as ABS(f(x1,...,xp)). The distribution results about the Cesaro (optimal) circulants are granted even if f is from L-one. Also suggested are new theorems on clustering that have to do with the preconditioning of multilevel Toeplitz matrices by multilevel circulants.

The approach is spread over to the case when a matrix family is constructed from some Toeplitz matrix families via basic matrix algebra operations, such as addition, multiplication, and inversion. If function f is obtained from the generating functions (that correspond to the Toeplitz matrix families at hand) with the help of similar but functional operations, then the singular values of the new structured matrix family are distributed as ABS(f), provided that certain conditions on the initial generating functions are fulfilled. The eigenvalues of Hermitian components of the matrices from the new family are distributed as Re(f) while those of the scew-Hermitian parts do as i*Im(f), where i is the imaginary unit. Among other things, these results seem to give a simplest possible way to explain nice properties of the circulant preconditioning.

Circulant matrices are customarily used as preconditioners for Toeplitz matrices. Previously known results that describe the spectrum after preconditioning are based on the assumptions that the generating function for Toeplitz matrices belongs to the Wiener class, and is strictly positive. Both assumptions are now weakened. Also proposed and studied the improved circulants. It is shown that (improved) circulants of Strang's type can be much more advantageous than optimal preconditioners. This crucially depends on the smoothness properties of f.

An interior-point method for convex optimization and its application to control problems

Lieven Vandenberghe[*] and Stephen Boyd[†]

Information Systems Laboratory, Electrical Engineering Department,
Stanford University, Stanford CA 94305

1 Motivation

Many problems in systems and control theory can be formulated (or reformulated) as optimization problems involving *linear matrix inequalities*, *i.e.*, constraints requiring an affine combination of symmetric matrices to be positive semidefinite. Reference [1] gives a broad survey of such problems.

These matrix inequalities are usually highly structured. One typical example is the (convex) *Lyapunov inequality* which has the form

$$APB + B^TPA^T + D \geq 0,$$

where the square matrices A, B and D are given, D is symmetric, and the symmetric matrix P is the optimization variable. Another important example is the (convex) *algebraic Riccati inequality:*

$$A^TP + PA + PBR^{-1}B^TP + Q \leq 0,$$

where A, B, Q and R are given, Q is symmetric, R is positive definite, and the matrix P is the optimization variable. Lyapunov and Riccati inequalities arise, for example, in stability analysis of dynamical systems. A typical optimization problem will have several matrix inequalities as constraints.

2 Interior-point methods

We describe an interior-point method for convex optimization problems involving matrix inequalities [5]. The method is based on the theory developed by Nesterov and Nemirovsky [4]. Alternatively, it can be interpreted as a generalization of Gonzaga and Todd's method for linear programming [3]. The algorithm has several important properties:

- It is based on the reduction of a potential function associated with the problem. This makes it possible to use large steps in each iteration.
- The method is primal-dual, which means that it processes a given problem and its dual simultaneously.
- A worst-case analysis shows that the number of iterations grows as the square root of the problem size. This bound is among the lowest achieved by interior-point methods.
- In practice the number of iterations appears to grow more slowly. As in other interior-point methods the overall computational effort is therefore dominated by the least-squares systems that must be solved in each iteration.

3 Numerical aspects

In each step a primal and a dual feasible direction are computed by solving a weighted least-squares problem. A type of conjugate-gradient algorithm can be used for this purpose, which results in important savings for two reasons. First, it allows us to take advantage of the special structure the problems often have (*e.g.*, Lyapunov or algebraic Riccati inequalities). Second, we show that the polynomial bound on the number of iterations remains valid even if the conjugate-gradient algorithm is not run until completion, which in practice can greatly reduce the computational effort per iteration.

We describe in detail how the algorithm works for optimization problems with L Lyapunov inequalities, each of size m. We prove an overall *worst-case* operation count of $O(m^{5.5}L^{1.5})$. The *average case* complexity appears to increase much more slowly with m, as $O(m^{\beta}L^{\gamma})$, with $\beta \approx 4$ and $\gamma \approx 1.5$. To appreciate these numbers, consider the following. A single Lyapunov equation $APB + B^TPA^T + D = 0$ (which is just a set of $m(m+1)/2$ linear equations for the $m(m+1)/2$ variables in P) can be solved in $O(m^3)$ operations by exploiting the special structure of the equations (see, *e.g.*, [2]). Therefore, it takes $O(m^3L)$ operations to solve L independent Lyapunov equations. Comparing this operation count to $O(m^4L^{1.5})$, we see that the relative cost of solving L coupled Lyapunov inequalities, compared to solving L independent Lyapunov equations, is only a factor of $mL^{0.5}$. A similar statement holds for Riccati inequalities.

4 Conclusion

The algorithm that we describe involves two important extensions beyond the methods described by Nesterov and Nemirovsky. First, it takes advantage of the special structure of

[*] Senior research assistant of the Belgian National Fund for Scientific Research (NFWO); on leave from Katholieke Universiteit Leuven, Belgium; Internet: vandenbe@isl.stanford.edu

[†] Research supported in part by AFOSR grant F49620-92-J-0013-P00001 and DARPA grant AF F49620-90-C-0014; Internet: boyd@isl.stanford.edu

the matrix inequalities we encounter, *e.g.*, Lyapunov or Riccati. Second, it allows the use of approximate search directions,

The most significant conclusion is the following. Much of modern control theory involves the solution of Riccati and Lyapunov equations. Our results show that the computational cost of extending current control theory to a theory based on the solution of (multiple, coupled) Lyapunov or Riccati inequalities is modest.

References

[1] S. Boyd, L. El Ghaoui, E. Feron and V. Balakrishnan. *Linear matrix inequalities in systems and control theory*, 1993. Monograph in preparation.

[2] G. Golub, S. Nash and C. Van Loan, "A Hessenberg-Schur method for the matrix problem $AX + XB = C$", *IEEE Transactions on Automatic Control*, AC-24:909-913,1979.

[3] C. C. Gonzaga and M. J. Todd, "An $O(\sqrt{n}L)$-iteration large-step primal-dual affine algorithm for linear programming, *SIAM Journal on Optimization*, 2(3):349-359, August 1992.

[4] Yu. Nesterov and A. Nemirovsky. *Interior point polynomial methods in convex programming: theory and applications*, SIAM, 1993.

[5] L. Vandenberghe and S. Boyd, "Primal-dual potential reduction method for problems involving matrix inequalities", *submitted to Mathematical Programming*, 1993.

Further improvements in nonsymmetric hybrid CG methods

Henk van der Vorst – Utrecht University

In the past few years new methods have been proposed that can be seen as combinations of standard Krylov subspave methods, such as Bi–CG and GMRES. One of the first hybrid schemes of this type is CGS, actually the Bi–CG squared method. Other such hybrid schemes include BiCGSTAB (a combination of Bi–CG and GMRES(1)), QMRS, TFQMR, Hybrid GMRES (polynomial preconditioned GMRES) and the nested GMRESR method (GMRES preconditioned by itself or other schemes). These methods have been successful in solving relevant sparse nonsymmetric linear systems, but there is still a need for further improvements.

Bi–CG has two break–down conditions, one of which can be removed by a look–ahead strategy, the other can be reomoved by the QMR approach. A weak point in BiCGSTAB is that it introduces one more break–down possibility on top of these, namely when the GMRES(1) part of the algorithm stagnates. This may happen, for instance in advection dominated pde–problems. Gutknecht has suggested to combine Bi–CG with GMRES(2): BiCGSTAB2 to overcome this problem. We will show that if this idea is implemented in a different manner, a very competitive BiCGSTAB variant is obtained, which is easy to implement. Moreover, it turns out to be relatively easy to obtain other obvious variants, like Bi-CG with GMRES(4), etc., which may be attractive if one has memory space available.

For GMRESR we will propose a further improvement obtained by preventing the preconditioning iteration scheme to construct search directions in previously explored subspaces, that is by maintaining orthogonality in the outer iteration as well as the inner iteration.

If time permits, we will discuss approaches to increase parallelism and to avoid synchronization points in Krylov subspace methods. While for preconditioned CG it seems possible to overlap all communication with computation, without giving up any stability in the method, this turns out to be much more difficult in GMRES. Our approaches will be illustrated by results obtained on distributed memory machines.

End of abstract for Householder Symposium

Relationships between Structured TLS and Constrained TLS with applications to Signal Enhancement

Sabine Van Huffel*
ESAT Laboratory, Department of Electrical Engineering,
Katholieke Universiteit Leuven
Kardinaal Mercierlaan 94, 3001 HEVERLEE, BELGIUM.

tel: 32/16/22 09 31 fax.: 32/16/22 18 55
e-mail: vanhuffel@esat.kuleuven.ac.be

*Research Associate of the Belgian N.F.W.O. (National Fund for Scientific Research)

Abstract

The Total Least Squares (TLS) method has been devised as a more global fitting method than Least Squares (LS) for solving overdetermined sets of linear equations $Ax \approx b$ in which A, as well as b, are noisy [4]. From a statistical point of view, TLS operates under the assumption that the errors in A and b are independently and identically distributed with zero mean and equal variance. If there is correlation among the errors, a noise whitening transformation can be applied or a Generalized TLS problem [4] can be solved and the error norm is appropriately modified. However if there is a linear dependence among the error entries in $[\Delta A; \Delta b]$ -which is the case when the data matrix is linearly structured: Hankel, Toeplitz,...- then the TLS solution may no longer yield optimal statistical estimators. This happens for instance in system identification when we try to estimate the impulse response of a system from its input and output by discrete deconvolution. The errors in the corresponding data matrix are obviously Toeplitz and this information is not used in the classical TLS problem. Another important example where the errors in the data matrix have a block Toeplitz structure and are hence linearly dependent is forward-backward linear prediction, used to estimate the frequencies of sinusoids from measurements contaminated by white noise. Other applications occur in frequency estimation, estimation of the angular location of emitters and superresolution harmonic analysis [1].

To get more accurate estimates of x, Abatzoglou and Mendel extended the classical TLS method to incorporate the algebraic dependence of the errors in $[A; b]$ and called their extension "constrained TLS".

Definition 1 Constrained TLS (CTLS) problem.

Let $$C = [A_{m\times n}; b_{m\times 1}], \Delta C = [\Delta A; \Delta b] = [\Delta c_1, \ldots, \Delta c_{n+1}]$$

and express each error column Δc_j as $\Delta c_j = F_j v$ where F_j is a matrix of an appropriate size and v is a zero-mean white noise vector of minimal dimensionality. Then, the CTLS solution $\hat{x}$ is obtained from the following constrained minimization problem involving v and x :

$$\underset{v,\, x}{\text{minimize}} \; \|v\|_2^2 \; \text{subject to} \; \{C + [F_1 v; \ldots; F_{n+1} v]\} \begin{bmatrix} x \\ -1 \end{bmatrix} = 0 \qquad (1)$$

This is a quadratic minimization problem which is subject to a quadratic constraint equation. $\hat{x}$ can be obtained as the solution of an unconstrained minimization problem. Suboptimal algorithms have been applied to minimize the above functional and rather successful results have been obtained so far. Also, a complex version of the Newton method for finding the minimum of a real function of several complex variables has been derived and applied to find the CTLS solution (see [1] for references).

In this paper, we show how CTLS is a special case of the Structured TLS (STLS) problem recently presented by B. De Moor [3] and formalized as follows.

Definition 2 Structured TLS (STLS) problem.
Let $C(r) = C_0 + r_1 C_1 + \ldots + r_n C_n$ be an affine matrix function of the parameter vector r where $C_i, i = 0, 1, \ldots, n$ are fixed given $m \times q$ matrices. Let a be an $p \times 1$ data vector and w

be a given vector of weights. Find a rank-deficient matrix in the affine set $C(r)$ such that a given quadratic function $[r, a, w]_2^2$ of the parameters r_i is minimized, i.e.

$$\min_{r \in \mathcal{R}^n} [r, a, w]_2^2 \quad \textit{subject to} \begin{cases} C(r)y = 0 \\ y^T y = 1 \end{cases}$$

The solution to the STLS problem follows from a nonlinear generalized SVD problem. A straightforward linear convergent algorithm is derived that is based on the inverse iteration method to find the smallest singular value and corresponding singular vectors of a matrix.

Although both methods use a different formulation and solve the problem in a quite different way, there are some nice similarities between both approaches which are described in the paper. Based on these similarities we could simplify the straightforward inverse iteration based algorithm outlined in [3] for solving the CTLS and STLS problem in cases where the data matrix $C = [A; b]$ is a linearly structured unweighted matrix. Numerical examples are given in which one wishes to approximate a Hankel matrix by one of lower rank. This problem is a key issue in the (partial) realization problem and in the enhancement of sinusoidal and exponentially modeled signals. Applications occur in system identification, modal analysis, biomedical signal processing such as Nuclear Magnetic Resonance spectroscopy [5], etc. The STLS and CTLS approaches are compared to currently used suboptimal approaches described in [2, 5]. The latter methods first reduce the rank of the Hankel data matrix by computing the truncated SVD approximation of the data matrix [2] or the minimum variance estimate of the signal-only matrix [5] and then restore the Hankel structure by finding the closest Hankel matrix (in Frobenius norm) which is simply obtained by replacing the antidiagonals by the average of their elements. However, the new Hankel matrix is no longer rank-deficient. One then iterates by again computing the truncated SVD or minimum variance estimate and restoring the Hankel structure, etc. This process converges but the solution of these suboptimal approaches does not satisfy any H_2 optimality condition.

References

[1] T.J. Abatzoglou, J.M. Mendel and G.A. Harada, *The constrained total least squares technique and its application to harmonic superresolution.* IEEE Trans. Signal Processing, Vol. SP-39, No.5, May 1991, pp.1070-1086.

[2] J. Cadzow, *Signal enhancement: a composite property mapping algorithm*, IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. ASSP-36, No. 1, January 1988, pp. 49-62.

[3] B. De Moor, *Structured total least squares and L_2 approximation problems*, ESAT-SISTA Report 1992-33, ESAT Laboratory, K.U.Leuven, Belgium, 1992 (submitted for publication).

[4] S. Van Huffel and J. Vandewalle, *The total least squares problem : computational aspects and analysis*, Frontiers in Applied Mathematics series, Vol.9, SIAM, Philadelphia, 1991.

[5] S. Van Huffel, *Enhanced Resolution Based on Minimum Variance Estimation and Exponential Data Modeling.* ESAT-SISTA Report 1992-22, ESAT Laboratory, K.U.Leuven, Belgium, April, 1992 (submitted to Signal Processing).

Stable numerical algorithms for equilibrium systems (extended abstract)

Stephen A. Vavasis*

September 28, 1992

NOTE: This extended abstract is not for distribution. If the reader would like a full version that may be circulated, please send me email (vavasis@cs.cornell.edu) and I will be happy to send a longer tech-report version.

An equilibrium system (also known as a KKT system, a saddle-point system, or a sparse tableau) is a square linear system with the following structure:

$$\begin{pmatrix} D & -A \\ A^T & 0 \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} b \\ c \end{pmatrix} \qquad (1)$$

G. Strang [1986] has observed that equilibrium systems arise in optimization, finite elements, structural analysis, and electrical networks.

For example, in optimization, D generally encodes the second derivative of the objective function and A encodes the constraints.

Recently, G. W. Stewart [1989] established a norm bound for a type of equilibrium system in the case that D is very ill-conditioned. In this paper we investigate the algorithmic implications of Stewart's result. We show that all standard textbook algorithms for equilibrium systems are unstable. Then we show that a certain hybrid method has the right stability property.

In all of the applications, the following two assumptions are commonplace, and they are made throughout the paper. First, matrix D is symmetric and positive definite. Second, matrix A has full column rank. These assumptions imply that (1) is a nonsingular linear system with a unique solution.

The main focus of this paper is what happens when D is severely ill-conditioned. In the case that D is well-conditioned, the numerical problems associated with solving (1) are generally not as troublesome, and most standard methods will give good answers.

The most natural framework for this assumption is an optimization algorithm involving a barrier function. The primary example of a barrier function in optimization is the class of interior point methods for linear programming. In an interior point method, matrix D becomes very ill-conditioned when the iterate approaches the boundary of the feasible region. (See Wright [1992] for a description of barrier methods, linear programming, and their relationship.) For linear programming, since the solution is always on the boundary of the region, ill-conditioning in D *always* occurs during the algorithm.

*Department of Computer Science, Upson Hall, Cornell University, Ithaca, NY 14853.

In order to carry out the analysis, we make the following further assumptions: first, we are more interested in recovering y in (1) rather than x. Second, $c = 0$. Third, D is a diagonal matrix. There are a number of applications where these assumptions are reasonable. In the full paper we discuss conditions under which these restrictions could be lifted.

Under these assumptions, we can apply Stewart's theorem. Simplifying (1), we have the following equation for y:

$$y = -(A^T D^{-1} A)^{-1} A^T D^{-1} b. \qquad (2)$$

Stewart's theorem states that the matrix $(A^T D^{-1} A)^{-1} A^T D^{-1}$ has a uniform norm bound depending only on A (not D).

Further analysis in the full paper shows that Stewart's theorem implies that y can be recovered accurately even if D is arbitrarily ill-conditioned. However, the standard algorithms for (1), including symmetric indefinite factorization, the range-space method, and the nullspace method, can give answers without any digits of accuracy even for simple three-node examples from electrical engineering. (By "standard algorithm" we mean that these algorithms are described in optimization textbooks such as Fletcher [1987], electrical engineering textbooks such as Chua, Desoer, and Kuh [1987], or civil engineering textbooks such as Timoshenko and Young [1965].)

We propose a new algorithm, called the NSH method (nullspace scaled hybrid method). The NSH method finds a certain nullspace basis V for the matrix $A^T D^{-1}$ in a careful way, and then solves the linear system

$$[A, V]\begin{pmatrix} y \\ q \end{pmatrix} = -b. \qquad (3)$$

The NSH method does not appear in standard textbooks, but similar approaches have appeared in the literature. For example, Coleman and Li [1989] suggest a similar approach (called the "full-space" method) for optimization, but with the scaling done in a different manner.

We prove that this method solves the problem stably, and verify that result with some computational tests.

References

L. O. Chua, C. A. Desoer, and E. S. Kuh [1987], *Linear and Nonlinear Circuits*, McGraw–Hill, New York.

T. F. Coleman and Y. Li [1989], A globally and quadratically convergent affine scaling method for linear l_1 problems, Technical Report 89–1026, Department of Computer Science, Cornell University, Ithaca, NY. Also, *Math. Progr.*, to appear.

R. Fletcher [1987], *Practical Methods of Optimization, 2nd Edition*, J. Wiley and Sons, Chichester.

G. W. Stewart [1989], On scaled projections and pseudoinverses, *Linear Algebra and its Applications* 112:189–193.

G. Strang [1986], *Introduction to Applied Mathematics*, Wellesley-Cambridge Press, Wellesley, MA.

S. P. Timoshenko and D. H. Young [1965], *Theory of Structures, 2nd Edition*, McGraw–Hill, New York.

M. H. Wright [1992], Interior methods for constrained optimization, in *Acta Numerica 1992*, Cambridge University Press, Cambridge.

The convergence of preconditioned Conjugate Residual iterations for the Stokes problem

Andy Wathen, University of Bristol
Bernd Fischer, University of Hamburg
David Silvester, UMIST

Discretisation of the incompressible Navier-Stokes equations gives rise to systems of equations of the form

$$\begin{pmatrix} A & B^t \\ B & O \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ O \end{pmatrix}.$$

For a number of important discretisation techniques for the Navier-Stokes equations and in particular for the classical Stokes problem for slow flow, the system is linear and the square submatrix A is symmetric and positive definite, thus rendering the complete system symmetric but indefinite. Since, for example for finite element approximation, these systems will be large and sparse for practical flow problems, solution using iterative methods is indicated.

Most commonly, iterative techniques are applied in a so-called 'pressure-correction' approach which in algebraic form is equivalent to a block elimination yielding the positive definite Schur complement system

$$BA^{-1}B^t p = BA^{-1}f$$

which can be effectively solved by the preconditioned (Hestenes-Steifel) Conjugate Gradient method. A further inner iteration can then be used to solve the systems indicated by the A^{-1} at each outer iteration.

In this talk we will describe and analyse an alternative iterative approach for Stokes-like problems using (non-nested) Preconditioned Conjugate Residual (MINRES) methods.

To set the stage for our analysis, we will briefly review the convergence theory which, as for other Krylov-subspace methods for normal matrices, is expressed in terms of polynomial approximation problems on the eigenvalue spectrum, viz

$$\frac{\|x_k - x\|}{\|x_0 - x\|} \le e_k \equiv \min_{\{p_k : p_k(0)=1\}} \max_{\lambda \in D} |p_k(\lambda)|$$

for some appropriate norm where x_k are the solution iterates, x the desired solution and D is an inclusion set for the eigenvalues.

We will then present eigenvalue estimates for preconditioned Stokes systems with preconditioners of the form

$$\begin{pmatrix} M_A & O \\ O & M_p \end{pmatrix}$$

where M_A is a preconditioner for the positive definite (discrete Laplacian) submatrix A, and M_p is an appropriate simple scaling matrix which we will identify. We will show that if the spectral condition number of the preconditioned positive definite matrix $M_A^{-1}A$ is κ, then the eigenvalues of the preconditioned Stokes system lie in

$$[-a, -b] \cup [c, d] \tag{1}$$

where a, b, c and d are positive values satisfying

$$a/b \le \sqrt{\kappa} \quad \text{and} \quad d/c \le \kappa. \tag{2}$$

Thus the negative eigenvalues are, in a precise way, more clustered than the positive eigenvalues.

The analysis applies to 'optimal' preconditioners M_A such as some based on domain decomposition as well as to other widely used preconditioners such as incomplete cholesky factorisation and its modifications and also to the simple case of scaling: $M_A = \text{diag}(A)$. For non-optimal preconditioners, the eigenvalue bounds are naturally expressed in terms of the asymptotically small mesh-size parameter, h.

Finally, we will present new results on the convergence rate of MINRES for symmetric indefinite matrices with eigenvalue spectra which are not symmetric about the origin, but for which the eigenvalues are contained in intervals of the form (1), (2).

These results show that MINRES for such problems will converge at a rate exactly intermediate between the convergence rate of Hestenes-Steifel preconditioned Conjugate Gradients applied to the positive definite system $M_A^{-1}A$ and Conjugate Gradients applyied to the normal equations

$$\begin{pmatrix} A & B^t \\ B & O \end{pmatrix} \begin{pmatrix} A & B^t \\ B & O \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} A & B^t \\ B & O \end{pmatrix} \begin{pmatrix} f \\ O \end{pmatrix}.$$

It is known that the rate of convergence of MINRES for a symmetric indefinite problem when the eigenvalues are contained in intervals which are symmetric about the origin is essentially the same as for Conjugate Gradients applied to the normal equations. (It is exactly the same if the individual eigenvalues are symmetric.) Thus the Stokes problem is of distinctly different character than the 'worst case' situation of symmetrically placed eigenvalues, and MINRES is an attractive solution method for such problems.

Precisely, we will show that if the eigenvalues of an indefinite matrix are contained in intervals of the form

$$[-a, -b\sqrt{\alpha}] \cup [c\alpha, d]$$

where α is an asymptotically small quantity (such as a simple montonic function of the mesh-size, h in a partial differential equation problem) and a, b, c and d are positive constants independent of α, then the asymptotic convergence rate of MINRES iteration is

$$\lim_{k\to\infty} |e_k|^{\frac{1}{k}} = 1 - O(\alpha^{\frac{3}{4}}).$$

For Conjugate Gradients applied to a positive definite problem with eigenvalues $\lambda \in [c\alpha, d]$ the standard result is

$$\lim_{k\to\infty} |e_k|^{\frac{1}{k}} = 1 - O(\alpha^{\frac{1}{2}}),$$

and for the normal equations it is correspondingly

$$\lim_{k\to\infty} |e_k|^{\frac{1}{k}} = 1 - O(\alpha).$$

The results of numerical computations on Stokes flow problems which we shall present show the accuracy of this theory in practice.

(Theoretical and practical comparison of the MINRES and Conjugate Gradient pressure correction approach for Navier-Stokes flow simulations using implimentations in the commercial FEAT industrial fluid flow simulation code are the subject of a second submitted talk to this meeting by Alison Ramage and Andy Wathen.)

Forward (In)Stability of the QR Algorithm

David S. Watkins
Washington State University

Consider a matrix A that is upper Hessenberg and has a zero on the subdiagonal. Say

$$A = \left[\begin{array}{cccc|ccc} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{17} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} & a_{27} \\ & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} & a_{37} \\ & & a_{43} & a_{44} & a_{45} & a_{46} & a_{47} \\ \hline & & & 0 & a_{55} & a_{56} & a_{57} \\ & & & & a_{65} & a_{66} & a_{67} \\ & & & & & a_{76} & a_{77} \end{array}\right].$$

If we perform a step of the QR algorithm on this matrix (without splitting the problem into two subproblems), the outcome will depend on whether we are using an explicit or an implicit implementation. The explicit implementation will effectively perform QR steps on the (1,1) and (2,2) blocks independently. On the other hand, the implicit (bulge-chasing) implementation will perform a QR step on the (1,1) block and leave the (2,2) block unchanged. The step "dies" when it hits the zero.

Now suppose we change the problem slightly and consider

$$A = \left[\begin{array}{cccc|ccc} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} & a_{16} & a_{17} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} & a_{27} \\ & a_{32} & a_{33} & a_{34} & a_{35} & a_{36} & a_{37} \\ & & a_{43} & a_{44} & a_{45} & a_{46} & a_{47} \\ \hline & & & \epsilon & a_{55} & a_{56} & a_{57} \\ & & & & a_{65} & a_{66} & a_{67} \\ & & & & & a_{76} & a_{77} \end{array}\right],$$

where ϵ is small, say 10^{-20}. (We assume that the other nonzero entries of A are of order 1.) What happens if we perform an implicit QR step (with roundoff errors) on this matrix? The conventional wisdom is that the effect of the near-zero ϵ will be similar to that of an exact zero. In this case (according to conventional wisdom) the (2,2) block will not be left unchanged, but it will be altered in a somewhat random way; the small ϵ triggers a loss of precision (forward instability) that "washes out" the step. If QR steps are repeated under these conditions, the convergence rate will be degraded severely, or so says the conventional wisdom.

We will demonstrate, by examples and by analysis, that the conventional wisdom is wrong. Assuming the matrix is well balanced, a small entry (or even several consecutive small entries) on the subdiagonal will not normally cause forward instability or degrade the convergence in any way. We will determine the conditions under which forward instability does occur and compare our results with those obtained by Parlett and Le [1] for the symmetric case.

Our findings have implications for pipelined, parallel implementations of the QR algorithm, for which it might not always be convenient to check for possible deflations after each iteration. Our results demonstrate that, contrary to popular belief, prompt deflations are not crucial to the success of the algorithm.

References

[1] B. N. Parlett and J. Le, *On the forward instability of the tridiagonal QR transformation,* SIAM J. Matrix Anal. Appl., 14 (1993), to appear.

Title: Circulant, Skew-circulant and Toeplitz Preconditioners
for Elliptic Problems

Speaker: Chiu-kwong Wong
Department of Mathematics,
University of Hong Kong, Hong Kong

Joint-work with: Raymond H. Chan
Department of Mathematics,
Hong Kong University of Science and Technology
Clear Water Bay, Hong Kong

Abstract

Linear systems $Ax = b$ arising from discretizations of second-order elliptic equations are solved via preconditioned conjugate gradient methods. Recently, R. Chan and T. Chan proposed using block circulant preconditioners for the solution of these problems. The preconditioners approximate the elliptic operator in all coordinate directions and can be chosen so that the condition number of the preconditioned system can be reduced from $O(n^2)$ to $O(n)$. In this talk, we extend the idea and propose circulant preconditioner C and skew-circulant preconditioner S which approximate the elliptic operator in all but one coordinate directions. We prove that the condition number of the preconditioned systems are also of $O(n)$. We then show that the preconditioners can be derived by using ideas similar to the INV algorithm proposed by Concus, Golub and Meurant. Toeplitz preconditioners based on C and S will be also discussed.

SOLVING EQUATIONS EXACTLY ON DISTRIBUTED MEMORY MULTIPROCESSORS

Deng Jian Xin
MIMD Systems, Inc., Belmont, CA 94002 USA

Abstract

A parallel congruence algorithm for the exact solution of integer system of linear equations is presented. The computations were executed on a five T800 Transputer network, a distributed memory multiprocessor. Our experiment shows that the algorithm is a practicable method and an excellent candidate for exact computation, high precision computations, and ill-condition problems.

1. Introduction

Let A be an integral nxn matrix, b and integral nx1 vector and d=det(A)≠0. A^{adj} is an adjoint matrix of A. Solving a given system of linear equations

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

over an integral domain is equivalent to solving following equation

$$Ay = db \tag{2}$$

where

$$y = A^{adj} b \tag{3}$$

From the definitions of adjoint matrix, A^{adj} is an integral nxn matrix, so that y is an integral nx1 vector. If the usual method is applied exactly to the solution of equation (2), then the numbers involved increase very rapidly. Congruence arithmetic is noted for avoiding this difficulty. Generally, system (2) can be solved exactly by congruence arithmetic with single precision arithmetic in the main process of computations [2]. The infinite precision integral system has to be used only in the first and the last stage. Moreover, the Chinese congruence theorem provides natural parallelism of congruence techniques. In this paper, a parallel algorithm is presented for solving systems of equations exactly on parallel computers. The numerical experiment shows that the natural parallelism of the congruence techniques makes the algorithm an excellent method for execution on distributed memory multiprocessor.

This research was supported by Chinese National Science Foundation and MIMD Systems,Inc.,Belmont,CA 94002 USA

N	det(H)									
10	10673	16531	9							
20	82083	55717	043							
30	11487	70530	78810	13						
40	38151	96815	22430	31360						
50	17577	48265	37839	63187	4314					
60	11266	29787	64640	76325	86174	37				
70	12629	53975	94210	89027	83133	642				
80	12518	38063	31869	02206	91121	20934	41961	3		
90	92238	45326	45356	25398	51915	01222	97303	938		
100	24027	88061	31639	57659	74293	88622	61505	20243	14535	3

Table 2

N	10	20	30	40	50	60	70	80	90	100
t_s	53101	91675	161612	212495	455517	693960	1008741	14173883	1893379	2574219
t_p	20430	24563	41836	68547	101724	151682	217966	301880	405520	534053
s_p	2.6	2.7	2.9	3.1	4.4	4.5	4.6	4.7	4.7	4.8

N--order of matrix, t_s--sequential time, t_p--parallel time (50- micro-seconds)

s_p--speed up

Table 3

6. Conclusion

Our research work shows that the congruence parallel algorithm for solving system of linear equations exactly on distributed memory multiprocessors is practicable. For the problem whose order of matrix is less then 100, the overhead is about 30 seconds in a five T800 network by our experimental program. The very high efficiency and speed up of the congruence parallel techniques makes the algorithm an excellent candidate for the exact computations, high precision computations and ill-condition problems.

References

1. Xu Xian yu, Li Jia kai, Xu Guo lian, Li Shu xin, The infinite precision arithmetic MP-system, Computing Center, Chinese Academy of Sciences, 1978.

A Breakdown-Free Variation of the Nonsymmetric Lanczos Algorithms

Qiang Ye
Department of Applied Mathematics
University of Manitoba
Winnipeg, Manitoba
Canada R3T 2N2

January 7, 1993

The nonsymmetric Lanczos tridiagonalization algorithm is essentially the Gram-Schmidt biorthogonalization method for generating biorthogonal bases of a pair of Krylov subspaces. It suffers from breakdown and instability when a pivot at some step is zero or nearly zero, which is often the result of mismatch of the two Krylov subspaces. There are serveral methods available in the literature to overcome this difficulty, most of which are based on constructing transformed biorthogonal bases of the same Krylov subspaces. We propose instead to modify one of the two Krylov subspaces by introducing a new-start vector when a pivot is small. The new-start vector generates another Krylov subspace, which we add to the old one in an appropriate way so that the Gram-Schmidt method for the modified subspaces yields a recurrence similar to the Lanczos algorithm. Then a banded Hessenberg matrix is obtained and used to approximate the original matrix. Our method enforces the pivots to be above a given tolerance and can handle the situations of both exact breakdown and near-breakdown. In particular, we recover the look-ahead Lanczos algorithm and the Arnoldi algorithm as two special cases.

Our method is also new and of significance in the symmetric case. For example, by applying the new method to the symmetric block Lanczos algorithm, we obtain an algorithm that allows to increase the block size during the iteration and thus eliminate the difficulty of choosing the block size.

Some theoretical analyses of the new method and numerical examples will be presented.

ITERATIVE SOLUTION STRATEGIS FOR LARGE DENSE LINEAR SYSTEMS COMING FROM 3D CFD AND ELECTROMAGNETICS PROBLEMS

A.Yeremin
Institute of Numerical Mathematics
of the Russian Academy of Sciences
E-mail: badger@adonis.ias.msk.su

Large dense real and complex linear systems frequently arise in industrial applications. Direct solution strategies seems to be very unattractive in this case by the following reasons:

(1) The arithmetic complexity grows like the third degree of the problem size, thus the solution of a problem of size 100 000 will take thousands of hours of the CPU time even on biggest modern computers.

(2) The direct solution of even medium sized problems (like 20 000 - 30 000) requires enormous amount of the I/O activities which can lead to a dramatics increse of the wall clock time despite of the full overlapping of arithmetics and communications.

(3) To maintain the numerical stability the direct solution methods may require pivoting strategies (especially when solving large dense linear systems). They are able to 'kill' any out of core solver.

The talk describes recent results related to construction of efficient parallel iterative methods for solving large dense linear systems. The suggested approach is based on exploitation of Block Diagonally Perturbed - Block Incomplete Triangular Factorizations accelerated by the Block Eigenvalue Translation based Block GMRES(k) method [1]. Ensuring the numerical stability of Block Incomplete Triangular Factorization by using Block Diagonal Perturbations we are able to construct high quality preconditioners contaning a relatively small number of nonzero entries. To process multiple right hand sides we exploit the Block GMRES(k) method where we perform one block iteration for several right hand sides.

This iterative solution strategy enables us to reduce substantially the arithmetic costs as compared with the direct methods (even when processing hundreeds of multiple right hand sides) and to decrease dramatically the required amount of the I/O acitivities. It should be also emphasized that our approach can keep all processors busy when communications and arithmetics can overlap.

The results of numerical experiments with dense linear systems of sizes about tens of thousands are presented. Numerical experiments are performed on CRAY/YMP/M90 computers.

REFERENCES

1. S.A. Kharchenko and A.Yu.Yeremin, Eigenvalue Translation Based Preconditioners for the GMRES(k) Method. Research Report RR-EM 2/92, Elegant Mathematics, Inc.(USA), 1992.

ABSTRACT

Lanczos Type Methods for the Solution of Nonsymmetric Linear Systems

by

David M. Young and Jen Yuan Chen
The University of Texas

In this paper we consider some Lanczos type methods for solving large systems of linear algebraic equations with sparse, nonsymmetric matrices. Such systems typically arise in the numerical solution of non-self-adjoint elliptic partial differential equations by finite difference methods or finite element methods.

The Lanczos type methods which we consider can be derived from a class of generalized conjugate gradient methods (GCG methods). With GCG methods one chooses an auxiliary matrix Z and for n=1,2... determines an approximate solution, $u^{(n)}$, by requiring that $u^{(n)} - u^{(0)} \varepsilon K_n(r^{(0)}, A)$ and that $(Zr^{(n)}, \mathrm{v}) = 0$ for all v $\varepsilon K_n(r^{(0)}, A)$. Here $u^{(0)}$ is the initial approximation to the solution $\bar{u} = A^{-1}b$ of the given system, $Au = b$, and $r^{(n)} = b - Au^{(n)}$. The Krylov space $K_n(r^{(0)}, A)$ is spanned by the vectors $r^{(0)}, Ar^{(0)} \ldots, A^{n-1}r^{(0)}$, where $r^{(0)} = b - Au^{(0)}$. Young and Jea [1980] considered three procedures for the determination of the $u^{(n)}$; these were referred to as ORTHODIR, ORTHOMIN and ORTHORES. ORTHODIR is, in theory, the most robust since it converges whenever ORTHOMIN and ORTHORES converge. However, numerical experiments, e.g. Abbassian [1983], indicate that ORTHODIR often suffers from numerical instability. Saad and Shultz [1986] developed a procedure, called GMRES, which is mathematically equivalent to ORTHODIR but which is more stable and requires fewer operators per iteration.

The amount of work required per iteration with the GCG methods usually increases linearly as the number of iteration increases. Jea and Young [1983] considered the application of the GCG method to the double system $\{A\}\{u\} = \{b\}$ where

(1) $$\{A\} = \begin{pmatrix} A & 0 \\ 0 & A^T \end{pmatrix}, \{u\} = \begin{pmatrix} u \\ \tilde{u} \end{pmatrix}, \{b\} = \begin{pmatrix} b \\ \tilde{b} \end{pmatrix}$$

By choosing the auxiliary matrix

(2) $$\{Z\} = \{E\} = \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix}$$

they obtained three Lanczos type methods which they referred to as LANDIR, LANMIN and LANRES. LANMIN is equivalent to the "biconjugate gradient method" (BCG method) considered by Fletcher [1976]. For each method the work per iteration does not increase as the number of iterations increases; however, the methods may break down and, even when they do not, they often exhibit erratic convergence or fail to converge.

A number of papers, for example the paper by Freund and Nachtigal [1990] on QMR and the paper by Van der Vorst [1992] on Bi-CGSTAB, have appeared recently which describe modifications of the BCG method which are designed to improve its convergence behavior. Other papers, for example Joubert [1990], have appeared which describe methods for avoiding or coping with breakdown of the BCG method.

The main focus of this paper is on several Lanczos type methods other than LANMIN (or BCG). It can be shown that one such method, LANDIR, converges whenever LANMIN converges. However, as in the case of ORTHODIR, the behavior of LANDIR is often very erratic. We consider modifications of LANDIR, LANMIN and LANRES corresponding to the use of the GCG methods with the modified auxiliary matrix $\{\tilde{Z}\} = \{A^T\}\{E\}$ applied to the double system (1). Also using the modified $\{Z\}$ we develop a procedure, which we call "LANGMRES". This procedure which is similar to GMRES in that it involves first determining a set of vectors which are mutually orthogonal with respect to $\{E\}$ and then finding a least squares solution to a related linear system. For all of these methods the amount of work required per iteration does not increase as the number of iterations increases. We are now carrying out numerical studies on LANGMRES as well as on the modified versions of LANDIR, LANMIN and LANRES, to determine how well they perform in comparison to BCG, LANDIR and LANRES. We will also compare the procedure with a modified version of LANDIR, based on re-scaling, which was recently developed by Mai [1992].

REFERENCES

Abbassian, R.O. [1983]. "Lanczos Algorithms for the Acceleration of Nonsymmetrizable Iterative Methods" Report CNA-193, Center for Numerical Analysis, The University of Texas, Austin, Texas.

Fletcher, R. [1976]. "Conjugate Gradient Methods for Indefinite Systems", *Lecture Notes in Mathematics* **506**, Springer-Verlag, New York.

Freund, Roland W. and Nachtigal, Noel [1991]. "QMR: a Quasi-Minimal Residual Method for Non-Hermitian Linear Systems," Numerische Math., **60**, 315-339.

Jea, Kang C. and Young, David M. [1983]. "On the Simplification of Generalized Conjugate-Gradient Methods for Nonsymmetrizable Linear Systems", *Linear Algebra and Its Applications*, 52/53, 399-417.

Joubert, Wayne [1990]. "Iterative Methods for the Solution of Nonsymmetric Systems of Linear Equations", Report CNA-242, Center for Numerical Analysis, The University of Texas, Austin, Texas.

Mai, Tsun-zee [1992]. "Modified Lanczos Method for Solving Large Sparse Linear Systems", to appear in *Communication in Applied Numerical Methods.*

Saad, Y. and Schultz [1986]. "GMRES: a Generalized Minimum Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM J. Sci. Stat. Comput.* **7**, 856-869.

Van der Vorst, H.A. [1992]. "Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems" *SIAM J. Sci, Stat. Comp.* **13**.

Young, David M. and Jea, Kang C. [1980]. "Generalized Conjugate Gradient Acceleration of Non-symmetrizable Iterative Methods", *Linear Algebra and Its Applications* **34**, 159-194.

THE CANONICAL CORRELATIONS OF MATRIX PAIRS THEORY, ALGORITHMS AND EXTENSIONS

HONGYUAN ZHA*

The concept of canonical correlations was first introduced by Hotelling to tackle the problem of identifying and measuring relations between two sets of random variables. Canonical correlation analysis has a wide variety of applications in statistics, econometrics, psychology, educational research, anthropology, botany, geography and ecology. In the discrete sample case, where A and B are two matrices representing observations of two sets of random variables, the canonical correlations of the matrix pair (A, B) are defined as follows.

DEFINITION 0.1. *Let $A \in R^{m\times n}$ and $B \in R^{m\times l}$, and assume that*

$$p = \text{rank}(A) \geq \text{rank}(B) = q.$$

The canonical correlations $\sigma_1(A,B), \cdots, \sigma_q(A,B)$ of the matrix pair (A,B) are defined recursively by the following formulae, for $k = 1, \cdots, q$,

$$\sigma_k(A,B) = \max_{\substack{Ax \neq 0,\, By \neq 0, \\ Ax \perp \{Ax_1, \cdots, Ax_{k-1}\}, \\ By \perp \{By_1, \cdots, By_{k-1}\}.}} \frac{y^T B^T Ax}{\|By\|_2 \|Ax\|_2} =: \frac{y_k^T B^T A x_k}{\|By_k\|_2 \|Ax_k\|_2}. \tag{0.1}$$

The following vectors of unit length,

$$Ax_i/\|Ax_i\|_2, \quad By_i/\|By_i\|_2. \quad (i = 1, \cdots, q),$$

in (0.1) are called the canonical vectors or canonical scores of (A,B); and

$$x_i/\|Ax_i\|_2, \quad y_i/\|By_i\|_2, \quad (i = 1, \cdots, q),$$

are called the canonical weights. The angles $\theta_k \in [0, \pi/2]$ satisfying $\cos\theta_k = \sigma_k(A,B)$ are called the principal angles between $\mathcal{R}(A)$ and $\mathcal{R}(B)$, the range space of A and B, respectively.

Traditional methods for computing the canonical correlations are based on matrix inversion and eigendecomposition. A significant improvement was achieved by Björck and Golub who proposed a numerical algorithm using QR decomposition and singular value decomposition (SVD) together with a first order perturbation analysis of the canonical correlations. In this talk, we survey several of the recent results concerning the analysis and numerical computations of the canonical correlations.

We first discuss pertubation analysis of the canonical correlations. We extend the first order perturbation analysis of Björck and Golub, and derive perturbation bounds of the canonical correlations for normwise as well as componentwise perturbations. In response to the question raised by J. Demmel,[1] we also discuss the relative perturbation bounds for the canonical correlations and demonstrate that it is generally not true that small relative perturbations in A and B will result in small relative perturbations in their canonical correlations. We then identify a class of matrix pairs for which good relative perturbation bounds exist.

For the numerical computation, we concentrate on the updating problem of canonical correlations when data points are added and on computing the canonical correlations of large sparse and/or structured matrix pairs. The updating problem involves two different cases: 1) one of the matrix A or B is augmented by a column vector. This corresponds to the situation when the number of random variables in either sets is increased; 2) Both of the matrix A or B are augmented by a row vector. This corresponds to the situation when the number of observations in both sets is increased. The first case can be reduced to rank-one SVD updating. The second case is more complicated and involves both additive and mutiplicative SVD updating. We will present algorithms based one a sequence of chasing.

For large sparse and/or structured matrix pairs, we present a modification of the Lanczos algorithm. Since it has the attractive feature that it is not necessary to compute the orthonormal basis of the column space of A or B as is required in the Björck-Golub algorithm, one can take full advantage of the sparsity and/or special structure (e.g, Hankel or Toeplitz structure) of the underlying matrix pairs. We demonstrate the efficiency of the algorithm by computing the canonical correlations between the past and future of stationary time series which involves matrix pairs that are Toeplitz metrices.

We also briefly discuss several extensions and applications of the concept of canonical correlations: canonical correlations associated with a general bilinear form; canonical correlations with (in)homogeneous linear constraints; canonical correlations of several sets of random variables.

* Computer Science Department, 309 Whitmore Laboratory, The Pennsylvania State University, University Park, PA 16802, USA. Email: zha@cs.psu.edu. Part of the work was done jointly with G. Golub.

[1] J. Demmel put forward the question about relative perturbation bounds for the canonical correlations at the IMA workshop in 1992.

Residual Smoothing Techniques For Iterative Methods

Lu Zhou and Homer F. Walker
Department of Mathematics and Statistics
Utah State University

An iterative method for solving a linear system $Ax = b$ produces iterates $\{x_k\}$ with associated residual norms that, in general, need not decrease "smoothly" to zero. We consider "residual smoothing" techniques that generate a second sequence $\{y_k\}$ via a simple relation $y_k = (1 - \eta_k)y_{k-1} + \eta_k x_k$.

The first smoothing process is QMRS (Quasi-minimizal Residual Smoothing), by which QMR can be obtained from BCG. By changing the basis of the Krylov subspace and solving the induced canonical least squares problem without changing the residual of the least squares problem, we explicitly show that QMR uses a "weighted mean" technique to smooth the BCG method. In the QMR method, the upper bound of the residual norm, $\sqrt{k+1}\tau_k$, is the square root of the harmonic mean of the squares of the previous residual norms of BCG,

$$\sqrt{k+1}\tau_k = \sqrt{\frac{1}{\frac{1}{k+1}\sum_{j=0}^{k}\frac{1}{\rho_j^2}}}, \quad \rho_j = \|r_j^{BCG}\|_2,$$

and the residual vector of QMR is a convex combination of the BCG residuals,

$$r_k^{QMR} = \frac{1}{\sum_{i=0}^{k}\frac{1}{\rho_i^2}}\sum_{i=0}^{k}\frac{1}{\rho_i^2}r_i^{BCG}.$$

From above equations, we are able to derive the following,

$$r_k^{QMR} = \frac{\tau_k^2}{\tau_{k-1}^2}r_{k-1}^{QMR} + \frac{\tau_k^2}{\rho_k^2}r_k^{BCG}, \quad x_k^{QMR} = \frac{\tau_k^2}{\tau_{k-1}^2}x_{k-1}^{QMR} + \frac{\tau_k^2}{\rho_k^2}x_k^{BCG},$$

where $\frac{1}{\tau_k^2} = \frac{1}{\tau_{k-1}^2} + \frac{1}{\rho_k^2}$.

In general, we can extend this smoothing process to any iterative method to get a new sequence of iterates $\{y_k\}$ with residuals $\{s_k\}$ by

$$y_k = \eta_k y_{k-1} + (1-\eta_k)x_k, \quad s_k = \eta_k s_{k-1} + (1-\eta_k)r_k,$$

by choosing $\eta_k = \frac{\tau_k^2}{\tau_{k-1}^2}$ and updating $\frac{1}{\tau_k^2} = \frac{1}{\tau_{k-1}^2} + \frac{1}{\rho_k^2}$, where $\rho_k = \|r_k\|_2 = \|b - Ax_k\|_2$. This y_k preserves the same quasi-minimizal residual property in the QMR method which minimizes the

related least squares problem. Freund's TFQMR and QMRCGSTAB of Chan et al. can be derived directly from CGS and Bi-CGSTAB by these relations.

The second smoothing process is MRS (Minimal Residual Smoothing), in which we just choose η_k to minimize $\|s_k\|_2$, where $s_k = \eta_k s_{k-1} + (1-\eta_k)r_k$. This idea was first introduced by Schönauer in order to get a norm nonincreasing function of the iteration index. We use a slightly different approach to damp the correction steps and get an equivalent mathematical result that has some numerical advantages.

We also could consider smoothing in the form

$$y_k = (1-\eta_k)x_k' + \eta_k x_k'',$$

where $\{x_k'\}$ and $\{x_k''\}$ are given iterates; Brezinski and Redivo Zaglia suggest this with η_k chosen to minimize the residual. Creating $\{x_k'\}$ and $\{x_k''\}$ from two different iteative methods is expensive. A cheap way is that

$$x_k' = x_k, \quad x_k'' = \eta_k' y_{k-1},$$

where η_k' is simply chosen to minimize $\|b - A(\eta_k' y_{k-1})\|_2$.

Another possibility is combining given $\{x_k^{(1)}\}, \cdots, \{x_k^{(m)}\}$ to produce $\{y_k\}$ by

$$y_k = \eta_k^{(1)}x_k^{(1)} + \cdots + \eta_k^{(m)}x_k^{(m)}, \quad \sum_{i=1}^{m}\eta_k^{(i)} = 1.$$

The QMR squared method of Freund and Szeto can be obtained from the CGS iterates and certain auxiliary quantities through above relation.

K. Zietak
University of Wroclaw
Institute of Computer Science
Wroclaw, Poland

Properties of linear approximations of matrices in the spectral norm

Let M be a linear subspace of the linear space of rectangular real matrices. We consider the problem of finding the best approximation from M to a given matrix with respect to the spectral norm. The problem is a particular case of the matrix nearness problems.

In the general case the spectral approximation is not unique. Therefore we define a strict spectral approximation. We prove that it always exists and that it is unique. The concept of the strict spectral approximation of a matrix is based on the strict Chebyshev approximation of a vector introduced by Rice.

Robust Image Processing for Remote Sensing Data (Larry Ammann)

Remote sensing has become an important resource for a variety of areas including energy and mineral exploration, environmental studies, land use studies, military surveillance, and archeology. An immense amount of remote sensing data from satellites such as the Landsat and SPOT series have been, and continue to be, collected and archived. Furthermore, new satellites with improved capabilities are currently being planned and constructed. How to efficiently utilize this information is still under active study. Part of the reason for this is that the data typically consists of a large number of multivariate observations, the structure of which can change over time. For example, the Landsat Thematic Mapper gives reflectances at each of 7 spectral bands — 3 visible, 2 near-infrared, 1 mid-infrared, and 1 thermal infrared band. Thus a 1K by 1K pixel data set consists of 1,048,576 observations on 7 variables.

In order to visualize such data, one can assign an RGB color value to each of 3 selected bands and then create a false-color image. Additional images can be created by using various band ratios to generate the RGB values. A common problem with satellite data is that topographic features produce uneven illumination due to shadowing from any appreciable relief present in the scene. One way to overcome the potential confusion from this situation is to use band ratios, since (at least theorectically) a surface should receive the same proportion of energy across the spectrum without regards to its orientation to the sun, and should reflect in proportion to its spectral reflectance properties. Furthermore, particular band ratios can be selected to emphasize the differential response of various surface components to different bands. However, in some exploratory applications it may be desired to examine a large number of band ratios in order to be certain that any important feature will be detected. In such cases, the use of band ratios can greatly expand the dimensionality of the dataset since there is a potentially large number of such ratios that can be defined.

Typically, the reflectance of a feature at one wavelength is correlated with its reflectance at other wavelengths, and so some of the information contained in the individual bands and band ratios is redundant. A commonly used approach to overcome this problem is to use principal component analysis (PCA) to "decorrelate" such information. In addition, PCA is useful as a tool for contrast enhancement of images constructed from this data. However, since PCA is based on the eigenvalue decomposition (EVD) of the covariance matrix, it is highly sensitive to the presence of outliers or to subpopulations that differ from the major features of a dataset. For this reason, a statistically robust version of PCA can provide a more useful decomposition of the data in that it can characterize the structure of the major feature of a dataset without the distorting effects of different subpopulations, and at the same time produce better separation of subpopulations from the major feature.

This talk discusses the applications of robust PCA to remote sensing data. An algorithm for robust covariance estimation was derived in Ammann (1993) and is described here in the context of robust principal components. Methods for image processing based on this robust principal component analysis (RPCA) are described, including such problems as output data storage, histogram equalization and other visualization tools, identification of unusual spectral response structures or features. Examples of the application of RPCA to datasets taken from the Landsat and SPOT satellites will be presented. Finally, problems associated with very high dimensional datasets and how RPCA can be efficiently applied in such cases will be considered.

ROUND-OFF ERROR ANALYSIS OF FAST TRIGONOMETRIC TRANSFORMS AND APPLICATION TO THE CHEBYSHEV PSEUDOSPECTRAL METHOD.

M. ARIOLI[1,2] AND L. VALDETTARO[1,3]

In [1] we show that the relative error in the maximum norm for the Fast Fourier transfom (defined as $\varepsilon_\infty = \|\mathrm{fl}(F_n x) - \hat{x}\|_\infty / \|\hat{x}\|_\infty$, where $\hat{x}$ denotes the Fourier transform of the vector x and $\mathrm{fl}(F_n x)$ the computed value), is bounded theoretically by $cu\sqrt{n}$, where c is a constant of order 10, u is the machine precision (defined as the maximum positive number such that $fl(1+u) = 1$) and n is the vector length. The analysis is based on a 'worst case' analysis, where all the rounding errors contribute in the same direction. In practice, the rounding errors have a statistical distribution and as a consequence the expected value for the relative error grows only as $\log n$. This result is supported by numerical experiments on several different types of initial signals, ranging from coherent signals (having only a few harmonics excited) to white noise and to "turbulent like" signals (obtained by considering a signal with an imposed turbulent spectrum and random phases); the same logarithmic behaviour was observed on machines which support the IEEE arithmetic standard as well as on the CRAY2 computer which does not have this arithmetic.

This very good error property of the FFT is however counterbalanced by the very bad conditioning of the derivative operators. The condition number of the matrix for the k^{th} derivative in spectral space scales like n^k for the Fourier and for the Spherical Harmonics expansion, and like n^{2k} for the Chebyshev expansion (n being the order of the truncation). The computation of the derivatives of a function at the grid points in a pseudo-spectral method is done by transforming the signal from real space to spectral space, applying the derivative matrix, and returning back to real space. The roundoff error in the first transform is then amplified due to the bad conditioning of the derivative matrix.

¿From the preceding discussion we expect that the total roundoff error made in computing the k^{th} derivative is of order $cun^k \log n$ for Fourier expansions. This is well confirmed by numerical tests.

For Chebyshev expansions the analysis is a little more involved. As we have seen previously, for the Chebyshev pseudospectral method using the Gauss Lobatto grid points ($x_j = \cos(\pi j/n), j : 0 \rightarrow n$), the transformations between real space (x_j) and spectral space (a_j) are defined by:

$$f(x_j) = \sum_{k=0}^{n} a_k \cos\frac{jk\pi}{n}, \qquad a_j = \frac{2}{n} b_n \sum_{k=0}^{n} b_k f(x_k) \cos\frac{jk\pi}{n}$$

where $b_j = 1, j : 1 \rightarrow n-1, b_0 = b_n = 1/2$. In both cases we must perform a cosine transform on the original vector. Special care must be taken when implementing the cosine transform, otherwise additional error is incurred. In [2], we analyze the four methods which are commonly used:

- The most straightforward way is to define a vector z of length $2n$:

$$z_0 = x_0$$
$$z_n = x_n$$
$$z_j = \frac{1}{2} x_j : \; j = 1, ..., n-1 \qquad z_j = z_{2n-j} : \; j = n+1, ..., 2n$$

  The vector z has the symmetry $z_i = z_{2n-i}$. We will call such a sequence Even Symmetric. The complex FFT of z is precisely the cosine transform of the original signal. The roundoff error for this transform is the same as that for the FFT, and the total error in computing the derivative scales like $n^{2k} \log n$. There are however two drawbacks: it requires twice the array storage because it uses a vector of double length, and it requires more computational work than is necessary.
- The second method takes into account directly the symmetries of the sequence. It is shown in [3] that the Fourier transform of an Even Symmetric sequence can be done in less operations than the FFT. The idea is that one identifies the intermediate symmetries that occur in the FFT of the symmetric sequence, and uses these symmetries to eliminate the duplicate or zero computations. The computations needed reduce to half of those for the full FFT. The error analysis is the same as that for the FFT, since the computations are a subset of those done during the FFT. The drawback of this method is that it is not straightforward to implement. While the FFT's are generally provided by the vendors of a machine in a very optimized form, the symmetric FFT's must be hand coded by the user.
- The third and fourth method are similar. They require half the operations and half the storage of the first method. They are based on a pre-processing stage, which is followed by a complex FFT transform of length n and a post-processing stage (see [4], Section 4.4, for more details). The total error in these cosine transforms scales like $n \log n$, and the k^{th} derivative is computed with a roundoff error which scales like $n^{2k+1} \log n$.

Summarizing, the straightforward cosine transform requires more computational cost and memory, but it is much less affected by roundoff error. Since the derivative matrix introduces a very large amplification of the error, it might be preferable to use the straightforward transform when high resolutions are required.

REFERENCES

[1] M. Arioli, H. Munthe-Kaas, and L. Valdettaro. *Componentwise error analysis for FFT's with applications to fast Helmholz solvers* Technical Report TR/IT/PA/91/55, CERFACS, 1991.

[2] M. Arioli and L. Valdettaro. *Roundoff error analysis of Fast Cosine Transforms and spectral derivatives* Technical Report in preparation.

[3] P. N. Swarztrauber (1986) *Symmetric FFTs Math. Comp.* 47, 323-346.

[4] C. Van Loan (1992) *Computationl frameworks for the Fast Fourier Transform SIAM, Frontiers in Applied Mathematics Serie.*

[1] Centre Européen de Recherche et de Formation Avancée enCalcul Scientifique, 42 av. G. Coriolis, 31057 Toulouse Cedex, France.

[2] Istituto di Analisi Numerica, Consiglio Nazionale delle Ricerche, corso Carlo Alberto 5, 27100 Pavia, Italy.

[3] Centro di Ricerca, Sviluppo e Studi Superiori in Sardegna, via N. Sauro 10, 09123 Cagliari, Italy.

The rate of convergence of the conjugate gradient method

O. Axelsson
Faculty of Mathematics and Informatics
University of Nijmegen, The Netherlands

Abstract

The rate of convergence of conjugate gradient type methods is considered. Frequently it is observed that the convergence of the norm of the residuals or iteration errors take place in three phases: an initial phase with rapid decay during the first few iteration steps, a middle phase of essentially linear rate and a final phase of superlinear rate of convergence. This is explained here using various quantitative estimates involving the eigenvalues of the iteration matrix. The case where the smallest eigenvalues are well separated is analysed in particular detail.

For the case of real and positive nondegenerate eigenvalues an expression is derived showing that for each relative accuracy ε, there is an optimal number p of small eigenvalues involved in the estimate in addition to the spectral condition number of the remainder of the spectrum. Also the actual distribution of the Fourier coefficients of the initial vector is involved. In this way using certain best polynomial approximations this is shown to explain the observed behaviour of the rate of convergence. It can even explain small details in the convergence during the seemingly linear rate phase. The above is an improvement of earlier estimates by Axelsson (1976), Jennings (1977) and Axelsson/Lindskog (1986).

An alternative à priori qualitative estimate has been derived by I.E. Kaporin and involves the K-condition number $(1/n\ tr(A))^n/det(A)$, where n is the order of A. Hence, here all eigenvalues are involved. However, for typical distributions of the eigenvalues it is shown that this estimate gives less accurate bounds, which can even be incorrect in its order of magnitude as a function of n, $n \to \infty$. This bound will be improved here, but it can then not be used as an à priori estimate.

A third type of estimates of qualitative type, i.e. not giving à priori bounds for the number of iterations have been derived by van der Sluis/van der Vorst. They involve the behaviour of the Ritz values for the corresponding "Lanczos" matrix. A comparison with our quantitative estimate is presented.

Finally the first class of estimates (involving only some isolated eigenvalues and a condition number for the remainder of the spectrum) are extended to the case of complex matrices and to degenerate matrices. These type of estimates are applicable for Orthomin type methods and for orthogonal residual type methods. It is examinated how the following numbers influence the rate of convergence

(i) The order of the Jordan boxes
(ii) The condition number of the transformation matrix S (to Jordan canonical form)
(iii) $\lambda_{min}(1/2\,(A+A^T)$ and $||A||_2$.

Among other things it is shown situations where cond(S) does not influence the number of iterations at all, even when cond(S) is very large.

The above estimates give the most complete explanation of the rate of convergence of the conjugate gradient method among publications known to the author.

References

O. Axelsson, A class of iterative methods for finite element equations, Comp. Math. Appl. Mech. Eng., 9(1976), pp. 123-137.

O. Axelsson and G. Lindskog, On the rate of convergence of the preconditioned conjugate gradient method, Numer. Math., 48 (1986), pp. 499-523.

A. Jennings, Influence of the eigenvalue spectrum on the convergence rate of the conjugate gradient method, J. Inst. Maths. Applcs, 20 (1977), pp. 61-72.

Y. Notay, On the convergence rate of the conjugate gradients in presence of rounding errors, Numer. Math., to appear.

O. Axelsson and M. Makarov, On a generalized conjugate gradient orthogonal residual method, Faculty of Mathematics and Informatics, University of Nijmegen, 1993.

O. Axelsson, On the rate of convergence of the conjugate gradient method, Conference of Numerical Linear Algebra, Fudan University, Shanghai, October, 1992.

I.E. Kaporin, An alternative approach to the estimation of the iteration number of the conjugate gradient method, Numerical Methods And Software, Acad. Sci. USSR, Department of Numerical Mathematics, Moscow, 1990, pp. 55-72 (In Russian).

I.E. Kaporin, New convergence results and preconditioning strategies for the conjugate gradient method, submitted, 1992.

A. Greenbaum and Z. Strakos, Matrices that generate the same Krylov residual spaces, IMA Preprint Series # 983, Institute for Mathematics and its Applications, University of Minnesota, 1992.

A. van der Sluis and H. van der Vorst, The rate of convergence of conjugate gradients, Numer. Math,. 48 (1986), pp. 543-560.

Progress in the Numerical Solution of the Nonsymmetric Eigenvalue Problem

Zhaojun Bai *

Abstract

With the growing demands from disciplinary and interdisciplinary fields of science and engineering for the numerical solution of the nonsymmetric eigenvalue problem, competitive new techniques have been developed for solving the problem. In this presentation, we examine the start-of-the-art of the algorithmic techniques and the software scene for the problem. Some current developments are also outlined.

Extended Abstract

Over several years working on the LAPACK project, and on algorithm and software development of the nonsymmetric eigenvalue problem and communication with a variety of users who work in diverse fields involving scientific computing, the author have seen a growing demand for the numerical solution of the nonsymmetric eigenvalue problems. Meanwhile, in numerical analysis community, since Parlett's exploratory review paper entitled "The Software Scene in the Extraction of Eigenvalues from Sparse Matrices" nearly one decade ago, and with the successful development of the symmetric eigenvalue problem, many new numerical methods and analysis have been developed for the nonsymmetric eigenproblem. The aim of this work is to review the origins of the problem and the progress of the numerical techniques over the past decade, and to share our view and expertise within scientific computing community.

The survey is by no means complete. One reason for this is that relevant articles may be found scattered throughout the scientific and engineering literature, and the task of tracking them all down is impossibly large. The author apologizes for the ignorance of some important contributions to the problem that are not mentioned here. A new book by Saad is an elegant source for studying the start-of-the-art in large eigenproblem techniques. This review will only focus on the nonsymmetric eigenvalue problem in the aspects of its origins, *algorithmic* techniques, software scene and work in progress.

As defined by Parlett one decade ago, there are two different user groups for the eigenproblem. One is called *intensive* user group and the other called *sporadic* user group. For the former group, spectral analysis is imperative to their entire work; they have been spending tremendous efforts in terms of times and funding for extracting the desired spectral information. But for the latter group, the need to compute eigenvalues arises occasionally and the user wants to obtain them with minimal fuss.

With the rapid advances of computer facilities, in particular the massively parallel computers, and the new engagement of interdisciplinary scientific computing activities, as proposed by J. W. Demmel, there are two different camps in each user group according to their different desired priorities, in terms of computation details, reliability and execution time, of a program. The first camp is made up of traditional library users, and the second camp is made up of high performance computing researchers. For the first camp, the desiderata can be characterized as follows:

1. easy user interface with hidden computation details,
2. reliability; the code should fail as rarely as possible,
3. execution time.

However for the second camp, the desiderata are

1. execution time,
2. be able to access to internal details to tune data structures to their applications,
3. reliability; A program should expend only a negligible amount of time, space or code in checking or taking precautions against rare eventualities that the user knows may never arise for his or her particular applications.

These different desiderata give an extra dimension to numerical algorithm development and analysis. To what extent can we satisfy both camps? In this presentation, we will try to address this interesting question with respect to the nonsymmetric eigenvalue problem.

*Current Address: Department of Computer Science, Texas A&M University, College Station, TX 77842, e-mail address: bai@cs.tamu.edu

Error Analysis of the Lanczos Algorithm for the Nonsymmetric Eigenvalue Problem

Zhaojun Bai*

Abstract

This work presents an error analysis of the Lanczos algorithm in finite-precision arithmetic for solving the standard nonsymmetric eigenvalue problem, if no breakdown occurs. An analogy of Paige's theory on the relationship between the loss of orthogonality among the Lanczos vectors and the convergence of Ritz values in the symmetric Lanczos algorithm is discussed in this paper. The theory developed illustrates that in the nonsymmetric Lanczos scheme, if Ritz values are well conditioned, then the loss of biorthogonality among the computed Lanczos vectors implies the convergence of a Ritz triplet in terms of small residuals. Numerical experimental results confirm this observation. The results of such error analysis provide insight into the need for robustness schemes, such as look-ahead strategies, which attempt to avoid the potential breakdown and instability in the nonsymmetric Lanczos procedure.

Extended Abstract

This work is concerned with an error analysis of the Lanczos algorithm for solving the nonsymmetric eigenvalue problem of a given real $n \times n$ matrix A. In the applications of interest, the matrix A is usually large and sparse, and only a few eigenvalues and eigenvectors of A are wanted. In [1], a collection of such matrices is presented describing their origins in problems of applied sciences and engineering.

The Lanczos algorithm, proposed by Cornelius Lanczos in 1950 [7], is a procedure for successive reduction of a given general matrix to a nonsymmetric tridiagonal matrix. The eigenvalue problem for the latter matrix is then solved. The remarkable feature in practice is that in this procedure, a few eigenvalues of A (often the largest ones in algebraic magnitude) appear as the eigenvalues of a smaller reduced tridiagonal matrix. The scheme references the matrix A only through the matrix-vector products Ax and $A^T x$; hence the structure of the matrix is maintained, which renders the scheme particularly useful for finding a few eigenvalues of a very large and sparse problem.

*Current Address: Department of Computer Science, Texas A&M University, College Station, TX 77842, e-mail address: bai@cs.tamu.edu

In the 1970's and 80's, great progress has been made on the Lanczos algorithm for solving a large linear system of equations with symmetric coefficient matrix and the symmetric eigenvalue problem. Paige [8] was the first to give an error analysis of the Lanczos algorithm in finite-precision arithmetic. Later, Parlett, Scott, Grcar, Simon, Greenbaum, Strakos and many others [9, 12, 5] presented further analysis of the Lanczos scheme and its variants. These analyses conclude that the loss of orthogonality among the computed Lanczos vectors is not necessarily a calamity, since it accompanies the convergence of a group of Ritz values to the eigenvalues of the original matrix. Today, the Lanczos algorithm is regarded as the most powerful tool for finding a few eigenvalues of a large symmetric eigenvalue problem. Software, developed by Parlett and Scott [9] and Cullum and Willoughby [3] can be accessed via *netlib*, a software distribution system.

In recent years, there has been considerable interest in the Lanczos algorithm for solving linear systems of equations with nonsymmetric coefficient matrix and the nonsymmetric eigenvalue problem. Parlett, Taylor and Liu [10], Freund, Gutknecht and Nachtigal [4] have proposed robust schemes for overcoming possible failure (called *breakdown*), or huge intermediate quantities (called *instability*) in the nonsymmetric Lanczos procedure. A theoretical investigation of the possible breakdown and instability of the nonsymmetric Lanczos procedure is made by Gutknecht [6], Boley et al [2] and Parlett [11].

Compared to the existing sophisticated error analysis of the Lanczos algorithm for the symmetric eigenvalue problem, much less progress has been made on error analysis of the nonsymmetric Lanczos algorithm. In this work, we give an error analysis for the simple nonsymmetric Lanczos algorithm and study the effects of finite-precision arithmetic. In the spirit of Paige's floating-point error analysis for the symmetric Lanczos algorithm [8], based on the rounding error model of the basic sparse linear algebra operations, such as saxpy, inner product, and matrix-vector multiplication, we present a set of matrix equations which govern all computed quantities of the simple nonsymmetric Lanczos algorithm in finite-precision arithmetic. An analogy of Paige's theory on the relationship between the loss of orthogonality among the computed Lanczos vectors and the convergence of a Ritz value for the symmetric eigenvalue problem is also discussed in this work. We conclude that if Ritz values are well conditioned, then the loss of biorthogonality among the computed Lanczos vectors implies the convergence of Ritz triplets in terms of small residuals. The error analysis results developed in this work also provide insight into the need for robustness schemes, such as look-ahead strategies [10, 4], to avoid the potential breakdown and instability in the nonsymmetric Lanczos algorithm.

References

[1] Z. Bai, *A collection of test matrices for the large sparse nonsymmetric eigenvalue problem*, in preparation, (1992)

[2] D. Boley, S. Elhay, G. H. Golub and M. H. Gutknecht, *Nonsymmetric Lanczos and finding orthogonal polynomials associated with indefinite weights*, Numerical Analysis Report NA-90-09, Stanford, Aug. (1990)

[3] J. Cullum and R. A. Willoughby, *Lanczos algorithms for large symmetric eigenvalue computations, Vol. 1, Theory, Vol.2, Programs*; Birkhäuser, Basel, (1985)

[4] R W. Freund, M. H. Gutknecht and N. M. Nachtigal, *An Implementation of the Look-Ahead Lanczos Algorithm for Non-Hermitian Matrices, Part I.* Tech. Rep. 90.45, RIACS, NASA Ames Research Center, Nov. (1990)

[5] A. Greenbaum, *Behavior of slightly perturbed Lanczos and conjugate-gradient recurrences*, Lin. Alg. Appl. 113, pp. 7–63, (1989)

[6] M. H. Gutknecht, *A completed theory of the nonsymmetric Lanczos process and related algorithms, Part I, II*, IPS Res. Rep. No. 90-10, Zürich, (1990)

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

[8] C. Paige, *Error analysis of the Lanczos algorithm for tridiagonalizing a symmetric matrix*, J. Inst. Math. Appl. 18, pp. 341–349 (1976)

[9] B. N. Parlett and D. S. Scott, *The Lanczos algorithm with selective reorthogonalization.* Math. Comp. 33, pp.217–238, (1979).

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

[11] B. N. Parlett, *Reduction to tridiagonal form and minimal realizations*, SIAM J. Mat. Anal. Appl. 13, pp.567–593,(1992).

[12] H. Simon, *Analysis of the symmetric Lanczos algorithm with reorthogonalization methods*, Lin. Alg. Appl. 61, pp.101–131, (1984)

Design of a Parallel Nonsymmetric Eigenroutine Toolbox

Zhaojun Bai and James Demmel
Texas A&M
College Station, TX

U. C. Berkeley
Berkeley, CA

Long Abstract:

It is a challenge to design a parallel algorithm for the nonsymmetric eigenproblem which scales for larger problems on larger machines, uses coarse grain parallelism effectively, deals with highly nonnormal matrices and strongly clustered spectra, and does not waste time dealing with the parts of the spectrum in which the user is not interested.
The conventional Hessenberg QR algorithm is a fine grain algorithm and has proven to be difficult to parallelize. Moreover, it finds all the eigenvalues, and essentially just one (or a few) at a time. In applications where only some eigenvalues are desired, one still has to compute all of them.
If one only wants an invariant subspace corresponding to a specified set of eigenvalues, one has to reduce the matrix completely to Schur form, and then swap the desired eigenvalues along the diagonal to group them together in order to form the desired invariant subspace.

In this working note, we propose a collection of tools from which hybrid eigenvalue algorithms may be constructed. The new tools we propose use the matrix sign function to both ''divide and conquer'' the matrix, as well as count the number of eigenvalues in a region of the complex plane.
We describe how these tools might be combined to deal with different kinds of spectra and user needs. We do not attempt to design a ''black box'' for this problem since we believe that such an algorithm would necessarily be much less efficient and reliable than one tuned for a particular application. All our algorithmic building blocks use at least several times as many flops as serial Hessenberg QR, but these are almost entirely in large and efficiently parallelizable block operations.
Furthermore, if we succeed in initially ''dividing and conquering'' the matrix at least a few times, most of the cost will be in these initial reductions.

Due to the potential ill-conditioning of the matrix inversion required for the matrix sign function, we also propose to use a ''staircase'' type algorithm as a preprocessing step to deflate near zero eigenvalues.
This will permit us to compute the sign function with forward and backward error proportional to the square root of machine precision. In other words, we only propose to achieve half precision in our implementation in order to save time.

In recent years, there have been similar efforts by Dongarra and Sidani, T.-Y. Li and Z. Zeng, Auslander, Lederman, Tsao and Turnbull, and Lin and Zmijewski. In the work of Dongarra and Sidani, an upper Hessenberg matrix is divided and conquered by setting the middle subdiagonal to zero, solving the two resulting subproblems recursively and in parallel, and then merging the two subparts using Newton's method. It fails a small but nonnegligible fraction of the time because Newton's method fails to converge, or converges to the wrong solution. Li and Zeng divide an upper Hessenberg matrix the same way, but merge the subparts using homotopy continuation applied to the determinant. It can also fail occasionally.
In work of Auslander, Tsao, Lederman and Turnbull, a sequence of polynomials is applied to map one part of the spectrum to zero, and the other part to one, and then the matrix is split into the corresponding invariant subspaces.
The resulting scheme is rich in matrix-matrix multiplies.
However, the restriction to polynomial mappings means the scheme is only suitable for finding eigenvalues of matrices with real spectrum, a very small fraction of real nonsymmetric matrices.
In work by Lin and Zmijewksi, instead of polynomial mappings, the matrix sign function is used, which can be regarded as a stabilized version of a scheme proposed by Beavers and Denman.
The goal is a ''black box'' for the eigenproblem.

In some sense the algorithms we propose using here are less sophisticated and reliable than their serial counterparts. This reflects the fact that it is only such simple, large block algorithms which map well to current massively parallel machines. This is something of a departure from traditional numerical analysis which can tend toward ever more sophisticated algorithms, but it reflects how well (or how poorly) we can currently exploit these machines.

The talk will discuss basic computational building blocks, existing iteration schemes for the sign function, numerical examples, and future research.

Efficient and Stable Algorithms for Downdating Two-sided Orthogonal Decompositions

Jesse L. Barlow Hongyuan Zha[*] Peter A. Yoon[†]

December 14, 1992

We discuss methods for downdating three different types of complete orthogonal decomposition of an $m \times n$ matrix A where $m \geq n$. Two of the methods are new. The three types of decomposition can be characterized by writing them in the form

$$A = U \begin{pmatrix} C \\ 0 \end{pmatrix} V^T \tag{1}$$

where $U \in \Re^{m \times m}$ and $V \in \Re^{n \times n}$ are orthogonal, and $C \in \Re^{n \times n}$ has one of the forms

$$C = \begin{pmatrix} R & S \\ 0 & T \end{pmatrix} \quad \begin{array}{l} R \in \Re^{k \times k}, T \in \Re^{(n-k)\times(n-k)} \text{ upper triangular} \\ \| (S^T\, T^T) \|_F \leq \epsilon \end{array} \tag{2}$$

$$C = \begin{pmatrix} L & 0 \\ F & G \end{pmatrix} \quad \begin{array}{l} L \in \Re^{k \times k}, G \in \Re^{(n-k)\times(n-k)} \text{ lower triangular} \\ \| (F\, G) \|_F \leq \epsilon \end{array} \tag{3}$$

$$C = \Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_n) \quad \begin{array}{l} \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \\ \| (\sigma_{k+1}, \ldots, \sigma_n) \| \leq \epsilon \end{array} \tag{4}$$

Here k is the computed rank of A and ϵ is related to the tolerence used to determine that rank. We use $\| \cdot \|$ to denote the Euclidean norm, $\| \cdot \|_F$ to denote the Frobenius norm, and $\| \cdot \|_{2,F}$ is for expressions that hold for both norms. For (2) and (3) it is presumed that some condition estimator has concluded that $\| L^{-1} \|_p^{-1} > tol$ or $\| R^{-1} \|_p^{-1} > tol$, where $p = 1, 2, \infty$ and $\epsilon = \sqrt{n-k} * tol$. For (4) we presume $\sigma_k \geq tol$.

[*] Computer Science Department, Penn State University, University Park, PA 16802-6103. e-mail: barlow@cs.psu.edu, zha@cs.psu.edu. The research of Jesse L. Barlow was supported by the National Science Foundation under grant no. CCR-9201692.

[†] Computer Science Department and Applied Research Laboratory, The Pennsylvania State University, University Park, PA 16802. The research of Peter A. Yoon was supported by the Office of Naval Research under the Fundamental Research Initiatives Progam.

The forms (2,3) are the URV decomposition and the ULV decomposition described by Stewart, and the form (4) is the familiar singular value decomposition(SVD), We refer to all of these decompositions as two-sided orthogonal decompositions(TSO decompostions). They are what Lawson and Hanson call HRK decompositions.

The downdating problem is that of obtaining the TSO decomposition of $\bar{A}$ where

$$A = \begin{pmatrix} w^T \\ \bar{A} \end{pmatrix}.$$

Here A denotes a matrix whose TSO decomposition is known, and w denotes a row of observations that we wish to delete. The opposite computation, updating, consists of obtaining the TSO decomposition of A from that of $\bar{A}$. Updating and downdating are important in signal processing and statistical applications.

The problem can be transformed into a problem of finding a matrix $\bar{C}$ of the same form as C, and an orthogonal matrix $\bar{V}$ such that

$$C^T C - zz^T = \bar{V} \bar{C}^T \bar{C} \bar{V}^T \tag{5}$$

where $z = V^T w$.

Our approaches to downdating the decompositions (3)-(4) use ideas from "chasing" algorithms and from the downdating algorithm due to Gill et al.

The chasing strategy for the SVD was originated by Rutishauser, recently considered by Abdallah and Hu, and improved by Zha. Van Huffel and Park extended Zha's result so that it can be applied to partial SVDs. Stewart gives chasing strategies for the URV and ULV decompostions. All of the above strategies were advocated only for updating, not downdating.

By themselves, these chasing procedures have two closely related weaknesses:

1. The numerical rank may no longer be "revealed" by the form of $\bar{C}$.
2. The small singular values of $\bar{C}$ may be very inaccurate.

We show that the downdating method of Gill et al. can be used to obtain forms that once again "reveal the rank".

The following are the main results of this paper:

- A blockwise procedure for downdating the ULV decomposition where

$$\bar{C} = \begin{pmatrix} \bar{L} & 0 \\ \bar{F} & \bar{G} \end{pmatrix} \quad \begin{array}{l} \bar{L}, \bar{G} \text{ lower triangular} \\ \| (\bar{F}\, \bar{G}) \|_{2,F} \leq \| (F\, G) \|_{2,F} \end{array} \tag{6}$$

  where the blocks are conformal with (3).

- A demonstration that the downdating technique of Gill et al. has a property similar to (6) for the URV decomposition.

- A procedure for downdating the SVD which obtains a bidiagonal form such that

$$\bar{C} = \begin{pmatrix} \bar{B}_1 & 0 \\ \bar{\gamma}_k e_1 e_k^T & \bar{B}_2 \end{pmatrix} \quad \begin{array}{l} \bar{B}_1, \bar{B}_2 \text{ lower bidiagonal} \\ \| B_2 \|_F^2 \leq \sum_{i=k+1}^{n} \sigma_i^2 \end{array} \tag{7}$$

  where the blocks are conformal with (4). This form preserves more of the accuracy of the small singular values. We can then use one of several algorithms to find the singular values of the bidiagonal matrix $\tilde{C}$ to relative accuracy.

- A perturbation theory for the singular values and vectors from downdated matrices and blockwise error bounds for the above procedures.

(2) Abstract:

Title: Structure preserving difference schemes for matrix differential equations

Authors: Simon Bell and Nancy K. Nichols
University of Reading

Abstract:

Matrix differential equations often have solutions with special structure, and numerical methods for generating approximate solutions that retain the special structure are desirable. We consider here systems for which the solution matrix is orthogonal.

Most numerical methods for solving such systems do not preserve orthogonality of the approximate solutions (even ignoring round-off errors). In this paper we present two simple second-order difference schemes that preserve orthogonality. The primary motivation for developing these schemes arises from the computation of the analytic singular value decomposition (ASVD) of a time-varying matrix A(t). In order to find continuous left and right singular factors of the matrix, orthogonal solutions of certain differential systems are needed.

We consider the following class of differential equations

$$dX/dt = Z(X,t)\,X \,, \qquad X(0) = X_0$$

where the matrix Z is skew-symmetric for all X and t. Under mild assumptions on continuity, if the initial matrix $X(0)$ is orthogonal, then the system has a unique solution that is orthogonal.
Two numerical schemes for solving this system are proposed. It is established that orthognality of the results (in exact arithmetic) is preserved. Consistency and stablility of the schemes are also shown. For linear systems, where $Z = Z(t)$ is not dependent on X, the schemes are explicit; for non-linear systems, an iteration procedure that preserves orthogonality for every iterate is derived and shown to converge under simple assumptions.

Numerical examples are presented to illustrate the behaviour of the schemes. Applications to the computation of the ASVD of a matrix are also described. The proposed methods can be generalized to higher order schemes.

An Oblique Projection Method for Solving Sparse Linear Systems

by Michele Benzi and Carl D. Meyer
Mathematics Department
North Carolina State University
Raleigh, NC 27695-8205

ABSTRACT

An oblique projection method is adapted to solve large sparse unstructured systems of linear equations. This row-action technique is a direct method which can be interpreted as an oblique Kaczmarz-type algorithm which converges in exactly n-1 steps where n is the size of the problem. When a sparsity-preserving pivoting strategy is incorporated, it is demonstrated that the technique can be superior in terms of both fill-in and arithmetic complexity to more standard sparse algorithms based on Gaussian elimination.

Improved Parallel Computations with Toeplitz-like and Hankel-like Matrices

Dario Bini
Dipartmento di Matematica
Università di Pisa
56100 Pisa, Italy

Victor Pan
Math & Computer Science Dept.
Lehman College, CUNY
Bronx, NY 10468 USA

(Supported by NSF Grant CCR 9020690, by
CUNY University Award 662478 and by 60% and 40% MURST funds)

Summary. The known parallel algorithms for computations with general Toeplitz, Hankel, Toeplitz-like, and Hankel-like matrices are inherently sequential. We develop some new techniques in order to devise fast parallel algorithms for such computations, including the evaluation of Krylov sequences for such matrices, traces of their power sums, characteristic polynomials and generalized inverses. This has further extensions to computing the solution or a least-squares solution to a linear system of equations with such a matrix and to several polynomial evaluations (such as computing gcd, lcm, Padé approximation and extended Euclidean scheme for two polynomials), as well as to computing the minimum span of a linear recurrence sequence. The algorithms can be applied over any field of constants, with the resulting advantages of using modular arithmetic. The algorithms consist of simple computational blocks (mostly reduced to fast Fourier transforms, FFT's) and have potential practical value. We also develop the techniques for extending all our results to the case of matrices representable as the sums of Toeplitz-like and Hankel-like matrices and in addition show some more minor innovations, such as an improvement of the transition to the solution to a general Toeplitz linear system $T\mathbf{x} = \mathbf{b}$ from two computed columns of T^{-1}, which extends the previous result of Ammar and Gader for symmetric Toeplitz systems.

Key words: Toeplitz matrices, Hankel matrices, parallel algorithms, displacement operator, displacement rank, Krylov sequences, polynomial gcd, linear recurrence, least-squares solution.

1991 Math Subject Classification: 65F05, 65Y05, 68Q40, 68Q25, 15A09

Unstructured Grid Problems on SIMD Machines with Application to Iterative Methods

Petter E. Bjørstad [*]   Robert S. Schreiber [†]

Large computational problems defined from highly unstructured discretizations, of three dimensional geometries, represent considerable challenges for the design of efficient parallel algorithms. In particular, there is a popular belief that these problems are very difficult to handle on SIMD-style data-parallel machines.

As an important model case of such a problem, we consider the efficient implementation of iterative methods where the computation of matrix-vector products are essential. We assume that the number of such products (with the same non-zero structure) is so large that preprocessing of the data can be afforded.

We consider a modular approach that can be divided into the following steps:

1. Partitioning of the grid using for example a spectral algorithm.
2. Further minimizing the inter-processor communication requirement by advanced mapping algorithms that assign the gridpoints of the mesh to processors.
3. The use of a router compiler to determine an efficient sequence of communication steps on the (nearest neighbor) communication network of the target machine.
4. The actual execution of the necessary communication and computation of the matrix-vector product during an iterative procedure.

We rely on work and software by other scientists for steps one [?, ?] and two [?], the emphasis in our work is on the two last steps.

In the actual implementation of our method, we focus on a SIMD style machine having a two dimensional mesh of processor interconnections. The MasPar MP-1 and MP-2 machines belong to this class and may (at first sight) appear unsuited for unstructured mesh calculations. According to the latest NAS Parallel Benchmark, this machine is among the most cost effective MPP machines in the industry. It is therefore of considerable interest to investigate how well this architecture can handle unstructured communication.

The talk will discuss our discoveries about routing in this situation. In particular, we report on:

- The use of longer distance communication compared to nearest neighbor only
- The use of diagonal communication
- The adaptive nature of our algorithm
- The balance of arithmetic and communication for complex 3-D unstructured grids
- The parallel implementation of the router compiler in Fortran 90
- The incorporation in a state of art iterative method

We show that one can implement these ideas achieving a good balance between arithmetic steps and communication steps for complex three dimensional unstructured grids mapped to a two dimensional machine. This holds true even in the case where the number of gridpoints per processor is quite modest. We present results from a test implementation of the complete process outlined above.

[*] Institutt for Informatikk, University of Bergen, N-5020 Bergen, Norway. This work was performed when the author visited RIACS

[†] Research Institute for Advanced Computer Science, Mail Stop T045-1, NASA Ames Research Center, Moffett Field, CA 94035-1000.

The periodic Schur decomposition. Algorithms and applications

Adam Bojanczyk
Cornell University, Dept. Electrical Engineering
Ithaca, NY 14853-3801

Gene Golub
Stanford University, Dept. Computer Science
Stanford, CA 94305

Paul Van Dooren
University of Illinois at Urbana-Champaign, Coordinated Science Laboratory
1308 W. Main Str., Urbana, IL 61801

Abstract.

In this paper we derive a unitary eigendecomposition for a sequence of matrices which we call the *periodic Schur decomposition*. We prove its existence and discuss its application to the solution of periodic difference equations arising in control. We show how the classical QR algorithm can be extended to provide a stable algorithm for computing this generalized decomposition. We apply the decomposition also to cyclic matrices and two point boundary value problems.

A General Model for Orthogonal Projection Methods

James R. Bunch *
Department of Mathematics
University of California, San Diego
9500 Gilman Drive
La Jolla, California 92093-0112

Ricardo D. Fierro
Department of Mathematics
University of California, Los Angeles
405 Hilgard Ave
Los Angeles, California 90024-1555

Abstract When the overdetermined system of linear equations $AX \approx B$ has no solution, compatibility may be restored by an orthogonal projection method. The idea is to determine an orthogonal projection matrix P (or R) by some method M such that $[\tilde{A}\ \tilde{B}] = P[A\ B]$ (or $[\tilde{A}\ \tilde{B}] = [A\ B]R$), and $\tilde{A}X = \tilde{B}$ is compatible. A compatibility condition for the lower rank approximation and subspace properties of $\tilde{A}$ in relation to the nearest rank-k matrix to A will be discussed briefly. We establish a model for a general orthogonal projection method M, such as least squares (LS), total least squares (TLS), or the rank revealing QR factorization RRQR, by reformulating the parameter estimation problem as an equivalent problem of nullspace determination. When the method is based on the singular value decomposition of the matrix $[A\ B]$, the model specializes to the well known TLS method. Further, denote by X_M the minimum norm solution to $\tilde{A}X = \tilde{B}$ using method M. We find upper and lower bounds for $\|X_M - X_{TLS}\|$, where X_{TLS} denotes the TLS solution. We consider M = LS and RRQR.

* Speaker

Unitary Hessenberg Methods for Toeplitz Approximations and Applications

Angelika Bunse-Gerstner and Chunyang He

Let T_N be an $N \times N$ Hermitian positive definite Toeplitz matrix and let $H(\rho) = H(\gamma_1, \ldots, \gamma_{N-1}, \rho)$ be a unitary Hessenberg matrix in parameterized form. If the parameters $\{\gamma_k\}_1^{N-1}$ are the reflection coefficients from the Levinson algorithm applied to T_N, then for any ρ on the unit circle the Cholesky decomposition of T_N is given by

$$T_N = t_0(e_1, H(\rho)e_1, \ldots, H^{N-1}(\rho)e_1)^H(e_1, H(\rho)e_1, \ldots, H^{N-1}(\rho)e_1).$$

Thus we can associate to T_N a family of unitary Hessenberg matrices $H(\rho)$. This relation can be used to develop numerical methods for the basic signal processing problem of approximating a signal by a sum of an sinusoids or more general a sum of exponentials. The frequencies and amplitudes are determined such that the Toeplitz matrix of autocorrelation lags of the signal is approximated in some sense by the corresponding Toeplitz matrix of the model. Two such models are the Pisarenko frequency model (PFM), and the the composite sinusoidal model (CSM) by Sagayama and Itakura. Here the amplitudes and frequencies are determined such that the first autocorrelation lags are matched. The relationship between the Toeplitz matrix and the unitary Hessenberg matrices has already been used in several papers by Ammar, Gragg and Reichel to improve the numerical computation of the required quantities by making use of special unitary eigenvalue methods for the Hessenberg matrix.

Here we consider the Toeplitz approximation model (TAM) introduced by Kung in 1981. The task is here to find an $N \times N$ Toeplitz matrices of prescribed rank n_1, such that $T_N - T_1$ is a Toeplitz matrix of rank n_2 with minimal trace.

Assuming that $n_1 + n_2 = N$ the relationship with the unitary Hessenberg matrix $H(\rho)$ can be used to see that the TAM is equivalent to the problem of finding

$$\min_{\rho} \sum_{j=1}^{n_2} h_{l_j}^2(\rho),$$

where $H(\rho) = X(\rho)\Lambda(\rho)X(\rho)^H$ is the spectral decomposition of $H(\rho)$, $h_j(\rho) = \sqrt{t_0}|x_{1j}(\rho)|$ and $h_j(\rho)$ is the j-th smallest $h_j(\rho), j = 1, \ldots, N$.

We present formulas for the derivatives of the eigenvalues and eigenvectors of $H(\rho)$ and use Newtons method to solve the optimization problem.

A. Bunse-Gerstner
Fachbereich Mathematik und Informatik
Universität Bremen
Postfach 33 04 40
D 2800 Bremen 33
FRG
electronic mail: angelika@mathematik.uni-Bremen.de

Chunyang He
Fakultät für Mathematik
Universität Bielefeld
Postfach 10 01 31
4800 Bielefeld 1
FRG
electronic mail: he@math5.mathematik.uni-Bielefeld.de

Distances and Conditioning in Computational Control

Ralph Byers
Department of Mathematics
University of Kansas
Lawrence, Kansas 66045

In this talk we investigate numerical methods for measuring the distance from a controllable system to the nearest uncontrollable system and briefly discuss some related problems. A linear control system

$$\begin{aligned} E\frac{dx}{dt} &= Ax + Bu \\ y &= Cx \end{aligned}$$

E, $A \in R^{n\times n}$, $B \in R^{n\times p}$, is said to be (strongly) controllable if

1. For all $\lambda \in C$, $[\lambda E - A,\ B]$ has full rank, and
2. If the columns of S_∞ form a basis of the null space of E, then $[E,\ AS_\infty,\ B]$ is full rank.

The distance to uncontrollability is defined by

$$\nu_{\mathcal{D}}(E,A,B) = \inf\left\{ |||[\Delta E,\ \Delta A,\ \Delta B]||| \;\middle|\; \begin{array}{l} (E+\Delta E, A+\Delta A, B+\Delta B) \text{ is un-} \\ \text{controllable and } (\Delta E,\ \Delta A,\ \Delta B) \\ \text{is restricted to a set } \mathcal{D} \text{ of ``allow-} \\ \text{able'' perturbations} \end{array} \right\}$$

The distance $\nu(E,A,B)$ is related to the conditioning of a variety of computational control problems. The special case in which $E = I$ and E is not allowed to be perturbed has been extensively studied but few provably reliable methods for calculating $\nu(E,A,B)$ are known.

One natural choice of the set $\mathcal{D}$ of allowable perturbations is to restrict ΔE to preserve the null space of E. This prevents perturbations from introducing differentiated variables not already present in the unperturbed system. Alternatively, $\mathcal{D}$ may be chosen to restrict ΔE to preserve the left null space of E. This preserves the character of the algebraic constraints. In some contexts, e.g., the case $E = I$, it may be best to select $\mathcal{D}$ to restrict ΔE to be zero.

In the case that $\mathcal{D}$ is chosen to preserve either the left or right null space of the set of allowable perturbations, the problem of calculating $\nu(E,A,B)$ can be to a one real variable parameter optimization problem involving the smallest singular value of a one real parameter family of matrices.

Similar techniques can be used to calculate the distance from a stable pencil (one with all eigenvalues in the open left half plane or one with eigenvalues in the unit disc) to the nearest unstable pencil.

The problem of calculating $\nu(E,A,B)$ boils down to the problem of finding the nearest rectangular pencil with a regular part to a generic singular rectangular pencil. A related problem is that of finding the distance from a given square, regular pencil $\lambda E - A$ to the nearest singular pencil. Reasonably effective heuristic methods for this problem have been known for some time, but it appears to be quite difficult to find a failure free numerical method.

Overlapping Graph Decomposition Methods for General Sparse Linear Systems

Xiao-Chuan Cai* Youcef Saad†

Domain decomposition methods have been extensively studied for finite element problems whereby the subproblems are obtained by a partition of the underlying mesh on which the finite element problems are formulated. In this presentation, we discuss algebraic extensions of the class of overlapping domain decomposition algorithms for general sparse matrices. The subproblems are created with an overlapping partition of the adjacency graph corresponding to the sparse structure of the matrix. These algebraic domain decomposition methods are especially useful for unstructured mesh problems.

The fundamental principle underlying this extension is to replace the *domain of definition* of the problem by the *adjacency graph* of the sparse matrix, i.e., the graph that represents its non-zero pattern. We note that by switching from a domain to a graph the concept of Euclidean distance, which plays an important role in the optimality analysis of these domain decomposition methods, is lost. We show in this presentation, mostly by means of numerical experiments, that the efficiency of the overlapping methods can be preserved to some extent with certain well-balanced overlapping graph decomposition.

In the practical implementation of the algebraic Schwarz algorithms, a crucial step is the non-numerical preprocessing consisting of the graph partitioning and coloring. Many of the useful schemes in graph theory, such as the perfect graph coloring scheme, are of NP-hard. On the other hand, simple and easy-to-implement heuristics that generally perform reasonably well exist.

*Department of Mathematics, University of Kentucky, Lexington, KY 40506. cai@ms.uky.edu. Work supported in part by the National Science Foundation and the Kentucky EPSCoR Program under grant STI-9108764 and in part by AHPCRC, University of Minnesota, under Army Research Office grant number DAAL03-89-C-0038.

†Department of Computer Science, University of Minnesota, Minneapolis, MN 55455. saad@cs.umn.edu. Work supported in part by NIST under grant number 60NANB2D1272 and and in part by AHPCRC, University of Minnesota, under Army Research Office grant number DAAL03-89-C-0038.

Two Hybrid Algorithms for the Tridiagonalization of Symmetric Sparse Matrices

Ian A. Cavers
Department of Computer Science
University of British Columbia
cavers@cs.ubc.ca

January 28, 1993

This paper considers sequential, direct methods for the reduction of a sparse, symmetric matrix to tridiagonal form, using sequences of Givens similarity transformations. One possible approach to sparse tridiagonalization is the construction of customized Givens reduction algorithms that attempt to exploit matrix sparsity. The experimentation of Duff and Reid [DR75], however, shows that, even with careful use of row interchanges, adaptations of Givens reduction for use with large sparse matrices usually experience prohibitive levels of fill.

Assuming A has a symmetric permutation P^TAP with moderate bandwidth, previously the best direct tridiagonalization method consisted of two distinct phases. First a bandwidth reducing preordering algorithm, perhaps GPS [GPS76, Lew82], permutes the sparse matrix, which is then reduced with the $O(bN^2)^*$ band-preserving, column-oriented tridiagonalization algorithm of Rutishauser [Rut63] and Schwarz [Sch68]. The latter algorithm, subsequently referred to as the Rutishauser–Schwarz or R–S algorithm, is used in EISPACK's BANDR implementation [GBDM77] and is the basis of the vectorized code, SSBTRD, in LAPACK [ABB$^+$92]. We have shown [Cav92], however, that this approach is almost completely dependent upon the preordering algorithm to take maximal advantage of sparsity. Typically the selected preordering leaves the band of the permuted matrix relatively sparse. Unfortunately, even if the application of a transformation by the Rutishauser-Schwarz algorithm is enhanced to take full advantage of band sparsity, the unreduced portion of the band fills rapidly and further opportunity to exploit sparsity is lost.

In this paper we discuss two new hybrid tridiagonalization algorithms, BANDHYB and HYBSBC, which also use bandwidth reducing preorderings and band-preserving reduction techniques. Both algorithms, however, rearrange the elimination sequence of nonzero entries, taking better advantage of sparsity within the band of the permuted matrix.

We motivate our first algorithm, BANDHYB [Cav92], by the following observation. A bandwidth reducing preordering frequently produces a permuted matrix, whose band consists of varying length spikes of nonzeros extending from the main diagonal. If the ends of the longest spikes are "clipped" off, the matrix's bandwidth can be significantly reduced at relatively low cost before the band becomes full. To execute this portion of the reduction, BANDHYB uses a band-preserving sparse reduction algorithm, Bandwidth Contraction (or BC), to eliminate band nonzeros diagonal by diagonal from the outside in. Once the intermediate matrix satisfies some measure of band "fullness", the reduction process switches to the Rutishauser-Schwarz algorithm to complete the contracted band's tridiagonalization. BANDHYB has been implemented and tested on a large number of symmetric problems from the Harwell-Boeing sparse matrix collection. In comparison to an GPS-BANDR approach, the GPS-BANDHYB algorithm shows significantly reduced CPU requirements for most sparse problems. Reductions of 25-45% in CPU time are common but select problems experience reductions as high as 63%.

Our newest algorithm, HYBSBC, is similar to BANDHYB'S approach, but improves upon two aspects of the sparse reduction. First, HYBSBC takes additional advantage of band sparsity by replacing the algorithm's Bandwidth Contraction stage with a novel algorithm, Split Bandwidth Contraction (or SBC). In addition, HYBSBC improves the regulation of the reduction's transition to the R–S algorithm.

For most sparse problems, the predominant cost of Bandwidth Contraction is bulge[†] chasing. In recognition of this fact, the Split Bandwidth Contraction algorithm modifies the elimination sequence of a diagonal's nonzeros to reduce the number of bulge chasing transformations required. Rather than eliminate a diagonal's nonzeros from top to bottom, Split Bandwidth Contraction begins by identifying a *split point*. Starting just above the split point, elimination proceeds back up the diagonal, while below the split point, elimination proceeds away from the split point, towards the end of the matrix. If needed, shortened bulge chasing transformation sequences, in the appropriate direction, accompany a band nonzero's elimination. Bulges produced by eliminations above the split point are chased off the top of the matrix, while bulges originating from other eliminations are chased off the bottom of the matrix.

A split point is a block of one or more zero entries in the diagonal under reduction. The zero entries completely isolate the transformations applied above and below the split point, permitting the elimination sequence to start in the middle of the diagonal. Of course, this type of reduction is not possible if the diagonal is dense. When the diagonal contains a single, centered zero entry, the split point is fixed. In this case the shortened, bidirectional bulge chasing sequences reduce the computational requirements of Split Bandwidth Contraction to approximately 1/2 the cost of Bandwidth Contraction. However, when split points are forced off center or several split points are available, we require strategies for proper split point selection and the regulation of the algorithm's transition to R–S .

For general sparse matrices, the optimal split point may not be immediately obvious. We present rationale motivating four split point selection strategies. Using experimental evidence, we demonstrate that a so-called minimum displacement split point selection strategy with damped tiebreaking is preferable.

When HYBSBC is applied to a sparsely banded matrix, we show that it is cost effective for the SBC stage to continue while a split point exists near the middle of the outermost nonzero diagonal. Unfortunately, as a sparse tridiagonalization proceeds, the unreduced portion of the band typically becomes more dense, and the best split points are forced towards either end of the diagonal. As a result, at some point allowing one extra step of SBC, using an off-center split point, and then switching to R–S may become more costly than switching to R–S immediately. The algorithm's transition strategy must resolve this cost comparison before the contraction of each subdiagonal by SBC.

The *delta* transition strategy developed for HYBSBC uses operation counts for the SBC and R–S algorithms to construct a general algebraic function that estimates the cost of delaying the transition. While the cost of one extra step of SBC followed by R–S is estimated to be less than immediately switching to R–S , the algorithm continues to contract the bandwidth

[*] The bandwidth, b, of the permuted matrix is defined as $\max_{i,j\in\{1\ldots N\},i\neq j}|i-j|$ such that $(PAP^T)_{ij}\neq 0$.

[†] A *bulge* is a nonzero entry, created by the application of a transformation, lying outside the current bandwidth.

with SBC . For practical sparse problems, we show that the transition bandwidths selected by the delta transition strategy are close to optimal and that the cost of evaluating the delta function is minimal.

In conclusion, we describe extensive testing of the HYBSBC algorithm with symmetric, sparse problems from the Harwell-Boeing test matrix collection. In comparison to GPS-BANDHYB, the GPS-HYBSBC algorithm reduces CPU time by 20-40% for many problems, and by more than 50% for select problems. These additional gains make GPS-HYBSBC a very impressive alternative to GPS-BANDR, with one problem tridiagonalized in approximately 1/5 of BANDR's time.

References

[ABB+92] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen. *LAPACK Users' Guide*. SIAM, Philadelphia, 1992.

[Cav92] Ian A. Cavers. A hybrid tridiagonalization algorithm for symmetric sparse matrices. Submitted to SIAM Journal on Matrix Analysis and Applications, March 1992. First revision, November 1992.

[DR75] I. S. Duff and J. K. Reid. On the reduction of sparse matrices to condensed forms by similarity transformations. *J. Inst. of Maths. Applics.*, 15:217–224, 1975.

[GBDM77] B. S. Garbow, J. M. Boyle, J. J. Dongarra, and C. B. Moler. *Matrix Eigensystem Routines - EISPACK Guide Extension*, volume 51 of *Lecture Notes in Computer Science*. Springer-Verlag, 1977.

[GPS76] N. E. Gibbs, W. G. Poole Jr., and P. K. Stockmeyer. An algorithm for reducing the bandwidth and profile of a sparse matrix. *SIAM J. Numer. Anal.*, 13(2):236–250, 1976.

[Lew82] J. G. Lewis. Implementation of the Gibbs-Poole-Stockmeyer and Gibbs-King algorithms. *ACM TOMS*, 8(2):180–189, 1982.

[Rut63] H. Rutishauser. On Jacobi rotation patterns. In *Experimental Arithmetic, High Speed Computing and Mathematics*, volume 15 of *Proceedings of Symposia in Applied Mathematics*, pages 219–239. AMS, April 1963.

[Sch68] H. R. Schwarz. Tridiagonalization of a symmetric band matrix. In J. H. Wilkinson and C. Reinsch, editors, *Linear Algebra*, volume II of *Handbook for Automatic Computation*, pages 273–283. Springer-Verlag, 1968.

DOMAIN DECOMPOSITION INTERFACE PRECONDITIONERS FOR COUPLED PDE SYSTEM
SEPTEMBER 8,1992

TONY F. CHAN AND JIANPING SHAO *

Abstract. Domain decomposition technique has been efficiently used to design numerical solver for elliptic problems on irregular domains and on the multiprocessor computer. Typically, a domain is decomposed into many smaller regular nonoverlapping subdomains. The capacitance system on the interface is usually solved by preconditioned conjugate gradient method. In this paper, by using Fourier approximation and probe technique, we propose several interface preconditioners for the model problems, the coupled elliptic system which comes from the linearization of semiconductor device simulation. We compare the convergence behaviors of these preconditioners in the performance of preconditioned iterative method through providing numerical results and their eigenvalue distribution. We found that the change of coupling parameter greatly affects the properties of the coupled discrete system. This brings difficulty in constructing efficient preconditioner for the coupled system. Numerical results are presented.

Key Words. domain decomposition, capacitance matrix, the coupled elliptic system, Fourier approximation, probe technique.

AMS subject classifications: 65N20, 65F10.

1. Introduction . Domain decomposition is a class of techniques used to design efficient algorithms for elliptic problems on an irregular domain and multiprocessor systems. The basic idea is to decompose a domain into many smaller regular subdomains and obtain global solution through iteratively solving subproblems on these subdomains. This is relatively old idea which can be traced to Schwarz's alternating procedure [?].

There are several reasons why such a procedure is attractive and useful. The first is its obvious advantage in implementation on multiprocessor systems. The second is its flexibility to the feature of problems, such as irregular domain, discontinuous coefficients, singular problem, and boundary layer,etc. The domain decomposition technique has been successfully applied to many problems, see for instance [?, ?, ?, ?].

In this paper, we consider applying the non-overlapping domain decomposition to the coupled elliptic systems which result from linearization of semiconduct device simulation problems [?, ?, ?, ?]. The main idea is to decompose a domain into many smaller regular subdomains, reduce the problem on the whole domain to the capacitance system on the interface and then solve the interface system. Since the capacitance matrix is expensive to evaluate and to solve directly, the reduced capacitance system is usually solved by iterative method, such as GMRES or BiCG conjugate gradient type method. To minimize the number of iteration, it is imperative to have a good precondtioner for the capacitance matrix. Our main purpose, here, is to derive interface preconditioners for the coupled elliptic system. We notice that the variation of coupling term may lead the coupled elliptic problem to become indefinite and unsymmetric. Hence, our main interest is in finding the preconditioner that slightly depends on or does not depend on the coupling parameter. Our efforts on this are based on two approaches. One is Fourier approximation. The other is probe technique.

Fourier transform has been used to derive exact eigen-decomposition or triangle similar matrix of the capacitance matrix for the simple model coupled elliptic system on the rectangle with constant coefficients. This makes it possible to construct preconditioners, which are similar to those proposed for Poisson equation by Dryja [?], Golub and Mayer [?], and Chan [?] respectively. One advantage of this approach is that the mathematical setting is extremely simple. These preconditioners may be used for variable coefficient coupled elliptic system on general domain through averaging coefficients and approximating the irregular domains by regular domain sharing the same interface. However, this generalization may be still sensitive to the variations in these other parameters.

Probe technique, which was developed by Chan and Resasco[?], Keyes and Gropp [?, ?], and Chan and Mathew [?], is used here to construct interface preconditioner for the coupled elliptic problem. Since probe technique is an algebraic method, it can easily be applied to any operator on any domain, provided having decay properties. We notice that the entries of the capacitance matrix decay rapidly away from its diagonal when the coupling parameter is very small. However, this decay property is very sensitive to the coupling parameter. In order to overcome this difficulty, we use Fourier transform to concentrate the off diagonal entries to the diagonal entries, and then use probe technique to approximately find these entries. This combination of Fourier transform and probe method will be refereed as Fourier-probe technique. Probe technique has been successfully applied to construct preconditioners to 4th order problems, to the Navier-Stokes [?, ?], to convection-diffusion problem [?], and etc..

This paper is organized as follows. In section 2, we describe the model coupled elliptic problem and give the capacitance matrix on the interface. In section 3, we derive the spectral decomposition of capacitance matrix and construct interface preconditioners by using Fourier approximation. Then, we discuss how to construct preconditioners by using probe technique in section 4. Finally, we present numerical results for these preconditioners and summarize the main properties.

* Department of Mathematics, University of California at Los Angeles, Los Angeles, CA. 90024

A Composite Step Conjugate Gradients Squared Algorithm for Solving Nonsymmetric Linear Systems

Tony F. Chan Tedd Szeto

Recently, the Composite Step Biconjugate Gradient method (CSBCG) was proposed by Bank and Chan [1, 2] to cure one type of breakdown inherent in the BCG algorithm. Specifically, CSBCG skips over steps for which the BCG iterate is not defined due to the singularity of the principal submatrix of the tridiagonal matrix generated by the underlying Lanczos process. We propose a method, the Composite Step Conjugate Gradients Squared Algorithm (CSCGS), which uses a similar technique to smooth the convergence of the CGS algorithm [4]. By doing this, we obtain a method which not only handles the breakdowns described above, but does so without involving multiplications by the transpose matrix and has a faster convergence rate. We further investigate smoother convergence by applying a residual smoothing technique as described by Schönauer [3] and Weiss [5]. We also prove a "best approximation" result for the method. Finally, numerical experiments are shown illustrating the practical performance of the method.

References

[1] R. Bank and T. Chan, "A Composite Step Bi-Conjugate Gradient Algorithm for Nonsymmetric Linear Systems," presented at the Fourth SIAM Conf. in Appl. Lin. Alg., Minneapolis, 1992.

[2] R. Bank and T. Chan, "An Analysis of the Composite Step Bi-Conjugate Gradient Method," presented at the Fourth SIAM Conf. in Appl. Lin. Alg., Minneapolis, 1992.

[3] W. Schönauer, *Scientific Computing on Vector Computers*, North-Holland, Amsterdam, New York, Oxford, Tokyo, 1987.

[4] P. Sonneveld, "CGS: a Fast Lanczos-type Solver for Nonsymmetric Linear Systems," SIAM J. Sci. Stat. Comput., v. 10, 1989.

[5] R. Weiss, "Convergence Behavior of Generalized Conjugate Gradient Methods," Ph.D. thesis, University of Karlsruhe, 1990.

Finite Precision Analysis of Inverse Iteration

S. Chandrasekaran*

We prove that inverse iteration can efficiently compute eigenvectors for a real symmetric matrix so that they are numerically orthogonal and their residue is small – even in the presence of pathologically close eigenvalues.

Given some set of computed eigenvalues of a real symmetric matrix, inverse iteration is the most efficient way of computing the corresponding set of eigenvectors. The method has plagued numerical analysts for some time as it involves the solution of highly ill-conditioned linear systems at each stage. Wilkinson was able to give convincing arguments to show that the method computes "good" eigenvectors in finite precision for distinct eigenvalues. But clustered eigenvalues were not amenable to his arguments leading him to remark in his book "The Algebraic Eigenvalue Problem" that "The problem of determining reliably full digital information in the subspace spanned by eigenvectors corresponding to coincident or pathologically close eigenvalues has never been satisfactorily solved.". In this talk we provide a satisfactory solution to Wilkinson's problem.

In the process we obtain rigorous criteria for stopping and for orthogonalisation. These criteria can be implemented by making simple modifications to existing implementations of inverse iteration in LAPACK. In addition we reduce the "average-case" time complexity of the method for symmetric tri-diagonal matrices of order n from $O(n^3)$ to $O(n^2)$, and also reduce the constant in front of the leading term in the time complexity by half.

We briefly mention the highlights of the analysis now. We assume that the eigenvalues are sufficiently accurate and that the eigenvectors are being computed from the smallest to the largest. We analyse the computation of the ith eigenvector by first assuming that no orthogonalisation is necessary. Traditionally the analysis was either carried out in the canonical basis or in the actual eigenvector basis. Instead we use the set of orthonormal columns nearest to the previously computed set of $i-1$ eigenvectors as the basis in which to carry out the analysis. Moreover we don't try to bound the norm of the error (as is traditional) but rather we bound the error in individual components in the new basis. The chief result we use here is that the error along an eigenvector is inversely proportional to the magnitude of the corresponding eigenvalue. We then show the worst sequence of iterates comes from the power iteration of an unsymmetric matrix. This enables us to prove that inverse iteration works in finite precision when no orthogonalisation is needed.

In case of orthogonalisation we divide the eigenvalues into several groups. As Wilkinson had already guessed it is necessary to perturb the computed ith eigenvalue in order to separate it from the $(i-1)$st eigenvalue. We present criteria to divide the eigenvalues into pathologically close eigenvalues, moderately close eigenvalues and far away eigenvalues. The eigenvectors corresponding to pathologically close eigenvalues are treated as a subspace. After each iteration the iterate is orthogonalised against all previously computed eigenvectors in the pathological space (in the current LAPACK implementation the iterate is orthogonalised even against those eigenvectors which are just moderately close). Our definition of the pathological space makes it possible to incorporate the orthogonalisation errors so that we can define again the sequence of worst iterates through the power iteration of an unsymmetric matrix. Upon convergence it is sufficient to orthogonalise the iterate once against those eigenvectors which are moderately close.

With the benefit of hindsight we see that the actual implementation of inverse iteration can be derived from purely infinite-precision considerations!

* Joint work with I. Ipsen.

MULTIFRONTAL SPARSE SOLVERS IN MESSAGE PASSING AND DATA PARALLEL PROGRAMMING ENVIRONMENTS: A COMPARATIVE STUDY.

JOHN M. CONROY, STEVEN G. KRATZER, AND ROBERT F. LUCAS

Supercomputing Research Center
Institute for Defense Analyses
Bowie, MD 20715-4300

January 1993

For many scientific applications that require sparse linear equation solution, the most reliable and flexible methods are those based on direct matrix factorization. Due to the high potential performance of many parallel machines, an efficient solution on such architectures would be highly desirable. Two types of programming environments have been developed for distributed-memory parallel machines – message passing and data parallel. Each provides a different level of control on the execution of a parallel program. This talk will describe work done on the CM-5, a machine that supports both programming models, to develop a set high performance sparse matrix routines.

The CM-5 architecture consists of k nodes each of which has four vector units. Each node is controlled by a sparc processor which both issues instructions to the vector units and handles communication between nodes. From a data parallel model the machine is most naturally viewed as $4k$ processors running synchronously, while the message passing model dictates viewing the machine as a collection of nodes, with the four VU's in each node acting as an ensemble. In both programming models the dense matrix kernels on each node have been coded to run at near optimal efficiency by using standard vectorization techniques.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

The elimination tree of a sparse matrix under a given ordering provides a precedence graph of a series of dense matrix problems to be solved. The factorization and solution exhibit two levels of parallelism – within each dense matrix operation and across independent matrix operations. Our data parallel approach exploits the former type of parallelism by mapping each dense problem (frontal matrix) to a square array of processors via a block torus wrapping. When such a mapping is applied to the overall sparse matrix, all communication between processors is performed during the factorization and solution of the dense frontal matrices. Assembly of frontal matrices and stack manipulation are local to each processor. For the model problem of a N by N grid this "parsing" of the elimination tree gives up to $O(N)$ parallelism with high efficiency.

In addition to the concurrency within each frontal matrix, our message passing approach also exploits the second form of parallelism, by factoring multiple frontal matrices concurrently in disjoint subsets of the processors. Recursively partitioning the matrix in this fashion leads to a subtree-subcube or nested mapping. If this process is followed to its logical conclusion, each processor will have a unique branch of the elimination tree. Within frontal matrices, communication is limited to broadcasts of pivot rows and columns. As frontal matrices that are factored in parallel are assembled, aggregate updates are sent among the processors.

To date we have scalar (Sparc only) implementations of complete sparse factorization programs for both the message passing and data parallel programming models. We are now integrating our vectorized/parallelized dense matrix kernels into the parallel sparse solvers. We will be able to present and contrast results of optimized implementations of both at the meeting in June.

Domain Decomposition Methods for Ill–conditioned Elliptic Problems with Applications to Semiconductor Device Modelling

Rob Coomer
School of Mathematical Sciences
University of Bath
Bath
BA2 7AY
England

Abstract

Stationary semiconducting devices are classically modelled by drift–diffusion equations. Employing the well-known change of variables to the quasi–Fermi potentials, these equations may be written as the following system of coupled nonlinear elliptic partial differential equations

$$\begin{aligned} -\lambda^2\Delta\psi + \delta(\exp(\psi - v) - (w - \psi)) - d &= 0, \\ -\nabla.(\exp(\psi - v)\nabla v) &= \rho_v r(w - v, \psi - v, w - \psi), \\ -\nabla.(\exp(w - \psi)\nabla w) &= -\rho_w r(w - v, \psi - v, w - \psi). \end{aligned}$$

These are to be solved for ψ, the electrostatic potential, and v and w, the electron and hole quasi–Fermi potentials respectively. The equations are solved on some generally polygonal domain in $\boldsymbol{R}^2$ subject to mixed boundary conditions. In many models λ, δ, ρ_v and ρ_w are constants, d is the (piecewise smooth) doping profile and r is some, generally nonlinear, model for the recombination/generation rate.

A typical solution strategy involves decoupling the three equations with a "Gummel" type iteration. That is, given a starting guess (ψ^0, v^0, w^0) we iterate around the following loop

$$\begin{aligned} -\lambda^2\Delta\psi^{k+1} + \delta(\exp(\psi^{k+1} - v^k) - (w^k - \psi^{k+1})) - d = 0, \\ -\nabla.(\exp(\psi^{k+1} - v^k)\nabla v^{k+1}) = \rho_v r(w^k - v^k, \psi^{k+1} - v^k, w^k - \psi^{k+1}), \\ -\nabla.(\exp(w^k - \psi^{k+1})\nabla w^{k+1}) = -\rho_w r(w^k - v^k, \psi^{k+1} - v^k, w^k - \psi^{k+1}). \end{aligned}$$

These partial differential equations are solved by piecewise linear finite elements on triangles. The discretization of the first equation (the potential equation) yields a nonlinear system for the approximation to ψ. We construct upper and lower solutions to this system and then use a parallel quasi–Newton method for its solution. We have shown that, under sufficient conditions, the iterates of this algorithm converge monotonically and quadratically to the unique solution. It is also known that the overall decoupling algorithm converges for sufficiently small voltage biases.

We are therefore interested in efficient methods for solving the large linear systems which arise from this decoupling strategy. Those associated with the potential equation are (relatively) well–conditioned in comparison to those arising from the second and third equations (the electron and hole continuity equations respectively). Thus in this work we focus on problems of the form

$$-\nabla.(a\nabla u) = f \text{ on } \Omega, \tag{0.1}$$

with appropriate boundary conditions and where a is piecewise smooth on the domain Ω but may take widely differing values from subregion to subregion. As shown by Markowich, the exponential coefficients in the electron and hole continuity equations typically have this property with jumps which may be $O(10^7)$ across interior layers.

We consider domain decomposition methods for (0.1) and their implementation on a 1K MasPar parallel architecture. The domain is first divided into quadrilateral substructures which are then further subdivided to obtain a fine triangulation on which (0.1) is discretized by finite elements. The resulting linear system is solved by locally eliminating unknowns associated with the interior nodes of the substructures and solving the remaining Schur complement system for the substructure boundary unknowns by a preconditioned conjugate gradient method (PCGM).

The preconditioner in its most general form is the sum of local block diagonal approximations to the Schur complement (involving unknowns along edges of substructures and/or unknowns local to a vertex of the coarse grid) plus a coarse grid operator which is needed to simulate the global interaction of the substructures.

Widlund and Dryja have shown that if a preconditioner consisting of the vertex spaces (with an overlap proportional to the coarse mesh diameter) together with the coarse grid operator is used, then the convergence rate of the PCGM is independent of the number of substructures and the mesh diameter of the triangulation. They have also shown that if the vertex space approximations are replaced with edge space approximations that have no overlap, then the convergence rate grows logarithmically with the ratio of coarse mesh to fine mesh diameters. Furthermore Smith has shown that the convergence of the latter strategy is independent of the jumps in the value of a across substructure boundaries.

We have so far implemented the optimal method of Widlund and Dryja on a 1K MasPar MP-1 parallel computer. This is a SIMD (Single Instruction Multiple Data) machine with a square array of 1024 processing elements each of which has 16KByte RAM. The architecture of the machine is such that to find and explicitly invert the preconditioners would be prohibitively expensive, so the preconditioning solves are done by parallel inner conjugate gradient loops. We have results obtained from various test problems which reflect the theoretical results of Widlund and Dryja (even though the preconditioning solves are inexact) and which are comparable with various other methods.

In the coming months we hope to enhance the speed of execution of our code by improving the coarse grid solver, which is proving to be expensive at present. We also hope to implement the version using edge space rather than vertex space approximation to obtain convergence independent of the jumps in a. We then hope to try a mixture of edge and vertex space preconditioners along with the coarse grid preconditioner in an attempt to obtain convergence independent of subdomains, triangulation and jumps in a. We feel that a method of this type would be well suited to our semiconductor application where a is effectively constant on large regions of the domain with very sharp fronts between these regions.

High Performance Computing in Linear Control

Biswa Nath Datta
Department of Mathematical Sciences
Northern Illinois University
DeKalb, IL 60115
e-mail: dattab@math.niu.edu

Remarkable progress has been made in both theory and applications of all important areas of control. The theory is rich and very sophisticated. Some beautiful applications of control theory are presently being made in aerospace, biomedical engineering, industrial engineering, robotics, economics, power systems, etc.

Unfortunately, the same assessment of progress does not hold in general for computations in control theory. Control theory is lagging behind other areas of science and engineering in this respect. Nowadays there is a revolution going on in the world of high performance scientific computing. Many powerful computers with vector and parallel processing have been built and have been available in recent years. These supercomputers offer very high speed in computations. Highly efficient software, based on powerful algorithms, has been developed to use on these advanced computers, and has also contributed to increased performance. While workers in many areas of science and engineering have taken great advantage of these hardware and software developments, control scientists and engineers, unfortunately, have not been able to take much advantage of these developments.

Progress in computational aspects of control theory, especially in the area of large-scale and parallel computations, has been painfully slow. On the other hand, there are practical situations—such as the design of large space structures, control of power systems, and others—that give rise to very large problems, some of which are so large that they can be considered **grand challenge problems.**

The need for expanded research in these areas has been clearly outlined in the recent panel report "Future Directions in Control Theory: A Mathematical Perspective." The control community has been urged to collaborate with numerical analysts, software engineers, and experts in large-scale, parallel, and symbolic computations to develop interdisciplinary projects for computer solutions of real-time control, complex control systems, intelligent control, stochastic control, nonlinear filtering, and other control problems.

In the last few years, the author, in collaboration with some well-known experts on large-scale and parallel computations (Chris Bischof of Argonne National Laboratory and Youcef Saad of the University of Minnesota, as well as several of the latter's Ph.D. students), has developed a few computationally viable parallel algorithms and algorithms for large-scale computations for important linear algebra problems arising in control [?], [?]–[?]. Among these problems are those of controllability, the eigenvalue assignment problem, design of observers, and matrix equations, to name a few. These algorithms have been implemented on some of the existing parallel architectures and (nearly) perfect speed and speed-up have been achieved in all cases. It is believed that these algorithms will be highly beneficial to practicing control engineers and will provide incentive for expanded research in this area.

In this paper, we will present an overview of these and other existing parallel algorithms for linear control problems, giving particular attention to block algorithms for high performance computing. The results of performance of some of these algorithms on existing parallel/vector architectures such as the ALLIANT FX/8, the CRAY YMP, Intel ipsc and others will also be presented. The lecture will conclude with remarks on **"Future Directions of Research on High Performance Computing for Control."**

References

[1] Arnold, M. and B. N. Datta, *An algorithm for the multi-input eigenvalue assignment problem*, IEEE Trans. Automatic Control, vol. 35 (1990), 1149-1152.

[2] Balas, M. J., *Trends in large space structure control theory: Fondest dreams, wildest hopes*, IEEE Trans. Auto. Control, AC-2 (1982), 522-535.

[3] Bischof, C., Datta, B. N., and A. Purkayastha, *A parallel algorithm for the multi-input Sylvester-observer equation*, Argonne Technical Report MCS-P2741-1191 (1991), 1-15. To appear in SIAM J. Scientific and Statistical Computing.

[4] Datta, B. N. and Y. Saad, *Arnoldi methods for large Sylvester-like observer matrix equation and an associated algorithm for partial spectrum assignment*, Lin." Alg." Appl., Vol." 154-156 (1991), 225-244.

[5] Datta, B. N. and Karabi Datta, *Efficient parallel algorithms for controllability and eigenvalue assignment problems*, Proceedings of the 25th IEEE Conference on Decision and Control, Athens, Greece, (1986), 1611-1616.

[6] Datta, B. N. and Karabi Datta, *Sequential and parallel computations and complexities for determining relative primeness, stability and inertia*, Contemporary Mathematics, Vol." 47 (1985), 95-109. (Special issue on *Linear Algebra and its Role in Systems Theory*.)

[7] Datta, B. N., *Parallel algorithms in control theory*, an **invited paper** in the Proceedings of the IEEE Conference on Decision and Control, (1991), 1700-1704.

[8] Datta, B. N. and F. Rincon, *On global feedback stabilization of large second-order models*, Proceedings of the IEEE Conference on Decision and Control, 1990 and in Transactions of the Society for Computer Simulation (1991), 99-108.

[9] Datta, B. N. and Y. Saad, *Numerical Solutions of Some Linear Algebra Problems in Control*, A talk at the Invited Session on Numerical Linear Algebra in Signals, Systems and Control, SIAM Conference on Linear Algebra in Signals, Systems and Control, Boston, August 1986.

[10] Datta, B. N., *Large-Scale and Parallel Matrix Computations in Control: A Tutorial* (invited **paper**), Proceedings of the American Control Conference, June 1992.

An extended Kaczmarz's method for ℓ_p minimum norm solutions

by

Achiya Dax

June, 1992

Hydrological Service
P.O.B. 6381
Jerusalem 91060
ISRAEL

Abstract. This paper presents a row relaxation method for solving the problem

$$\begin{aligned} &\text{minimize} \quad \|\underline{x}\|_p^p/p \\ &\text{subject to} \quad A\underline{x} = \underline{b} \end{aligned}$$

where $1 < p < \infty$. It is shown that the dual of this problem has the form

$$\text{maximize} \quad \underline{b}^T\underline{y} - \|A^T\underline{y}\|_q^q/q$$

where $q = p/(p-1)$. Moreover, let $\underline{y}$ solve the dual and let $\underline{x} = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$ solve the primal, then $x_j = |\underline{c}_j^T\underline{y}|^{q-1}\text{sign}(\underline{c}_j^T\underline{y})$ where $\underline{c}_j$ denotes the j-th column of A. That is, a primal solution is easily retrieved from a dual one. Maximizing the dual objective function by changing one variable at a time results in an iterative scheme that resembles Kaczmarz's method. This feature makes the new scheme suitable for problems in which A is large, sparse and unstructured. Numerical experiments illustrate the ability of the proposed method to handle very large problems of this kind.

Key words: ℓ_p minimum norm solutions, Large sparse problems, Duality relations, Row relaxation methods.

Numerical Methods for Optimal Control Problems

June M. Donato*

January 26, 1993

Here we consider a two-sided game for a non-local competitive system where the control is on the source terms. The two original equations give rise to two more adjoint equations. Thus, we are led to solve four coupled non-linear parabolic partial differential equations in four variables.

First, we consider the numerical solution of two-dimensional elliptic versions of the coupled systems. Several methods for solving these nonlinear systems are utilized. Of the methods tested we chose the point Gauss-Seidel Newton iteration for the further study of system. Parameters within the system are varied and the behavior of the solutions is compared against theory.

Next, we consider the parabolic versions of the systems. The difficulty becomes one of handling the time steps. This is made difficult in that the two original equations are specified forward in time (initial conditions are given), whereas the adjoint equations are specified backward in time (final time conditions are given). We again investigate a variety of techniques in solving these systems, including a method based on the concept of multigrid. But here, we use a multigrid-like method in time, not in space.

This work was done in conjunction with Dr. Lenhart (University of Tennessee, Knoxville) and Dr. Protoposescu (Oak Ridge National Laboratory, Tennessee).

*Oak Ridge National Laboratory, Tennessee

Solution of symmetric indefinite sparse linear equations

Iain S Duff

Rutherford Appleton Laboratory and CERFACS

Abstract

The solution of sparse symmetric indefinite equations is a common subproblem of many numerical calculations, particularly in constrained optimization problems, for example the KKT equations or subproblems in the solution of linear programs by interior point methods. The resulting linear system is of the form

$$\begin{matrix} H & A^T \\ A & 0 \end{matrix}$$

where the matrix H is symmetric but is sometimes non-definite.

For some years we have had a code in the Harwell Subroutine Library to solve sparse symmetric indefinite problems. The code, called MA27, uses a modified form of the Bunch-Parlett-Kaufman pivoting algorithm to maintain stability and exploits sparsity by using a multifrontal approach.

This code does not work well on structured problems of the form shown above. We discuss why this is the case, and examine design changes to improve the performance of this code on problems with zeros on the diagonal. We illustrate the effect of our changes and discuss the design of our software that implements these new techniques.

We also consider some other methods for solving the augmented system shown above, including iterative methods that use eigenvalue approximations to accelerate convergence.

A Supernodal Approach to a Sparse Partial Pivoting Code

Stanley C. Eisenstat, John R. Gilbert, and Joseph W.-H. Liu

Abstract

The problem of solving sparse symmetric positive definite systems of linear equations on sequential and vector processors seems to be relatively well understood. Normally the solution process is broken into two phases:

1. symbolic factorization to determine the nonzero structure of the Cholesky factor

2. numeric factorization and solution.

The use of elimination trees and compressed subscripts has reduced the time and space for the symbolic factorization to a low order term. The use of supernodal elimination (or multifrontal elimination) has allowed the use of dense vector operations for nearly all of the floating-point computation, thus reducing the symbolic overhead in the numerical factorization to a low order term.

For unsymmetric systems where pivoting is required in order to maintain numerical stability, the progress has been less satisfactory.

Recently, Eisenstat and Liu showed how to exploit structural symmetry to decrease the amount of structural information required for the symbolic factorization of a sparse unsymmetric matrix (i.e., for obtaining the nonzero structures of the factor matrices). They then showed how to use this technique of symmetric reduction to improve the performance of a class of partial pivoting codes for the LU factorization of large sparse unsymmetric matrices. The result was that the time and space for symbolic factorization was reduced to a low order term (the resulting speedup was more than a factor of two for some problems).

In this talk we will describe how a similar approach can be used to generalize supernodes to the unsymmetric case. Preliminary results suggest that the total time can be as much as 25% less than the time to factor the matrix *given the sequence of pivots and the symbolic factorization of the reordered matrix.*

Fast Numerical Solution of the Radiative Helmholtz Equation by Imbedding

Oliver Ernst

The *Helmholtz* or *reduced wave equation* describes the spatial part of wave phenomena, when constant propagation speed and harmonic time dependence are assumed. In many physical applications, known as scattering problems, the wave field resulting from the interaction of a physical body, known as the scatterer, with some known incident wave is sought. This requires solving the Helmholtz equation

$$-\Delta u - k^2 u = f,$$

on an unbounded domain $\mathcal{D}$. Here, $k \in I\!R$ is the wave number and f is a source term. A unique solution to this equation is singled out by imposing the asymptotic condition

$$\lim_{r\to\infty} r^{\frac{d-1}{2}}(u_r - iku) = 0$$

known as the Sommerfeld radiation condition, r denoting the radial coordinate and d the dimension of the underlying space. Physically, this asymptotic condition determines whether the solution will be an incoming or outgoing wave.

When the infinite domain is truncated for numerical computation of the solution, this introduces a new, artificial boundary $\mathcal{B}$ on the computational domain. It then becomes very important to accurately incorporate the radiation condition into the finite-domain problem. This has recently been solved by Keller and Givoli[3], using the so-called *Dirichlet-to-Neumann* (DtN) mapping, which works whenever the artificial boundary is a circle or sphere in two or three dimensional domains respectively. This method has the advantage that the solution of the problem formulated with the DtN boundary condition has as its solution the restriction of the solution of the unbounded-domain problem to the computational domain, which is as good as one can do.

In this talk, we will describe the linear system of equations that arises when the Helmholtz equation is discretized on a truncated domain using finite differences and how the DtN formulation fits in well with classical fast solvers based on the fast Fourier transform and cyclic reduction techniques (cf. [1]). These techniques have a serial complexity of $O(n \log n)$, where n is the number of gridpoints of the discretization. To apply these fast techniques, it is necessary for the underlying domain to be separable, i.e. that the boundary consist of coordinate surfaces. For the circular or spherical artificial boundaries required by the DtN boundary condition, this is achieved by using polar coordinates. Direct application of fast solvers would then work only for circular scattering bodies centered at the origin. In order to extend the method to arbitrarily shaped scattering bodies, we use a computational technique known as the *capacitance matrix method* (see e.g. [2, 4]). This is an imbedding method, in which the linear system of equations resulting from the discretization is written as a low-rank modification of the same problem on a separable domain (in this case an annulus), for which the fast algorithm is available. The capacitance matrix method then requires applying the fast solver once and, in addition, the solution of a dense linear system of dimension p, the capacitance matrix equation. Here, p denotes the number of gridpoints in the domain for which at least one of its neighbors in the difference stencil used to approximate the Laplacian fails to lie inside the domain.

In its original version [2], the approach was a purely algebraic one and the capacitance matrix equation was based on the Woodbury formula for the inverse of a rank-p modification of a matrix. In [4], another appoach, which mimics discretely the integral equations of potential theory for solving elliptic boundary value problems, is used. The main idea is to view the fast solver applied to a problem on a larger (possibly infinite) domain as acting like a discrete free-space Green's function, which can then be used to incorporate the boundary condition imposed on the smaller domain. This yields well-conditioned capacitance matrices for a larger range of boundary conditions than does the purely algebraic approch. While the work in [4] was mainly concerned with positive definite operators, we demonstrate that these techniques also work well in the case of the indefinite Helmholtz operator.

References

[1] B. Buzbee, G. Golub, and C. Nielson. On direct methods for solving Poisson's equation. *SIAM J. Numer. Analysis*, 7:627–656, 1970.

[2] B.L. Buzbee, F.W. Dorr, J.A. George, and G.H. Golub. The direct solution of the discrete Poisson equation on irregular regions. *SIAM J. Numer. Anal.*, 8:722–736, 1974.

[3] J. B. Keller and D. Givoli. Exact non-reflecting boundary conditions. *J. Comput. Phys.*, 82:172–192, 1989.

[4] W. Proskurowski and O. B. Widlund. On the numerical solution of Helmholtz' equation by the capacitance matrix method. *Math. Comp.*, 30:433–468, 1976.

On Numerical Methods for Unitary Inverse Eigenvalue Problems

Heike Faßbender
Department of Mathematics and Computer Science
University of Bremen
Postfach 330 440
2800 Bremen 33
Federal Republic of Germany
heike@mathematik.uni-bremen.de

KEYWORDS. inverse eigenvalue problem, unitary matrix, Szegő polynomial, orthogonal rational function, discrete least squares approximation.

A number of signal processing problems can be seen to require numerical methods for different unitary eigenvalue problems.

One of these problems is the discrete least-square approximation of a real-valued function f given at arbitrary distinct nodes $\{\theta_k\}_{k=1}^{m}$ in $[0,2\pi)$ by trigonometric polynomials t in the discrete norm $\|f-t\| = (\sum_{k=1}^{m} |f(\theta_k)-t(\theta_k)|^2\omega_k^2)^{\frac{1}{2}}$, where the $\{\omega_k^2\}_{k=1}^{m}$ are positive weights. The problem can easily be reformulated as the standard least square problem of minimizing $DAc - Dg$ over all coefficient vectors c in the Euclidian norm, where $D = diag(\omega_1, ..., \omega_m)$ and A is the transposed $m \times n$ Vandermonde matrix

$$A = \begin{pmatrix} 1 & z_1 & z_1^2 & \cdots & z_1^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & z_m & z_m^2 & \cdots & z_m^{n-1} \end{pmatrix}$$

with $z_k = exp(i\theta_k)$.

The usual way to solve this least square problem is to compute the QR decomposition of DA. But DA is just the Krylov matrix $K(\Lambda, q_0, n) = [q_0, \Lambda q_0, ..., \Lambda^{n-1}q_0]$ (where $\Lambda = diag(z_1, ..., z_m)$ and $q_0 = (\omega_1, ..., \omega_m)^T$). We may therefore use the following consequence of the Implicit Q Theorem to compute the desired QR decomposition. If there exists a unitary matrix U such that $U^H \Lambda U = H$ is a unitary upper Hessenberg matrix with positive subdiagonal elements, then the unique QR decomposition of $K(\Lambda, q_0, m)$ is given by UR with $R = K(H, e_1, m)$. The construction of such a unitary Hessenberg matrix from spectral data, here contained in Λ, is an inverse eigenproblem. Thus the best trigonometric approximation to f can be computed via solving this inverse eigenproblem.

A different approach is to reformulate the approximation problem as the standard least square problem of minimizing $D\tilde{A}\tilde{t} - D\tilde{f}$ over all coefficient vectors $\tilde{t}$ in the Euclidian norm, where

$$\tilde{A} = \begin{pmatrix} 1 & \sin\theta_1 & \cos\theta_1 & \cdots & \sin l\theta_1 & \cos l\theta_1 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 1 & \sin\theta_m & \cos\theta_m & \cdots & \sin l\theta_m & \cos l\theta_m \end{pmatrix}$$

and l is the degree of the desired trigonometric polynomial. $D\tilde{A}$ is the product of the modified Krylov matrix $\kappa(\Lambda, q_0, l) = [q_0, \Lambda q_0, \Lambda^H q_0, \Lambda^2 q_0, \Lambda^{H^2} q_0, ..., \Lambda^l q_0, \Lambda^{H^l} q_0]$ and a block diagonal matrix $F = diag(1, B, B, ..., B)$ with $B = \begin{pmatrix} -i & 1 \\ i & 1 \end{pmatrix}$. If there exists a unitary matrix $\tilde{Q}$ such that $\tilde{Q}^H(\Lambda - \lambda I)\tilde{Q}G_e = G_o - \lambda G_e$ is a unitary matrix pencil in parametrized form, where G_o, G_e are unitary block diagonal matrices with block size at most two, then the unique QR decomposition of $\kappa(\Lambda, q_o, l)$ is given by $\tilde{Q}\tilde{R}$ with $\tilde{R} = \kappa(G_o G_e^H, e_1, l)$. From this a unique (real-valued) QR factorization of $D\tilde{A}$ is easily obtained. The construction of such a unitary matrix pencil in parametrized form from spectral data is a generalized inverse eigenproblem.

In this talk we present algorithms for discrete least-square approximations that are based on schemes for the solution of an inverse eigenproblem for unitary Hessenberg matrices $H = H(\gamma_1, ..., \gamma_n)$ and for unitary matrix pencils in parametrized form $G_o - \lambda G_e$.

Reichel, Ammar and Gragg observe in [1] that solving an inverse eigenproblem for unitary Hessenberg matrices is equivalent to computing Szegő polynomials, that is to computing polynomials that are orthogonal with respect to an inner product on the unit circle. The scheme for solving an inverse eigenproblem for unitary matrix pencils in parametrized form is developed from a backward stable algorithm given by Bunse-Gerstner and Elsner in [2] which reduces a unitary matrix pencil to parametrized form. It is shown that this is equivalent to computing rational functions that are orthonormal with respect to an inner product on the unit circle.

The algorithms require only $O(mn)$ arithmetic operations as compared with $O(mn^2)$ operations needed for algorithms that ignore the special structure of DA. We compare the presented algorithms with each other and with a general QR decomposition. We will see that the proposed algorithms produce consistently accurate results that are often better than those obtained by general QR decomposition methods for the least-squares problem.

References

[1] L. Reichel, G. S. Ammar and W. B. Gragg. Discrete Least Squares Approximation by Trigonometric Polynomials. *Math. Comp.* 57, pp 273 - 289, 1991.

[2] A. Bunse-Gerstner and L. Elsner. Schur Parameter Pencils for the Solution of the Unitary Eigenproblem. *Lin. Alg. and its Appl.*, 154 - 156, pp 741 - 778, 1991.

Bounding the Subspaces from Rank Revealing Two-Sided Orthogonal Decompositions

Ricardo D. Fierro

Department of Mathematics
University of California, Los Angeles
405 Hilgard Ave
Los Angeles, California 90024-1555
e-mail: fierro@math.ucla.edu

James R. Bunch

Department of Mathematics
University of California, San Diego
9500 Gilman Drive
La Jolla, California 92093-0112
e-mail: jbunch@math.ucsd.edu

Abstract. The singular value decomposition (SVD) is a widely used computational tool in various applications. However, in some applications the SVD is viewed as computationally demanding or difficult to update. The rank revealing QR (RRQR) factorization and the recently proposed rank revealing two-sided orthogonal (URV or ULV) decompositions are promising alternatives. In this paper we prove sharp *a posteriori* bounds for assessing the quality of the subspaces obtained by rank revealing URV or ULV decompositions. We implement the algorithms in an adaptive manner, which is particularly useful for applications where the "noise" subspace must be computed, such as in signal processing or total least squares. Our analysis shows that the quality of the URV or ULV decomposition depends on the quality of the estimated start vectors, and not on a gap condition. From our analysis we conclude that the rank revealing two-sided orthogonal decompositions may be more accurate alternatives to the SVD than the RRQR factorization.

December 1992; Revised February 1993.
UCLA CAM Technical Report 92-51

GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING CAN FAIL IN PRACTICE

LESLIE V. FOSTER[†]

Abstract. Even though Gaussian elimination with partial pivoting is very widely used, one can construct n by n matrices where the error growth in the algorithm is proportional to 2^{n-1}. Thus for moderate or large n, in theory, there is a potential for disastrous error growth. However, prior to this year no reports of such an example in a practical application have appeared in the literature. We present two related examples that arise naturally in practice and which lead to disastrous error growth in Gaussian elimination with partial pivoting.

Key words. Gaussian elimination, numerical stability

AMS(MOS) subject classifications. 65F05,65R20,65G05

1. Extended Abstract.

Gaussian elimination with partial pivoting (gepp) is one of the most widely used algorithms in scientific computing. When applied to an $n \times n$ matrix A it results in a factorization $PA = LU$, where P is a permutation matrix, L is lower triangular and U is upper triangular. Let $\dot{\mathbf{x}}$ represent the solution to $A\mathbf{x} = \mathbf{b}$ computed in floating point arithmetic on a computer with relative machine precision ϵ. Then it is known [Wil] that

$$\frac{\|\dot{\mathbf{x}} - \mathbf{x}\|_\infty}{\|\mathbf{x}\|_\infty} \le 4n^2 \text{cond}_\infty(A)\rho\,\epsilon \tag{1.1}$$

where $\mathbf{x}$ is the exact solution, $\text{cond}_\infty(A)$ is the condition number of A in the supremum norm and ρ is the growth factor,

$$\rho = \frac{\max_{i,j,k} |a_{i,j}^{(k)}|}{\max_{i,j} |a_{i,j}|} \tag{1.2}$$

with $a_{i,j}^{(k)}$ denoting the i,j element after the kth step of elimination. Thus gepp is considered numerically stable unless ρ is large.

The theory for Gaussian elimination with partial pivoting suggests that ρ can be very large. The sharpest bound is $\rho \le 2^{n-1}$ and this is attained, for example, for matrices A_n of the form [Wil]

$$A_5 = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ --1 & 1 & 0 & 0 & 1 \\ -1 & -1 & 1 & 0 & 1 \\ -1 & -1 & -1 & 1 & 1 \\ --1 & -1 & -1 & -1 & 1 \end{pmatrix} \tag{1.3}$$

[†] Department of Mathematics and Computer Science, San Jose State University, San Jose, California, 95192.

Thus for moderate or large n the growth factor can be large. However, more than 25 years ago Wilkinson reported:

> "It is our experience that any substantial increase in size of elements of successive A_n is extremely uncommon even with partial pivoting. $\cdots$ No example which has arisen naturally has in my experience given an increase by a factor as large as 16."

Since Wilkinson made his remarks, [DBMS] reports an example where ρ is 23 and [HH] reports several natural, non-contrived examples where the growth factor is between $n/2$ and n. Although the growth factors reported in these papers are larger than those mentioned by Wilkinson, they do not grow exponentially with n and are far from the theoretical limit of 2^{n-1}. Recently Wright [Wri] presented a class of practical examples where the growth factors do grow exponentially. Wright's paper and ours are related in that we both consider boundary value problems with coupled end conditions. Wright discretizes these problems using the multiple shooting method and we rewrite the differential equations as integral equations and use the quadrature method [Bak, DM]. The growth factor for our matrices can be closer to the theoretical limit than the growth factor for Wright's matrices and our examples can have large growth factors for problems arising from a single differential equation whereas Wright requires a system of two or more differential equations for large growth factors. Wright's examples involve sparse matrices and our matrices are dense. Finally, our results include an analysis of problems arising from Volterra integral equations. These are not discussed by Wright.

In [HH] Higham and Higham characterize all matrices A where partial pivoting has growth factor 2^{n-1}. However a matrix need not be of the form (1.3) or in the class of [HH] to still have a large growth factor. For example, if we replace all the -1's in the matrix A_n with random numbers selected uniformly between 0 and -1 the new matrix will still have a large growth factor for moderate or large n. For example for $n = 60$ the median growth factor for 30 such matrices was 1.9×10^{10}. This is less than the maximum growth factor of 5.7×10^{17} but is still quite large.

In this paper we will present several classes of problems that arise naturally in practice where the growth factor is less than 2^{n-1} but is still very large. First we discuss a class of first order boundary value problems where the discretization used is the trapezoid method. More generally, we then discuss a class Volterra integral equations where higher order Newton-Cotes numerical integration is used for the discretization. We illustrate that large growth factors can lead to large errors when using gepp with two specific practical examples - a solution mixture problem and a problem involving an LRC circuit. We compare our examples with those recently developed by Wright. Finally, we discuss the implications such examples have on software development and testing.

REFERENCES

[Bak] C. T. Baker, *The numerical treatment of integral equations*, Oxford University Press, Oxford, 1977.

[DBMS] J. Dongarra, J. Bunch, C. Moler, and G. Stewart, *Linpack User's Guide*, Society for Industrial and Applied Mathematics, Philadelphia, 1979.

[DM] L. Delves and J. Mohamed, *Computational methods for integral equations*, Cambridge Uni-verstiy Press, Cambridge, 1985.

[HH] N. Higham and D. Higham, *Large growth factors in Gaussian elimination with pivoting*, SIAM J. Matrix Anal. Appl., 10 (1989), pp. 155-164.

[Wil] J. Wilkinson, *The algebraic eigenvalue problem*, Oxford University Press, London, 1965.

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

Stable Algorithms for Fast Triangular Factorization of General Hankel and Toeplitz Matrices

Roland W. Freund
AT&T Bell Laboratories
600 Mountain Avenue, Room 2C–420
Murray Hill, New Jersey 07974–0636, U.S.A.

Extended Abstract

A matrix T is called *Toeplitz* if its entries are constant along each diagonal; a matrix H is called *Hankel* if its entries are constant along each anti-diagonal. There are tremendously many applications that require the solution of systems of linear equations whose coefficient matrices are nonsingular Hankel or Toeplitz matrices. For example, Hankel systems arise in connection with orthogonal polynomials, Padé approximation, the minimal realization problem in systems theory, and the Berlekamp-Massey algorithm for decoding Reed-Solomon and BCH codes. Applications leading to Toeplitz systems include time-series analysis, linear prediction, image processing, statistics, probability theory, and solution of integral equations.

It is well known that these special structures of the coefficient matrices can be exploited when solving Hankel or Toeplitz systems, and in both cases there are classical *fast* algorithms that require only $\mathcal{O}(N^2)$ arithmetic operations for the solution of systems of order N, as compared to $\mathcal{O}(N^3)$ operations for general systems. All fast Hankel solvers compute either an *inverse* triangular factorization of the Hankel matrix H of the type

$$U^T H U = D, \tag{1}$$

or a triangular factorization of H of the form

$$H = U^T D U. \tag{2}$$

Here, in (1) and (2), U is a unit upper triangular matrix, and D is a diagonal matrix. Similarly, all fast Toeplitz solvers compute either an inverse triangular factorization of the Toeplitz matrix T of the type

$$V^T T U = D, \tag{3}$$

or a triangular factorization of T of the form

$$T = V^T D U, \tag{4}$$

where U and V are unit upper triangular matrices, and D is again a diagonal matrix. The celebrated Levinson-Trench algorithm for Toeplitz matrices is of the type (3). The first Toeplitz solver based on (4) was proposed by Bareiss. The classical Hankel solver of type (1) is due to Trench. Algorithms based on (2) were devised by Rissanen and Kalman.

Unfortunately, for nonsingular matrices H and T, factorizations of the form (1) and (2), respectively (3) and (4), exist if, and only if, the matrix H, respectively T, is *strongly regular*, i.e., all its leading principal submatrices are nonsingular. Indeed, all classical fast Hankel and Toeplitz solvers require that the coefficient matrix is strongly regular. If H, respectively T, has singular submatrices, then breakdowns—triggered by division by 0—occur in these algorithms. On the other hand, it is well known that the classical Hankel and Toeplitz solvers can be extended to handle exactly singular leading principal submatrices, and numerous such algorithms have been proposed. These algorithms are again based on factorizations of the type (1)–(4), where now D is a block-diagonal matrix. However, in finite-precision arithmetic, it is not enough to skip only over exactly singular submatrices, and numerically robust Hankel and Toeplitz solvers also must be able to handle nonsingular, yet ill-conditioned leading principal submatrices. Two algorithms of this type were recently proposed by Freund and Zha. Both algorithms use so-called *look-ahead* techniques to skip over singular and ill-conditioned submatrices, and they generate inverse factorizations of the type (1), respectively (3), with block-diagonal D. First, they devised a look-ahead version of Trench's classical Hankel solver based on the decomposition (1) of H. Second, they developed a look-ahead version of the Levinson-Trench algorithm based on the inverse factorization (3) of T.

In this talk, we present two counterparts to the algorithms of Freund and Zha that are based on *direct* triangular factorizations of the type (2) and (4), rather than the decompositions (1) and (3). The first algorithm is a stable extension of the classical Hankel solver by Rissanen and Kalman for strongly regular matrices to general Hankel matrices H. It computes a factorization of H of the type (2) where U is still a unit upper triangular matrix and D is now in general a block-diagonal matrix. The second algorithm is a stable extension of the Bareiss algorithm for strongly regular matrices to general Toeplitz matrices T. It generates a decomposition of T of the form (4) where U and V are still unit upper triangular matrices and D is now block-diagonal. We give implementation details and operations counts for both algorithms, and we describe the look-ahead strategies used to detect singular and nonsingular, but ill-conditioned leading principal submatrices. We present numerical stability analyses for both algorithms. We consider the look-ahead Bareiss algorithm for certain special cases, such as Hermitian indefinite or banded Toeplitz matrices. Moreover, we discuss the implementation of the proposed algorithms on parallel and vector machines, and we show that they are superior to algorithms based on inverse triangular factorizations. We report results of numerical experiments with Hankel and Toeplitz systems with ill-conditioned submatrices of various kinds.

Finally, we present two generalizations of the proposed algorithms. First, we consider extensions to the more general case of block-Hankel and block-Toeplitz matrices. Toeplitz matrices are a special case of structured matrices classified by their displacement rank. Second, we discuss extensions of the look-ahead Bareiss algorithm to the factorization of matrices with low displacement rank.

A Wedderburn-Householder Generalization of the Singular Value Decomposition

Bob Funderlic [1]
Department of Computer Science
North Carolina State University
Box 8206, 121 Daniels
Raleigh, NC 27695-8206

Householder 93
Lake Arrowhead, California
June 13-18, 1993

Abstract

A class of matrix decompositons will be given based on Wedderburn's 1934 observation that

$$A - \frac{A\mathbf{x}\mathbf{y}^T A}{\mathbf{y}^T A\mathbf{x}} \tag{0.1}$$

has rank one less than that of A. Alston Householder gave a converse in his 1964 book and Cline and Funderlic [1977, LAA] provided considerable unifying theory associated with rank(A-B) along with generalizations of the rank one case of Wedderburn and Householder.

The Wedderburn-Householder result of (0.1) suggests a very general class of decompositions:

$$A = \sum_{i=1}^{k} \omega_i^{-1} A_i \mathbf{x}_i \mathbf{y}_i^T A_i,$$

$$A_{i+1} = A_i - \omega_i^{-1} A_i \mathbf{x}_i \mathbf{y}_i^T A_i, \quad \mathbf{y}_i^T A_i \mathbf{x}_i = \omega_i.$$

By choosing column vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ as certain linear combinations of the x's and y's, then

$$U_k^T A V_k = diag(\omega)$$

with U_k and V_k having full column rank k.

Based on a norm idea of Householder, it will be shown that the ω's, x's and y's may be chosen to give the singular value decomposition. Another choice gives the special generalization where the ω's are generalized singular values determined by ellipsoidal norms. Furthermore, the currently popular URV decompositions can be thought of as Wedderburn decompositions. The Wedderburn decompositions thus provide a fresh and promising approach to updating the singular value decomposition. More generally they unify several matrix decompositions through a historical progression of important linear algebra ideas starting with Wedderburn's rank reduction idea and progressing via Householder's norm approach to the modern view of the singular value decomposition.

[1] Joint work with Gene Golub, Stanford University

ON THE SOLUTION OF COMPLEX SYMMETRIC SYSTEMS WITH MULTIPLE RIGHT-HAND SIDES

E. GALLOPOULOS* AND V. SIMONCINI†

Abstract

We are interested in the solution of linear systems $A[x^{(1)},\ldots,x^{(s)}] = [b^{(1)},\ldots,b^{(s)}]$ where A is a complex symmetric matrix of dimension n. It is known that in designing iterative algorithms one can exploit the symmetry of A by using the indefinite product $(x,y) = x^T y$, instead of the conjugate transpose x^*y [3][5][4]. In order to overcome the lack of minimality properties of the corresponding Lanczos methods, the quasi-minimal residual approach has been recently used to design successful iterative methods for a single right-hand side [1][2]. The aim of this talk is (i) to present new block iterative methods to solve $A[x^{(1)},\ldots,x^{(s)}] = [b^{(1)},\ldots,b^{(s)}]$, which use the complex symmetric implementation of Lanczos algorithms together with a quasi-minimal residual approach; (ii) to present the theoretical justification, design and computational experience of a novel method, we term MRMULTI. We evaluate the performance of all methods using matrices from important applications areas and show that the new method is a competitive alternative to block and single right-hand side solvers.

REFERENCES

[1] R. W. FREUND, *Conjugate gradient-type methods for linear systems with complex symmetric coefficient matrices*, SIAM J. Sci. Stat. Comput., 13 (Jan. 1992), pp. 425–448.

[2] R. W. FREUND AND N. M. NACHTIGAL, *An implementation of the QMR Method based on coupled two-term recurrences*, Tech. Rep. 92.15, RIACS, June 1992.

[3] D. A. H. JACOBS, *A generalization of the conjugate-gradient method to solve complex systems*, IMA J. Numer. Anal., 6 (1986), pp. 447–452.

[4] T. K. SARKAR, X. YANG AND E. ARVAS, *A limited survey of various conjugate gradient methods for solving complex matrix equations arising in electromagnetic wave interactions*, Wave Motion, 10 (1988), pp. 527-546.

[5] G. MARKHAM, *Conjugate Gradient Methods for Indefinite, Asymmetric, and Complex Systems*, IMA J. Num. An. Vol. 10 (1990), pp. 155-170.

[*] Department of Computer Science and Center for Supercomputing Research and Development, Univ. of Illinois at Urbana-Champaign, Urbana, Illinois 61801. E-mail stratis@csrd.uiuc.edu. This research was supported by the the Department of Energy grant DOE DE-FG02-85ER25001 and the National Science Foundation under grant NSF CCR-9120105.

[†] Center for Supercomputing Research and Development, Univ. of Illinois at Urbana-Champaign, Urbana, Illinois 61801. E-mail simoncin@csrd.uiuc.edu. Supported by fellowship #2043597 from the Consiglio Nazionale delle Ricerche, Italy.

A Constrained Quadratic Form (Walter Gander & Urs Von Matt)

Let A be a real symmetric n-by-n matrix. We consider the problem of finding a vector $\mathbf{x}$ such that

$$\begin{aligned} \mathbf{x}^T A\mathbf{x} - 2\mathbf{b}^T\mathbf{x} &= \min, \\ \|\mathbf{x}\|_2 &= \alpha. \end{aligned}$$

Such constrained quadratic forms occur, for instance, in the area of nonlinear optimization.

The solution of the constrained quadratic form is analysed by means of the Lagrange equations. We are led to the so-called secular equation

$$f(\lambda) := \mathbf{b}^T(A+\lambda I)^{-2}\mathbf{b} = \alpha^2,$$

that we must solve for its largest zero λ.

Conventional approaches to solving the secular equation are based on the eigenvalue decomposition of A. Thus, their computational complexity is of order $O(n^3)$. For large matrices, however, this becomes infeasible.

Our new approach relies on Gauss quadrature to approximate the secular function $f(\lambda)$. We will give a short overview of the theory of Gauss quadrature as far as it will be used by our approach. This includes the discussion of orthogonal polynomials, the calculation of Gauss quadrature rules by means of an eigenvalue decomposition, and the error law.

In order to apply the theory of Gauss quadrature to approximating the secular function $f(\lambda)$ we use the Lanczos algorithm as a means to compute the orthogonal polynomials and their three-term recurrence relationship. Thus, we are able to give lower and upper bounds on the secular function. After k steps of the Lanczos algorithm we have

$$\mathcal{L}_k(\lambda) \le f(\lambda) \le \mathcal{U}_k(\lambda).$$

Our overall algorithm for the solution of the constrained quadratic form executes the Lanczos algorithm until the secular function $f(\lambda)$ is approximated accurately enough. The precise termination criterion is considered in more detail.

As soon as the Lagrange multiplier λ is known we can compute the solution $\mathbf{x}$ of the constrained quadratic form from the linear system

$$(A+\lambda I)\mathbf{x} = \mathbf{b}$$

by a conjugate gradient method, for instance.

The Orthogonal Quotient-Difference Algorithm

The orthogonal qd-algorithm is a novel way of computing the singular value decomposition of a bidiagonal matrix. Unlike the corresponding subroutine in the LINPACK library, which computes all the singular values to the same absolute accuracy, the orthogonal qd-algorithm is capable of computing all the singular values to high relative accuracy.

First, we present the tool of the generalized Givens transformation, which is designed to introduce a given value σ into a vector. It can be described by the equation

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \sqrt{x_1^2 + x_2^2 - \sigma^2} \\ \sigma \end{bmatrix}.$$

Then, we introduce the so-called orthogonal qd-steps which form the heart of the orthogonal qd-algorithm. One such step is the orthogonal left lu-step with shift s

$$Q \begin{bmatrix} L \\ \sigma I \end{bmatrix} = \begin{bmatrix} U \\ \sqrt{\sigma^2 + s^2} I \end{bmatrix},$$

which transforms the lower bidiagonal matrix L into the upper bidiagonal matrix U by means of the orthogonal matrix Q. The singular values of L are diminished by the amount of the shift s, i.e.

$$\sigma_i^2(U) = \sigma_i^2(L) - s^2.$$

By an appropriate sequence of orthogonal qd-steps we can compute the decomposition

$$\begin{bmatrix} B \\ 0 \end{bmatrix} = P \begin{bmatrix} 0 \\ \Sigma \end{bmatrix} Q^T,$$

where P denotes an orthogonal $(2n)$-by-$(2n)$ matrix, Q denotes an orthogonal n-by-n matrix, and Σ denotes a diagonal n-by-n matrix with the singular values of B.

Finally, we discuss the various deflation opportunities during the execution of the orthogonal qd-algorithm, as well as a way to obtain the final singular value decompoition

$$B = U\Sigma V^T.$$

Some numerical results are presented.

Construction of Richardson Polynomials and Polynomial Preconditioning

K. Gärtner[1]

Abstract. The solution of linear nonsymmetric problems is one of the fields where much progress was made during the last years [1]. The most often used Lanczos-type iterative methods are based on three-term recursions, extended now to longer recursions to fulfil stability requirements. More or less explicitly the information from an iteration with the transposed matrix is used too. Error minimizing generalized conjugate-gradient methods like GMRES are ignored here due to the fact that we are interested in applications where the L_∞ or the H_1 norm of the error has a natural meaning. In contrast, GMRES minimizes the H_2 norm of the error in the case of a second order elliptic partial differential equation discretized by finite elements. Another option is Richardson iteration ([2,3]), which accomplishes the construction of the residual polynomial not by a three-term recurrence relation but by its factorization. In exact arithmetic the minimal polynomial of degree n can be determined in $2n$ matrix-vector operations as in the Lanczos case. From the point of view of polynomial preconditioning the Richardson process has the advantage of lower storage and computational costs (as long as the zeros of the polynomial are real) and offers in some applications the possibility to include a few well separated eigenvalues explicitly in the polynomial without further effort. Such eigenvalues appear systematically if a MILU preconditioner is used. Concerning stability requirements in finite arithmetic it seems worth noting, that long term recurrencies can be constructed simply by defining the residual polynomial of degree n recursively as $P_n(x) = P_{n_k}(P_{n_{k-1}}(...P_{n_1}(x)))$ with logarithmic ($\log(n)$) growing storage requirements.

The solution of the linear system is often part of an outer iteration process. A polynomial approximating zero sufficiently well over some part of the spectrum (including the complex eigenvalues) and having positive values less than 1 at the remaining real eigenvalues is very well suited for an approximate solution in a Newton iteration when damping is necessary: the computed correction in its spectral decomposition is either correct or too small. Thus an implicit damping occurs which can be exploited systematically in some problems with boundary layers (different scales in space which are reflected in the eigenvalue distribution).

The stability problem is met again in the Richardson iteration: the ordering of the zeros of the polynomial is relevant. Stable orderings for Chebyshev polynomials (T_n, U_n) are known since more than twenty years [4], the general case was studied by Reichel [5] and leads to Leja orderings. Appling Richardson iteration, it seems natural to exploit Richardson iteration itself to estimate the outermost parts of the spectrum of the iteration matrix and to introduce this knowledge in a polynomial preconditioner for reducing the condition number of the remaining problem. This approach diminishes the numerical stability problems and the number of inner products. The latter may be of considerable interest if, on parallel machines, the trend to increased relative communication cost (measured relative to the floating-point performance of the nodes) continues.

The main problem of constructing a fixed polynomial preconditioner is that only incomplete information is available. In view of the fact that well separated eigenvalues are often observed in applications the construction of the residual polynomials should respect them. The degree of a polynomial incorporating once all estimates of separated eigenvalues is limited rather fast by the errors of these estimates and the possible ill-conditioning of the matrix. Thus, one is forced to introduce as many copies of the estimates as stability requires. In other words: one replaces an isolated eigenvalue by a disk with the radius of the error centered at the estimate and applies the Leja ordering procedure. The other possibility is to use a polynomial of limited degree in a cyclic manner: this leeds to the recursive definition of the residual polynomial and has the advantage that further eigenvalue estimates are made now in the range of the polynomial. A limited degree (n_k) of the polynomial on each recursive level (k) would restrict all zeros of the polynomial to a set excluding an ε_k-disk centered at zero. This seems to be a natural way to split the original problem into simpler ones. In the recursive process one has the choice to proceed further and further by Richardson iteration or to fix the preconditioning polynomial and to switch to a Lanczos like procedure. In the Richardson case the inclusion of new eigenvalue estimates is no problem and can be done also on the recursive level before the actual one. In the Lanczos case one would have to accept large values of the modulus of an intermediate polynomial over parts of the spectrum, to introduce look-ahead strategies, or to restart the Lanczos process. In this talk we are going to show:

- how during Richardson iteration error components related to a few eigenvalues can be made dominant,
- how these eigenvalues can be estimated, and

[1] Interdisciplinary Project Center for Supercomputing, ETH Zurich, ETH-Zentrum, CH-8092 Zurich, Switzerland

- how a polynomial preconditioner can be used to damp out these error components afterwards.

The application to the solution of the semiconductor device equations and some other test problems is considered. For instance, for the sherman5 matrix scaled by its diagonal (the iteration matrix has then at least two complex conjugate eigenvalue pairs with modulus > 1), a polynomial of degree 10 was generated. With this polynomial as preconditioner, the residual norm decreased monotonically during 100 plain iterations. Nevertheless the polynomials modulus is > 1 on parts of the boundary of the convex polygon constructed from its zeros.

References

[1] R. W. Freund, G. H. Golub and N. M. Nachtigal, Iterative Solution of linear systems, Acta Numerica (1992), 1 - 44

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

[3] M. Eiermann and W. Niethammer, On the construction of semiiterative methods, SIAM J. Numer. Anal. 20 (1983), 1153 - 1160

[4] V. I. Lebedev and S. A. Finogenov, Ordering of the iterative parameters in the cyclical Chebyshev iterative methods (russ.), USSR Comput. Math. and Math. Phys. 11 (1971), 155 - 170

[5] L. Reichel, The application of Leja Points to Richardson Iteration and Polynomial Preconditioning, Linear Algebra Appl. 154 - 156 (1991), 389 - 414

Geometric Mesh Partitioning and Nested Dissection

John R. Gilbert[*] Gary L. Miller[†] Shang-Hua Teng[‡]

January 30, 1993

1 Introduction

Many sparse matrix computations require that the graph of a matrix be partitioned into pieces of roughly equal size, with few connections between the pieces. Nested dissection [1] uses recursive partitioning to construct orderings for Gaussian elimination with low fill and good parallel load balance. In implementing iterative methods on distributed-memory computers, the efficiency of matrix-vector multiplication depends on how well the matrix is divided among the processors [2].

Graph partitioning is a hard combinatorial problem, and it pays to take advantage of all the available information about any particular instance. If the graph comes from a linear system that arises in physical simulation, it usually has an underlying geometric structure: its vertices are located at specific points in space. Vertices near to each other in the graph are also near to each other in space, in some sense.

A partitioner can exploit this geometric information in several ways. First, good space partitions may be easier to find than good graph partitions. Second, using a technique called *geometric sampling*, the partitioner can work with a small randomly selected subset of mesh points but still generate a good partition with high probability; this makes the partitioner more efficient. Finally, most computational meshes are composed of elements (triangles or tetrahedra, for example) that are in some sense *well shaped*. Bounds on the shapes of the elements can sometimes be used to prove bounds on the quality of the partition.

Here we report on the implementation of a geometric partitioning algorithm. The basis for the algorithm is theoretical work by Miller, Teng, Thurston, and Vavasis on geometric separators [4, 5, 7]; we describe several simplifications and extensions of that work, which lead to an efficient and practical mesh partitioner. We compare and contrast our approach to partitioning with the spectral methods investigated by Hendrickson, Leland, Pothen, Simon, and Liou [3, 6]. We describe experiments with our partitioner in partitioning meshes for parallel matrix-vector multiplication, and in producing *geometric nested dissection* orderings for sparse factorization.

2 Geometric separators

Miller et al. [4] define a class of "d-dimensional" graphs called *overlap graphs*. This class includes planar graphs and k-nearest neighbor graphs. Most significantly, it includes all finite element meshes, no matter how irregular, provided that the angles in the individual mesh elements are bounded away from zero or from 180°. The theoretical result says that a d-dimensional mesh of size n can be partitioned evenly by cutting $O(n^{(d-1)/d})$ vertices. (The exponent $(d-1)/d$ is what one would expect from the trivial case of regular square grids. In two dimensions, an n-point square grid can be partitioned by a cut of size $n^{1/2}$, and an n-point cubic grid in three dimensions needs a cut of size $n^{2/3}$.)

To find this separator, the mesh in d dimensions is mapped onto the surface of a sphere in $d+1$ dimensions. The mapping can be chosen in such a way that any d-dimensional hyperplane through the center of the sphere divides the mesh approximately in half, and that most such hyperplanes cut only a small number of mesh edges (at most $O(n^{1-1/d})$, to be precise).

Practical implementation requires attention to several issues. A key step is to find a *center point* for a set of points in $d+1$-dimensional space. The theoretical algorithm to find a centerpoint uses linear programming and is too slow in practice; we have implemented a heuristic that runs very fast and gives very satisfactory results. (In separate work, Eppstein and Teng recently proved that the heuristic always gives good approximate center points.) Finding a good cutting hyperplane is also an important problem; in theory a randomly chosen hyperplane should be good enough, but in practice it is worthwhile to spend some effort searching.

Geometric sampling is crucial to our implementation's efficiency in several ways. The approximate center point is based on a sample of the input points. When searching for a good hyperplane, a candidate is rejected quickly if it does not partition a sample of the points well.

Some applications require an exact 50/50 split of the input points rather than the approximate split guaranteed by the theory. We describe some simple techniques for evening the split without making the separator much larger.

Our first implementation uses Matlab; it is a sequential program. We expect that the partitioning algorithm itself will parallelize well. In most graph partitioners, the flow of data in the algorithm is mostly along the edges of the graph, which introduces a sort of bootstrapping problem for parallel implementation: the partitioner will run faster on the parallel machine if a good partitioning among the processors has already been found. In our partitioner, on the other hand, most of the computation deals only with the vertex coordinates, and does not depend on the edges of the graph at all.

References

[1] Alan George and Joseph W. H. Liu. *Computer Solution of Large Sparse Positive Definite*

[*] Xerox Palo Alto Research Center, 3333 Coyote Hill Road, Palo Alto, California 94304-1314 (gilbert@parc.xerox.com). Copyright © 1993 by Xerox Corporation. All rights reserved.

[†] School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213 (glmiller@theory.cs.cmu.edu). This author's work was supported in part by National Science Foundation grant DCR-8713489.

[‡] Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139 (steng@bourbaki.mit.edu).

Systems. Prentice-Hall, 1981.

[2] Steven Hammond and Robert Schreiber. Solving unstructured grid problems on massively parallel computers. Technical Report TR 90.22, Research Institute for Advanced Computer Science, 1990.

[3] Bruce Hendrickson and Robert Leland. An improved spectral graph partitioning algorithm for mapping parallel computations. Technical Report SAND92-1460, Sandia National Laboratories, Albuquerque, NM, 1992.

[4] Gary L. Miller, Shanghua Teng, and Steven A. Vavasis. A unified geometric approach to graph separators. In *Proceedings of the 32nd Annual Symposium on Foundations of Computer Science*, pages 538–547. IEEE, 1991.

[5] Gary L. Miller and William Thurston. Separators in two and three dimensions. In *Proceedings of the 22nd Annual Symposium on Theory of Computing*. ACM, 1990.

[6] Alex Pothen, Horst D. Simon, and Kang-Pu Liou. Partitioning sparse matrices with eigenvectors of graphs. *SIAM Journal on Matrix Analysis and Applications*, 11:430–452, 1990.

[7] Shang-Hua Teng. *Points, Spheres, and Separators: A Unified Geometric Approach to Graph Partitioning*. PhD thesis, Carnegie-Mellon University, Pittsburgh, Pennsylvania, August 1991.

A Modified Newton Method Based on a Partial Cholesky Factorization

Philip E. GILL

Department of Mathematics
University of California at San Diego
La Jolla, California 92093-0112

The effectiveness of Newton's method for finding an unconstrained minimizer of a strictly convex twice continuously differentiable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ has prompted the proposal of various *modified* Newton methods for the nonconvex case.

The class of *linesearch modified Newton methods* generates a sequence $\{x_k\}$ of improving estimates of a local minimizer by performing a linesearch along a path formed from a descent direction s_k and a direction of negative curvature d_k. If these directions are *sufficient* in the sense that the sequences $\{s_k\}$ and $\{d_k\}$ are bounded and satisfy

$$\begin{aligned} \nabla f(x_k)^T s_k \rightarrow 0 \quad &\text{implies} \quad \nabla f(x_k) \rightarrow 0 \text{ and } s_k \rightarrow 0, \\ d_k^T \nabla^2 f(x_k) d_k \rightarrow 0 \quad &\text{implies} \quad \min\{\lambda_{\min}(\nabla^2 f(x_k)), 0\} \rightarrow 0 \text{ and } d_k \rightarrow 0, \end{aligned}$$

then every limit point of the sequence $\{x_k\}$ will satisfy the second-order necessary conditions for optimality.

It has been observed in practice that the number of iterates at which the Hessian is positive definite is large compared to the total number of iterations. Since linesearch methods revert to Newton's method when the Hessian is sufficiently positive definite, it would seem sensible to use a modified Newton method based on the most efficient method for solving a symmetric positive-definite system. This is the motivation for the modified Cholesky factorization proposed by Gill and Murray. However, it has been shown by Moré and Sorensen that this factorization may not give directions of negative curvature that are sufficient in the sense above.

In this talk we propose an efficient method for computing a descent direction and a direction of negative curvature that is based on a partial Cholesky factorization of the Hessian. The partial Cholesky factorization of H is a variant of the standard Cholesky factorization with diagonal pivoting. The algorithm is defined in *outer-product form*, where the Schur complement associated with the unfactorized part of H is updated explicitly at each step.

It will be shown that this factorization may be used to define an algorithm with not only the efficiency and simplicity of the Cholesky factorization, but also the guarantee of convergence when used in conjunction with a suitable linesearch.

Rounding errors in the Gauss quadrature calculations and in computing continued fractions[1]

Abstract

Gene H. Golub,
Dept. of Comp. Science, Stanford University, Stanford, USA
and
Zdenek Strakos,
Institut of Computer Science, Academy of Science of the Czech Republic, Prague, Czech Republic

We consider the effect of rounding errors in computing the Gauss quadrature for the distribution function $\omega(\lambda)$ with the N points of increase λ_i, $i = 1, \ldots, N$, $0 < \lambda_1 < \lambda_2 < \ldots < \lambda_N$. From the relation between orthogonal polynomials, the Lanczos method and Jacobi matrices, it is well known that the abscissas μ_j and weights ω_j, $j = 1, \ldots, n$, of the n-point Gauss quadrature for the Riemann-Stieltjes integral

$$(1) \qquad \int_\zeta^\xi f(\lambda) d\omega(\lambda) = \sum_{i=1}^{N} \sigma_i^2 f(\lambda_i)$$

can be determined as the eigenvalues and the square of the first elements of the normalized eigenvectors of the Jacobi matrix T_n having the Lanczos coefficients as its elements [1], [2]. In exact arithmetic, the error of the quadrature formula is expressed as

$$(2) \qquad \sum_{i=1}^{N} \sigma_i^2 f(\lambda_i) = \sum_{j=1}^{n} \omega_j f(\mu_j) + R_n(f),$$

$$(3) \qquad R_n(f) = \sum_{i=1}^{N} \sigma_i^2 \{ f(\lambda_i, \mu_1, \mu_1, \ldots, \mu_n, \mu_n) \prod_{k=1}^{n} (\lambda_i - \mu_k)^2 \},$$

where $f(\lambda, \mu_1, \mu_1, \ldots, \mu_n, \mu_n)$ is the $2n$-th divided difference of a function f with respect to the abscissas $\lambda, \mu_1, \mu_1 \ldots, \mu_n, \mu_n$, [2]. In particular, for the important case $f(\lambda) = \lambda^{-1}$, (2) can be written into the form

$$(4) \qquad (T_N^{-1})_{11} = (T_n^{-1})_{11} + R_n(\lambda^{-1}).$$

Consider a matrix A, $A = Udiag(\lambda_i)U^T$, $U^T U = I$, and the initial "residual" vector $r^0 / \parallel r^0 \parallel = \sum_{i=1}^{N} \sigma_i u_i$, u_i is the i-th normalized eigenvector of A. Then (3) gives

$$(5) \qquad R_n(\lambda^{-1}) = (1 / \parallel r^0 \parallel^2) \parallel x - x^n \parallel_A^2,$$

where $\parallel x - x^n \parallel_A$ is the energy norm of the error of the conjugate gradient (CG) process for A, r^0 [3], [4].

Rounding errors may affect the computation crucially. As a consequence, the Lanczos coefficients, the actually computed abscissas $\hat{\mu}_j$, and weights $\hat{\omega}_j$, $j = 1, \ldots, n$, may differ substantially from their theoretical counterparts. Despite that, the n-point Gauss quadrature frequently gives very precise results. Our analysis explains this behavior.

Using the backward error analysis of the Lanczos process developed by Greenbaum [5] we will show, that the results of the n-point Gauss quadrature for (1), computed in finite precision arithmetic, is characterized by

$$(6) \qquad \sum_{i=1}^{N} \sigma_i^2 f(\lambda_i) = \sum_{j=1}^{n} \hat{\omega}_j f(\hat{\mu}_j) + \overline{R}_n(f) - P_n(f),$$

where $P_n(f)$ is a modest multiple of the machine precision ε. $\overline{R}_n(f)$ is given in terms of $\{ f(\lambda, \hat{\mu}_1, \hat{\mu}_1, \ldots, \hat{\mu}_n, \hat{\mu}_n) \prod_{k=1}^{n} (\lambda - \hat{\mu}_k)^2 \}$. The original distribution function $\omega(\lambda)$ is not used, but using a particular distribution function $\varpi(\lambda)$, possibly having many more points of increase (denoted by $\overline{\lambda}_k$) than N, and whose points of increase all lie within tiny intervals about the points of increase λ_i of the distribution function $\omega(\lambda)$. Moreover, the sum of "weights" $\sum_{k \in C_i} \overline{\sigma}_k^2$, where C_i denotes the set of indices k for which $\overline{\lambda}_k$ is close to λ_i, approximates the original "weight" σ_i^2 of λ_i, $i = 1, 2, \ldots, N$. The exact values $\overline{\lambda}_k$, $\overline{\sigma}_k$, $k = 1, 2, \ldots$, are determined by the actual values of rounding errors in steps 1 thru n of the finite precision Lanczos process.

In this way, the **total error (including the roundoff errors)** of the Gauss quadrature for (1) is expressed as the **truncation error** of the Gauss quadrature for a specific but different problem.

In particular, for $f(\lambda) = \lambda^{-1}$, the rate of convergence of the associated continued fraction computed in a **finite precision arithmetic** is described by

[1] This paper is being submitted for consideration for the XII. Householder Meeting on Numerical Linear Algebra

$$(T_N^{-1})_{11} = (\hat{T}_n^{-1})_{11} + o(1)\frac{1}{\| r^0 \|^2} \| x - \hat{x}^n \|_A \tag{7}$$

where the superscript ˆ denotes for the actually computed quantities. Thus, (7) is the **finite precision analogy of exact arithmetic results (4)-(5)**. This analogy seems natural, but its derivation is far from trivial [4].

We emphasize relations between quadratures, orthogonal polynomials, Jacobi matrices, continued fractions, Lanczos and Conjugate gradient methods. Exploiting these relations one can easily reformulate a question from one area into the language of the other area. As it is demonstrated on the examples mentioned above, this approach may lead to interesting results.

References

[1] G.H. Golub and J.H. Welsch, Calculation of Gauss Quadrature Rules, Math. Comp. **23**, 221-230, 1969.

[2] W. Gautschi, A Survey of Gauss-Christofel Quadrature Formulae, In: E.B. Christofel - The Influence of His Work on Mathematics and the Physical Sciences, P.L. Bultzer and F.Fehér (eds.), Birkhauser, Boston, 73-157, 1981.

[3] G. Dahlquist, G.H. Golub and S.G. Nash, Bounds for the Error in Linear Systems, in Proc. of the Workshop on Semi-Infinite Programming, R.Hettich ed., Springer, 154-172, 1978.

[4] G.H. Golub and Z. Strakos, Estimates in Quadratic Formulas (in preparation).

[5] A. Greenbaum, Behavior of Slightly Perturbed Lanczos and Conjugate Gradient Reccurences, Linear Algebra and its Appl. 113, 7-63, 1989.

A New Bound on the Convergence Rate of the GMRES Algorithm

Anne Greenbaum*

January, 1993

A new bound on the convergence rate of the GMRES algorithm is given in terms of the eigenvalues of the iteration matrix and the norms of the orthogonal complements of each eigenvector, relative to the space spanned by the other eigenvectors. The bound consists of the sum of squares of values of a polynomial at each of the eigenvalues, and a weighted sum of squares of differences between the values of the polynomial at different eigenvalues. The weights involve the reciprocals of the norms of the orthogonal complements of the corresponding eigenvectors. It is demonstrated numerically that this bound is quite reasonable in many cases where other error bounds, such as those involving pseudo-spectra or the condition number of the eigenvector matrix, give large overestimates.

Let A be a diagonalizable matrix with eigendecomposition $A = V \Lambda V^{-1}$, where $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ and V has columns $v^1, \ldots, v^n$, each with norm one. Let p be any polynomial. Then it can be shown that the Frobenius norm of $p(A)$ satisfies

$$\|p(A)\|_F^2 = \sum_{i=1}^{n} |p(\lambda_i)|^2 - \sum_{i=2}^{n} \sum_{j=1}^{i-1} |p(\lambda_i) - p(\lambda_j)|^2 \, (V^*V)_{ij} \, (V^*V)_{ij}^{-1}.$$

Here $(V^*V)_{ij}$ denotes the (i,j) element of V^*V and $(V^*V)_{ij}^{-1}$ denotes the (i,j) element of $(V^*V)^{-1}$. Let w^i be the orthogonal complement of v^i, relative to the space spanned by the other eigenvectors. Then the Frobenius norm of $p(A)$ is bounded by

$$\|p(A)\|_F^2 \le \sum_{i=1}^{n} |p(\lambda_i)|^2 + \sum_{i=2}^{n} \sum_{j=1}^{i-1} |p(\lambda_i) - p(\lambda_j)|^2 \, \frac{\min\left\{\sqrt{1 - \|w^i\|^2}, \sqrt{1 - \|w^j\|^2}\right\}}{\|w^i\| \, \|w^j\|}. \tag{1}$$

If the eigenvectors of A are nearly orthogonal so that the weights in the second sum in (??) are small, then this bound on $\|p(A)\|_F$ is approximately the same as that for a normal matrix with the same eigenvalues. In contrast, if the eigenvectors of A are far from orthogonal, then the second sum in inequality (??) will dominate, unless the values of $p(\lambda_i)$ and $p(\lambda_j)$ are almost equal for all eigenvalues λ_i and λ_j corresponding to nearly-dependent eigenvectors.

The residual at step k of the GMRES algorithm for solving a linear system $Ax = b$ is given by

$$r^k = P_k(A) r_0$$

where P_k is the k^{th} degree polynomial with value one at the origin that minimizes $\|r^k\|$. It follows that for any other such polynomial p_k, we have

$$\|r^k\| / \|r^0\| \le \|p_k(A)\| \le \|p_k(A)\|_F.$$

Taking p_k to be the polynomial that minimizes the expression in (??) we obtain the bound

$$(\|r^k\| / \|r^0\|)^2 \le \sum_{i=1}^{n} |p_k(\lambda_i)|^2 + \sum_{i=2}^{n} \sum_{j=1}^{i-1} |p_k(\lambda_i) - p_k(\lambda_j)|^2 \, \frac{\min\left\{\sqrt{1 - \|w^i\|^2}, \sqrt{1 - \|w^j\|^2}\right\}}{\|w^i\| \, \|w^j\|}.$$

While this bound is not sharp, we argue that it is very reasonable and gives good estimates of the actual convergence rate of the GMRES method (for the worst possible initial residual) in many cases where other bounds are overly pessimistic.

*Courant Institute of Mathematical Sciences, 251 Mercer St., New York, NY 10012.
This work was supported by DOE contract DEFG0288ER25053.

Downdating the Singular Value Decomposition

Ming Gu[1]

The *singular value decomposition* (*SVD*) of a matrix $A \in \mathbf{R}^{m\times n}$ with $m > n$ is

$$A = (U_1 \;\; U_2)\begin{pmatrix} D \\ 0 \end{pmatrix} V^T \quad ,$$

where $U = (U_1 \quad U_2) \in \mathbf{R}^{m\times m}$ and $V \in \mathbf{R}^{n\times n}$ are orthonormal, with $U_1 \in \mathbf{R}^{m\times n}$ and $U_2 \in \mathbf{R}^{m\times(m-n)}$; and $D \in \mathbf{R}^{n\times n}$ is non-negative diagonal. The columns of U and V are the *left singular vectors* and the *right singular vectors* of A, respectively; the diagonal entries of D are the *singular values* of A.

In many least squares and signal processing applications, we repeatedly update A by appending a row or a column, or downdate B by deleting a row or a column. After each update or downdate, we compute the *SVD* of the resulting matrix. In this paper we consider the problem of downdating the *SVD*.

We only consider the case where a row is deleted. Without loss of generality, we further assume that the last row is deleted. Thus, we can write

$$A = \begin{pmatrix} \bar{A} \\ a^T \end{pmatrix} \quad ,$$

where $\bar{A} \in \mathbf{R}^{(m-1)\times n}$ is the downdated matrix. Let the *SVD* of $\bar{A}$ be

$$\bar{A} = (\bar{U}_1 \;\; \bar{U}_2)\begin{pmatrix} \bar{D} \\ 0 \end{pmatrix} \bar{V}^T \quad ,$$

where $\bar{U} = (\bar{U}_1 \;\; \bar{U}_2) \in \mathbf{R}^{(m-1)\times(m-1)}$ and $\bar{V} \in \mathbf{R}^{n\times n}$ are orthonormal, with $\bar{U}_1 \in \mathbf{R}^{(m-1)\times n}$ and $\bar{U}_2 \in \mathbf{R}^{(m-1)\times(m-n-1)}$; and $\bar{D} \in \mathbf{R}^{n\times n}$ is non-negative diagonal. We would like to compute the *SVD* of $\bar{A}$ by taking advantage of some knowledge of the *SVD* of A.

There are three downdating problems:

- Problem 1: Given V, D and a, compute $\bar{V}$ and $\bar{D}$;
- Problem 2: Given U (or U_1), V and D, compute $\bar{U}$ (or $\bar{U}_1$), $\bar{V}$ and $\bar{D}$;
- Problem 3: Given U (or U_1) and D, compute $\bar{U}$ (or $\bar{U}_1$) and $\bar{D}$.

[1] Joint work with Stanley C. Eisenstat.

We show that for Problem 1

$$\bar{A}^T\bar{A} = \bar{V}\bar{D}^2\bar{V}^T = V(D^2 - zz^T)V^T \quad ,$$

where $z = V^T a \in \mathbf{R}^n$. Assuming that there is a solution, the singular values of $\bar{A}$ can be computed by the eigendecomposition

$$D^2 - zz^T = S\bar{D}^2 S^T \quad ,$$

where $S \in \mathbf{R}^{n\times n}$ is orthonormal. The right singular vector matrix $\bar{V}$ can be computed as VS. We present Algorithm I to solve Problem 1 stably.

Since Problem 1 is associated with the eigendecomposition of $D^2 - zz^T$, small perturbations in D and a can cause large perturbations in $\bar{D}$ and $\bar{V}$. We also analyze the ill-conditioning of the singular values.

We show that for Problems 2 and 3 there exists a column orthonormal matrix $F \in \mathbf{R}^{(m-1)\times n}$ such that

$$\bar{A} = FCV^T \quad ,$$

where $C \in \mathbf{R}^{n\times n}$ is of the form

$$C = \left(I - \frac{1}{1+\mu}uu^T\right) D \quad ,$$

with u a vector and $\mu \geq 0$ a scalar. The singular values of $\bar{A}$ can be computed by the *SVD*

$$C = Q\bar{D}W^T \quad ,$$

where Q and W are orthonormal. The left singular vector matrix $\bar{U}_1$ can be computed as FQ. The right singular vector matrix $\bar{V}$ can be computed as VW. We ignore $\bar{U}_2$ here for simplicity. We present Algorithm II to solve Problems 2 and 3 stably.

For Problems 2 and 3, the singular values are well-conditioned with respect to perturbations in input data, whereas the singular vectors can be very sensitive to such perturbations.

Problems 1 and 2 have been considered by Bunch and Nielsen. They reduce Problem 1 to the eigendecomposition of $D^2 - zz^T$ as well. But their scheme for finding this eigendecomposition can be unstable. They solve Problem 2 by reducing it to Problem 1. This risks solving a well-conditioned problem using an ill-conditioned process for the singular values.

Algorithm I solves Problem 1 in $O(n^3)$ time, and Algorithm II solves Problems 2 and 3 in $O(mn^2)$ time. We show that Algorithm I can be accelerated by the fast multipole method of Carrier, Greengard and Rokhlin to solve Problem 1 in $O(n^2)$ time, and that Algorithm II can be accelerated to solve Problems 2 and 3 in $O(mn)$ time. This is an important advantage for large matrices.

Finally we point out that all these results hold similarly for downdating the *SVD* of an $m \times n$ matrix A with $m \leq n$.

Title

Gulliksson

Solving constrained and weighted linear least squares

Abstract

When solving *nonlinear*, constrained and weighted least squares problem with the Gauss-Newton method, *linear*, constrained and weighted least squares problems on the form

$$\begin{aligned} &\min_{x \in \mathbf{R}^n} \frac{1}{2}(b_2 - A_2 x)^T W_2 (b_2 - A_2 x) \qquad (1) \\ &\text{s.t.} \quad A_1 x = b_1, \end{aligned}$$

where W_2 is a diagonal weight matrix, arise as subproblems. Weighted linear least squares problems with very different weights also emerge when using interior point methods for solving linear programming problems.

We present an algorithm for solving (1) where arbitrarily large weights can be handled which is stable, simple and easy to extend to other problem classes, see SIAM J. Matrix Anal. Appl., 29:268–296, 1992. Define W as a diagonal weight matrix with nonnegative, possibly infinite, elements. The algorithm is based on doing a *weighted QR decomposition*

$$A\Pi = Q \begin{bmatrix} R \\ 0 \end{bmatrix} \qquad (2)$$

of A, where $R \in \mathbf{R}^{n \times n}$ is an upper triangular matrix, Π is a permutation matrix and Q satisfies

$$Q^T W Q = W. \qquad (3)$$

We call a matrix, Q, that satisfies the relation (3) *W-invariant.*

A new backward rounding error analysis for the solution of (1), using the weighted QR decomposition is presented and a special perturbation analysis is applied to get explicit normwise relative errors on the solution.

Using a variant of the weighted QR decomposition we will extend our algorithm to underdetermined weighted linear least squares problems on the form

$$\begin{aligned} &\min_{x \in \mathbf{R}^n} \frac{1}{2} x^T W x \qquad (4) \\ &\text{s.t.} \quad A_1 x = b_1 \end{aligned}$$

where the weighted QR decomposition now is done on A_1^T.

If we define

$$A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

$W_2^{-1} = M_2 = \text{diag}(\mu_i)$, with $\mu_{p+1} > 0$ then an equivalent formulation of (1) is

$$\begin{bmatrix} 0 & 0 & A_1 \\ 0 & M_2 & A_2 \\ A_1^T & A_2^T & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ x \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix}, \qquad (5)$$

where λ_1 is the vector of Lagrange multipliers and $M_2\lambda_2$ is the residual. We will show how iterative refinement for (1) can be done using the formulation in (5). By using perturbation analysis we analyze the convergence behavior of the iterative process and give a few numerical examples.

Last, but not least, we describe how modified Gram-Schmidt can be generalized to constrained and weighted linear least squares. The usual way of normalizing the vectors, q_i, in the Gram-Schmidt method when using a weighted norm would be to impose the condition $q_i^T W q_i = 1$. We will instead use the normalizing condition $q_i^T W q_i = w_i$ and in that way we are able to handle constraints too. Especially we will explain the geometry involved in determining a W-invariant Q when using the modified Gram-Schmidt for weighted linear least squares.

A modified GMRES algorithm for flow problems.

Bertil Gustafsson
Dept. of Scientific Computing, Uppsala University, Uppsala, Sweden
and
Per Lötstedt
SAAB-SCANIA, Linköping, Sweden and
Dept. of Scientific Computing, Uppsala University, Uppsala, Sweden

Extended abstract.

The stationary Euler equations of fluid flow are a system of first order partial differential equations of importance to the simulation of the air flow around airplanes in the aircraft industry. The numerical solution of such equations is often obtained by an iterative method such as Runge-Kutta time-stepping (RK) (or truncated and restarted Richardson iteration), ref. 1, or the Generalized Minimum Residual method (GMRES), ref. 2. The convergence of these methods can be explained as a combination of two effects, refs. 3,4,5:

* wave propagation of smooth error modes out through the open boundaries,
* damping of the amplitude of oscillatory error modes.

The algorithms we advocate here enhance the wave propagation. Theoretical investigation of linear model problems using Fourier analysis support this view, refs. 3,5. Substantial improvements in convergence rate and increased robustness are obtained in numerical experiments. The gain in efficiency of the algorithms is probably also due to better damping of oscillatory modes, but this is more difficult to verify theoretically.

In Krylov subspace methods such as RK and GMRES, see e g ref. 6, the propagation of smooth error waves is modeled by the differential equation imbedded in a time-dependent problem. Wave propagation depends on the coefficient in front of the first Krylov vector. The second coefficient influences the damping of smooth waves.

A modified truncated GMRES (mGMRES) method introduced in refs. 3,4 is tested and compared with the original GMRES and the very successful RK. In the modified method the coefficient multiplying the first Krylov vector is fixed. One advantage with RK and (m)GMRES is that only residual evaluations are needed in each iterative step. GMRES and mGMRES also have a local optimality property. A disadvantage with GMRES and mGMRES is the extra storage requirements compared to RK. Another problem with GMRES is that sometimes the iteration stagnates and the residual ceases to decrease after a number of iterations. The remedy is to introduce mGMRES or a proper preconditioner. Both RK and mGMRES have a secured wave propagation property. The (m)GMRES algorithm for nonlinear problems is as in ref. 7.

We consider two kinds of preconditioning of the Euler equations: residual smoothing and multigrid iteration. Both of them improve the propagation of smooth error waves. The residual smoothing technique in ref. 8 is chosen. It contains the original one, ref. 9, as a special case and has better convergence properties and is less sensitive to parameters. The parameters in the multigrid acceleration are selected so that the wave speed of smooth error modes increases.

In 2D problems with constant coefficients modeling the equations of inviscid flow mGMRES is often superior to both RK and GMRES when counting the number of iterations. The computed coefficients of (m)GMRES are transformed to equivalent RK-coefficients. The behavior of the coefficients of mGMRES are in general much smoother than those of GMRES.

The convergence of the iterative algorithms to the solution of the stationary Euler equations is studied in numerical experiments in 2D and 3D. The number of iterations and the total CPU-time are compared for three-stage RK, GMRES and mGMRES combined with residual smoothing and multigrid iteration. It is found that the convergence rate is improved considerably by techniques that enhance the propagation of smooth error modes out from the computational domain. Stagnation problems with GMRES disappear with mGMRES or multigrid preconditioning. When GMRES stagnates then the first coefficient vanishes and the others start oscillating wildly. mGMRES is often more efficient than RK in terms of number of iterations but not always if the CPU-seconds are compared. There is an extra evaluation of the residual and more overhead per iteration in (m)GMRES. However, mGMRES appears to be more robust than RK. The computational examples include channel geometries, airfoils and wings at subsonic, transonic and supersonic speeds.

References.

1. A. Jameson, W. Schmidt, E. Turkel, AIAA paper 81-1259, 1981.
2. Y. Saad, M. H. Schultz, SIAM J. Sci. Stat. Comput., 7, 856-869, 1986.
3. B. Gustafsson, P. Lötstedt, The GMRES method improved by securing fast wave propagation, manuscript, 1992.
4. B. Gustafsson, P. Lötstedt, Convergence acceleration for first order systems, to appear in Proceedings of the Conference on Numerical Methods for Fluid Dynamics, Reading, 1992.
5. P. Lötstedt, SIAM J. Numer. Anal., 29, 1370-1394, 1992.
6. R. W. Freund, G. H. Golub, N. M. Nachtigal, Acta Numerica , 1992.
7. P. Brown, Y. Saad, SIAM J. Sci. Stat. Comput., 11, 450-481, 1990.
8. R. Enander, Grid patching and residual smoothing for computations of steady state solutions of first order hyperbolic systems, Ph. D. thesis, Uppsala University, 1992.
9. A. Jameson, Comm. Pure Appl. Math., XLI, 507-549, 1988.

CONTROL STRATEGIES FOR THE ITERATIVE SOLUTION OF NONLINEAR EQUATIONS IN ODE SOLVERS

KJELL GUSTAFSSON* AND GUSTAF SÖDERLIND†

Abstract. We develop new control strategies for handling the iterative solution of nonlinear equations in ODE solvers, i.e.

- automatic switching between fixed-point and Newton iterations,
- deriving an "optimal" convergence rate with respect to total work per unit step,
- a strategy for when to reevaluate the Jacobian,
- a strategy for when to refactorize the iteration matrix, and
- coordination with stepsize selection

Examples will be given, that demonstrate that the new overall strategy works efficiently. In particular, the new strategy admits a restrained stepsize variation without refactorizations, thus permitting the use of a smoother stepsize sequence. The strategy is of equal importance for Runge-Kutta and multistep methods.

1. Introduction. The numerical integration of an ODE $\dot{y} = f(y)$ by implicit time-stepping methods leads to the problem of solving a nonlinear equation on every step. The generic structure of this equation is

$$y = \gamma h f(y) + \psi,$$

where h is the stepsize, γ is a constant characteristic of the discretization method, and ψ is a known vector. In nonstiff computations, i.e. when $hL[f] \ll 1$, where $L[\cdot]$ is the Lipschitz constant, fixed-point iterations are used — convergence is fast and the iterations are inexpensive. In stiff computations $hL[f] \gg 1$, and fixed-point iterations do not converge. Instead one uses (some variant of) Newton's method. The ability to use large steps motivates the extra expense incurred by this iteration.

The Newton iteration reads

$$\begin{aligned}(I - \gamma h J)\,\delta y^k &= -y^k + \gamma h f(y^k) + \psi \\ y^{k+1} &= y^k + \delta y^k,\end{aligned}$$

where J is some approximation to the Jacobian $\partial f/\partial y$. The iteration matrix $M = I - \gamma h J$ varies with J and h; this may call for reevaluations of the Jacobian and/or refactorizations of M. Likewise, the convergence of the iteration (whether fixed-point or modified Newton) will depend on the stepsize and should be controlled such that efficiency is maintained. Thus, strategies for an efficient solution of the nonlinear equation interact with the stepsize selection strategy; the important considerations are listed in the abstract above.

2. Optimal convergence and switching. A switch from fixed-point to Newton iteration can easily be accomplished based on monitoring the convergence rate. As the stepsize increases, the convergence rate of fixed-point iteration slows down and eventually causes a strong limitation on stepsize. Newton iterations are then advantageous.

Let h_α denote the stepsize associated with the optimal convergence rate of fixed-point iterations and h_r the stepsize suggested by accuracy considerations alone. Switching from fixed-point to Newton is carried out if

$$\frac{h_r}{h_\alpha} > Q$$

where the factor Q determines how large a stepsize increase is deemed necessary for Newton's method to be less expensive per unit step. In principle, Q should depend on the size of the problem, but a large Q will require very small iteration errors.

Switching back to fixed-point iteration could be done based on the Jacobian J which is available during Newton iteration — one can easily check whether $\gamma h f$ is a contraction or not. Several switching strategies of a similar type have been used in practical computations earlier, but remain important in good adaptive implementations.

Convergence is linear both for fixed-point and modified Newton iterations. The convergence rate α can be estimated in the computational process; α is typically proportional to h when the stepsize varies. Convergence occurs only if $\alpha < 1$, and is faster the smaller the value of α. The total work per unit step increases quite rapidly, however, should the convergence rate slow down and approach 1. For very short steps, convergence is fast, but the step does not take the integration very far.

For multistep as well as Runge-Kutta methods, one can show that under very general conditions

$$\frac{m}{h} \sim -\frac{1}{\alpha \log \alpha},$$

where m/h is work per unit step and α is the convergence rate. This function has a minimum at $\alpha = 1/e$, which is therefore "optimal" for a linearly converging iteration. More precisely, it does not pay off to increase the stepsize at the expense of a slower convergence, since total work per unit step then increases without a corresponding accuracy gain. Therefore, one should never allow a slower convergence rate than $1/e$. Since the minimum is quite flat, a slightly conservative strategy might be preferred; we recommend $\alpha \leq \alpha^* = 0.2$. Experience with practical computations provides strong support for this strategy.

3. Refactorization and reevaluation. For modified Newton iteration, we assume that the Jacobian J remains fixed for several consecutive steps, and that the iteration matrix $M = I - \gamma h' J$ is not necessarily refactorized when the stepsize varies, i.e. h' denotes the stepsize when M was last factorized. One can show that the convergence rate is

$$\alpha \leq \nu \|(h'J)^{-1}\delta(hJ)\|,$$

where $\delta(hJ)$ denotes the current value of hJ deviates from the value used when forming M. The factor $\nu \approx 1$ in stiff computation, and the linearization $\delta(hJ) \approx \delta hJ + h\delta J$ readily yields the convergence rate estimate

$$\alpha \leq \left|\frac{\delta h}{h'}\right| + \|J^{-1}\delta J\|.$$

Thus the convergence rate is bounded by the relative stepsize change, plus the "relative change" in the Jacobian since its last update. This furnishes a simple but effective strategy for reevaluating the Jacobian and/or refactorizing M due to stepsize variation. No changes are necessary as long as α is small; if it becomes larger

[*] Computer Science, MJH 309, Stanford University, Stanford, CA 94305, USA (speaker)

[†] Computer Science, Lund University, P.O. Box 118, S-221 00 Lund, Sweden

than α^*, check whether this can be attributed to $|\delta h/h'|$. If so, a refactorization is sufficient, otherwise the Jacobian must have changed significantly, and it is necessary to reevaluate J and then refactorize M.

Practical experience with this strategy shows that it is not at all necessary to refactorize M whenever the stepsize changes, nor is it necessary to prevent stepsize changes for efficiency reasons. On the contrary, moderate stepsize variation is permitted and can be used to advantage with elaborated error control, see Gustafsson (1991) and (1992). This makes for a smoother local error control (closer to the prescribed tolerance), and the global error may show a more consistent behavior for different tolerances.

4. References.

GUSTAFSSON, K. (1992). *Control of Error and Convergence in ODE Solvers*, Ph.D. thesis, Dept. of Automatic Control, Lund.

GUSTAFSSON, K. (1991). *Control Theoretic Techniques for Stepsize Selection in Explicit Runge-Kutta Methods*, ACM TOMS **17:4**, pp 533–554.

GUSTAFSSON, K., M. LUNDH, and G. SÖDERLIND, *A PI stepsize control for the numerical solution of ordinary differential equations*, BIT **28:2**, pp 270–287.

HAIRER, E., AND WANNER, G. (1991). *Solving Ordinary Differential Equations II.* Springer.

A weakly stable, generically superfast algorithm for non-Hermitian Toeplitz systems

Martin H. Gutknecht[1]

Abstract. It has been known for a long time (Levinson, 1947; Durbin, 1959; Trench, 1964; and others) that an $N \times N$ linear system with Toeplitz matrix $\mathbf{T}$ can be solved fast, in $O(N^2)$ operations. More recently, superfast $O(N \log^2 N)$ algorithms were found (Musicus, 1984; de Hoog, 1987; Ammar and Gragg, 1986-88). However, these algorithms require a strongly regular matrix; i.e., one whose leading principal minors are all nonzero. But what is worse is that these methods are easily seen to be unstable if any of these minors is very small. Hence, the remaining challenge has been to find fast solvers that are stable (in an appropriate sense).

Recently, Chan and Hansen developed a generalization of the Levinson algorithm that seemed to resolve this challenge: it produces an inverse block LDU decomposition of $\bar{h}$, and requires $O(N^2) + O(N\bar{h}^2)$ operations if $\bar{h}$ is the size of the largest block. However, some doubts remained regarding the optimality and the generality of the block steps, and, in Summer 1992, Freund and Zha indeed showed that the case of two directly neighboring blocks was not handled correctly by this algorithm.

Also in 1992, the author introduced four new algorithms that were all based on general recurrences in the Padé table [1]. Two are of 'Levinson type' and require $O(N^2) + O(N\bar{h}^2)$ operations, the other two are of 'Schur type' and require $O(N \log^2 N) + O(N\bar{h}^2)$ operations. However, none of these algorithms reduces in the case of a matrix with well-conditioned leading principal submatrices to the Levinson or to the Schur (or Bareiss) algorithm. In particular, they differ markedly from the Chan and Hansen algorithm and also from the Freund and Zha look-ahead Levinson algorithm that has been submitted shortly after our proposals [2]. Since then, yet another pair of algorithms has been developed in joint work with Marlis Hochbruck [2]. Starting from Padé recurrences also, we have reformulated them in terms of linear algebra; they are similar (though not identical) to the Freund-Zha algorithm and a corresponding look-ahead Schur algorithm that is currently being developed by Freund.

In this talk we want to reformulate also the algorithms from [1] in the language of matrix analysis, so that they become accessible to a wider audience. Of prime interest are the fast Levinson type and the superfast Schur type algorithms that were originally based on generalized sawtooth recurrences since these recurrences are the simplest ones among all these methods. The price one pays is a loss of the symmetry that is one of the features of the classical Levinson and Schur algorithms.

Although detailed stability proofs have not been worked out, one can expect that the algorithms we proposed are weakly stable, i.e., forward stable for well-conditioned problems.

This is partly joint work with Marlis Hochbruck from the University of Würzburg.

References

[1] M. H. GUTKNECHT, Stable row-recurrences in the Padé table and generically superfast look-ahead solvers for non-Hermitian Toeplitz systems, IPS Research Report 92-14, IPS, ETH-Zürich, August 1992.

[2] M. H. GUTKNECHT AND M. HOCHBRUCK, Look-ahead Levinson and Schur Algorithms for non-Hermitian Toeplitz Systems. In preparation.

[1] Interdisciplinary Project Center for Supercomputing, ETH Zurich, ETH-Zentrum, CH-8092 Zurich, Switzerland

[2] The December 1991 date on the Freund and Zha report is misleading; the paper was not available before September 1992.

Look-ahead Levinson and Schur Algorithms for non-Hermitian Toeplitz Systems

Martin H. Gutknecht[1] and Marlis Hochbruck[2]

Abstract. Systems of linear equations with *Toeplitz* coefficient matrices $\mathbf{T}_N \in \mathbf{C}^{N,N}$ arise in many important applications. It is well-known that the Toeplitz structure of $\mathbf{T}_N$ can be exploited when solving linear systems with Toeplitz coefficient matrices. There are mainly two basically different types of algorithms of complexity $O(N^2)$ for the recursive solution of a Toeplitz system $\mathbf{T}_N\mathbf{x}_N = \mathbf{b}_N$. Levinson type algorithms compute – at least implicitly – an LDU decomposition of the inverse of $\mathbf{T}_N$. Schur type algorithms, such as the Bareiss algorithm, compute an LDU decomposition of $\mathbf{T}_N$ itself. The main difference between both types of algorithms is that Levinson's algorithm requires the computation of two inner products of size $n+1$ in step n, while Schur's algorithm is of purely recursive nature. Thus, Schur type algorithms might be better suited for implementation on parallel architectures.

In addition, a number of superfast algorithms, which are of complexity $O(N \log^2 N)$, have been proposed (e. g. by de Hoog; Musicus; Ammar and Gragg). These algorithms are generalizations of Schur type algorithms which do not compute the complete LDU decomposition of $\mathbf{T}_N$ but instead only find the last row of L and the last column of U and then apply an inversion formula (like the Gohberg-Semencul formula) for inverting $\mathbf{T}_N$.

However, a major drawback of all these algorithms is that they require all the leading principal submatrices of $\mathbf{T}_N$ to be nonsingular. If only one of these submatrices $\mathbf{T}_n$ is singular, the algorithms break down. In finite precision arithmetic such exact breakdowns are rare, but near singular or ill-conditioned submatrices may occur frequently. In this case, the algorithms become unstable and may compute solutions which are far away from the exact solution.

In the past few years, the interest in the solution of indefinite or nonsymmetric Toeplitz systems has grown substantially. Chan and Hansen proposed a look-ahead solver that jumps over ill-conditioned submatrices. Since this algorithm estimates the condition of each submatrix $\mathbf{T}_n$, the overhead is quite large, even if no look-ahead steps are performed. In addition, there is still a possibility of breakdowns if two or more consecutive look-ahead steps are necessary. Recently, Freund and Zha came up with a generalization of the classical Levinson algorithm, which can overcome these difficulties. Their approach is based on an interpretation of the Levinson algorithm in terms of formally biorthogonal polynomials. The new algorithm can handle exact and near breakdowns and has low overhead. At the same time, Gutknecht published a paper on general row recurrences in the Padé table and showed how these recurrences are related to the solution of Toeplitz systems. Making use of this Padé connection, he proposed different types of fast and superfast solvers, which can handle exact and near breakdowns. However, none of the algorithms he proposed is a generalization of either the Schur or the Levinson algorithm in the sense that it reduces to one of the classical algorithms in the absence of look-ahead steps.

In this talk we present generalizations of the algorithms by Levinson and Schur which can also handle the general case. The LDU decompositions computed in the classical algorithms then become block LDU decompositions, as in the algorithms of Freund and Zha and of Chan and Hansen. Therefore, it is possible to compute the solution of any subsystem $\mathbf{T}_n\mathbf{x}_n = \mathbf{b}_n$, if desired. The new algorithms are derived from a purely linear algebra point of view. In particular, we do not need to have knowledge about the Padé connection, although the recurrences are closely related to some in the Padé table.

Moreover, we show how this approach can be used to obtain a superfast look-ahead solver for general Toeplitz systems, which reduces to the variant by Ammar and Gragg if no look-ahead steps are necessary. To our best knowledge, this is the first implementation of a superfast Toeplitz solver with look-ahead.

The look-ahead strategy used in our algorithms is based on local information only. In particular, no condition estimates of the submatrices $\mathbf{T}_n$ are necessary. Therefore, the overhead of the look-ahead algorithms is low.

Finally, we present several numerical examples for the new fast and superfast algorithms. These examples indicate that the proposed algorithms have good numerical properties. Since the computation of the residual and the application of the inversion formula requires only $O(N \log N)$ arithmetic operations when FFT techniques are used, iterative refinement to improve the accuracy can be applied very effectively. In all our examples, one step of iterative refinement was sufficient to reduce the relative error to the order of the machine precision.

[1] Interdisciplinary Project Center for Supercomputing, ETH Zurich, ETH-Zentrum, CH-8092 Zurich, Switzerland. E-Mail: mhg@ips.ethz.ch

[2] Institut für Angewandte Mathematik und Statistik, Universität Würzburg, Am Hubland, D-8700 Würzburg, Germany. E-Mail: marlis@mathematik.uni-wuerzburg.de

Experience with Regularizing CG Iterations

Per Christian Hansen*

The numerical treatment of ill-posed problems has always been a challenge, because the computation of a useful solution depends to a very high degree on choosing the appropriate way to regularize, or stabilize, the solution (the ordinary least-squares solution x_{LSQ} is not useful since it is completely dominated by noise). A variety of theoretical and numerical algorithms have appeared over the years [4], and the regularization method due to Tikhonov is undoubtedly the most well known and perhaps also the most used method.

Iterative regularization methods that only require matrix-vector products are very important alternatives to Tikhonov's method for large-scale problems (and they may also be useful on some high-performance computers where matrix-vector products can be computed fast).

Among the iterative regularization methods, the conjugate gradient (CG) method applied to the normal equations $A^TAx = A^Tb$ is probably the most promising—provided that it is implemented as in CGLS [2, p. 560] or in the algorithms based on Lanczos bidiagonalization, such as LSQR [2, p. 566] and the new algorithm based on modified moments [1]. See also the approach in [5] based on Gauss quadrature. The iteration vector $x^{(k)}$ for the CG process is said to exhibit *semi-convergence*: during the first iterations, the error $\|x^{(k)}-x^*\|_2$ (where x^* is the exact, unregularized solution to a problem without noise) decreases, while at later stages of the CG process the error increases again until $x^{(k)}$ converges to the least-squares solution x_{LSQ}. We stress that x_{LSQ} is not the desired solution since it is completely dominated by the noise.

This behavior of $x^{(k)}$ can, to some extent, be explained in terms of spectral filtering. Let $\theta_j^{(k)}$, $j = 1,\ldots,k$ denote the Ritz values associated with applying k steps of CG to A^TA with starting vector A^Tb, and let $A = \sum_{i=1}^n \sigma_i u_i v_i^T$ denote the SVD of A. Then $x^{(k)}$ can be written as a filtered expansion:

$$x^{(k)} = \sum_{i=1}^{n} f_i^{(k)} \frac{u_i^T b}{\sigma_i} v_i , \qquad \textit{with} \qquad f_i^{(k)} = 1 - \prod_{j=1}^{k} \frac{\theta_j^{(k)} - \sigma_i^2}{\theta_j^{(k)}} . \tag{1}$$

Here, $f_i^{(k)}$ are the *filter factors* for CG. It is easy to see that $f_i^{(k)}$ will be close to 1 if some $\theta_j^{(k)}$ has converged to σ_i^2. Moreover, if no $\theta_j^{(k)}$ has converged to σ_i^2, then $f_i^{(k)} = \sigma_i^2 \sum_{j=1}^k (\theta_j^{(k)})^{-1} + O(\sigma_i^4/(\theta_i^{(k)})^2)$, and if σ_i is small then $f_i^{(k)}$ is also small. In other words, the spectral filtering is associated with the number of converged Ritz values which, in turn, is related to the number of iterations k. Hence, k essentially plays the role of a regularization parameter.

The above analysis illustrates that a full understanding of the regularizing properties of CG iterations requires a good understanding of the approximation properties of the Ritz values $\theta_i^{(k)}$—in infinite precision as well as finite precision. Work along this line has been presented by van der Sluis & van der Vorst [11]. In this talk, we will present some new results derived in collaboration with Dianne P. O'Leary & G. W. Stewart, Univ. of Maryland. In particular, we give conditions in which small Ritz values will not appear until each of the significant eigenvalues has been resolved. Our results are consistent with those of [11] in that the decay of the coefficients $|u_i^Tb|/\sigma_i$, which is related to the discrete Picard condition [7], plays a central role.

In a practical implementation of regularizing CG, a reliable algorithm for choosing the optimal number of iterations is crucial, due to the semi-convergent nature of the CG process when applied to these problems. We demonstrate that the method of generalized cross-validation (GCV) [2, p. 604]—which is a reliable method for Tikhonov regularization—is not as reliable for the CG method, especially because the underlying GCV function cannot be computed correctly without explicit knowledge of all the singular values of A. We also demonstrate that a stopping rule based on the L-curve criterion [8, 9] seems to be useful for regularizing CG iterations. Some of this work has been presented in the survey paper [6] written jointly with Martin Hanke, Univ. of Karlsruhe.

In our talk we will also discuss a "hybrid" implementation of the CG process, based on [10] and [2, p. 604], and which incorporates regularization adaptively in each step of the CG process. The "hybrid" version is well suited if the CG process starts to approximate small eigenvalues of A^TA before all the large eigenvalues have been captured. In these instances, the adaptive regularization in each CG step aims to filter out the influence of the smallest Ritz values on $x^{(k)}$. We show that the L-curve criterion is suited for choosing the regularization parameter in each CG step.

Finally, we will report on our comparison of CG regularization with other methods such as Tikhonov regularization and truncated SVD (TSVD). Surprisingly, we find that the optimal (i.e., minimal) error for CG and TSVD is often smaller than that for Tikhonov's method when both the matrix A and the right-hand side b are perturbed. We will discuss a possible explanation for this, namely, that the "sharper" filter factors of TSVD and CG are better suited for suppressing the errors in A than the Tikhonov filter factors.

Our numerical examples come from a collaboration with J. Christensen-Dalsgaard, Århus University on inverse helioseismology [3].

*UNI•C (Danish Computing Center for Research and Education), Building 305, Technical University of Denmark, DK-2800 Lyngby, Denmark. Email: unipch@wuli.uni-c.dk.

References

[1] M. Berry & G. H. Golub, *Estimating the largest singular values of large sparse matrices via modified moments*, Numer. Alg. 1 (1991), 353–374.

[2] Å. Björck, *Least Squares Methods*; in P. G. Ciarlet & J. L. Lions, *Handbook of Numerical Analysis*, Vol. I, North-Holland, 1990.

[3] J. Christensen-Dalsgaard, P. C. Hansen & M. J. Thompson, *GSVD analysis of helioseismic inversions*, submitted to Monthly Notices of the Royal Astronomical Society.

[4] N. W. Engl, *Regularization methods for the stable solution of inverse problems*, Surveys on Mathematics for Industry (1993), to appear.

[5] G. H. Golub & U. von Matt, *Quadratically constrained least squares and quadratic problems*, Numer. Math. 59 (1991), 561–580.

[6] M. Hanke & P. C. Hansen, *Regularization methods for large-scale problems*, Surveys on Mathematics for Industry (1993), to appear.

[7] P. C. Hansen, *The discrete Picard condition for discrete ill-posed problems*, BIT 30 (1990), 658–672.

[8] P. C. Hansen, *Analysis of discrete ill-posed problems by means of the L-curve*, SIAM Review 34 (1992), 561–580.

[9] P. C. Hansen & D. P. O'Leary, *The use of the L-curve in the regularization of discrete ill-posed problems*, SISSC, to appear.

[10] D. P. O'Leary & J. A. Simmons, *A bidiagonalization-regularization procedure for large-scale discretizations of ill-posed problems*, SISSC 2 (1981), 474–489.

[11] A. van der Sluis & H. A. van der Vorst, *SIRT- and CG-type methods for the iterative solution of sparse linear least-squares problems*, Lin. Alg. Appl. 130 (1990), 257–302.

Parallelism Versus Stability in Linear Equation Solvers

Nicholas J. Higham*

January 29, 1993

Extended abstract for the Householder Symposium XII meeting.

Parallel algorithms in matrix computations tend to be less stable than their serial counterparts. This fact has become clear in recent years as more parallel algorithms are derived and more error analyses are done of both existing and new algorithms. In this talk I analyse the tradeoff between parallelism and stability for linear equations solvers. A recurring theme in the talk is that for a parallel method the backward error often depends on the condition number.

I begin by considering algorithms for solving triangular systems, concentrating on algorithms that are not simply rearrangements of the usual substitution algorithm (which, of course, is known to have ideal stability properties). A method of particular interest because of the generality of the underlying idea can be described as follows. If $L \in I\!R^{n\times n}$ is lower triangular, we can write $L = L_1L_2\dots L_n$, where L_k is an elementary lower triangular matrix that differs from the identity only in the kth column. To solve $Lx = b$, we write $x = L^{-1}b = L_n^{-1}L_{n-1}^{-1}\dots L_1^{-1}b$, and we evaluate this product in parallel in $\log_2 n$ stages using a tree structure. This evaluation requires more flops than substitution, but it requires less parallel stages. Sameh and Brent [6] have shown that this method has a backward error bound proportional to $\kappa^2(L)\|L\|u$, where u is the unit roundoff. I will discuss the sharpness of this bound and the average case stability (as opposed to the worst case given by the bound).

Another parallel triangular solver, also based on matrix inversion, employs the formula

$$\begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix}^{-1} = \begin{bmatrix} L_{11}^{-1} & 0 \\ -L_{22}^{-1}L_{21}L_{11}^{-1} & L_{22}^{-1} \end{bmatrix}^{-1}.$$

The idea is to invert L_{11} and L_{22} in parallel, and then evaluate the (2,1) block. I explain the relation of this method to Strassen's inversion method [8]. The L_{ii} inversions can be done recursively by the same technique. Note that although methods based on matrix inversion are usually avoided because of potential numerical instability, they are worth considering in the parallel context because instability can be shown not to occur for well-conditioned matrices [3].

A parallel method for solving sparse triangular systems with many right-hand sides has recently been considered by Pothen, Schreiber and others. The method employs a partition into sparse factors of the product form of the inverse of the coefficient matrix. I show that while the method can be unstable, stability is guaranteed if a certain scalar that depends on the matrix and the partition is small, and that this scalar is small when the matrix is well-conditioned [5]. Moreover, when the partition is chosen so that the factors have the same sparsity structure as the coefficient matrix, the backward error matrix can be taken to be sparse.

Turning to full systems of equations I briefly consider Gaussian elimination with parallel pivoting strategies. One particular method carries out operations only on ***adjacent rows***:

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - (a_{ik}^{(k)}/a_{i-1,k}^{(k)})\,a_{i-1,j}^{(k)}.$$

Before carrying out the operation we interchange the rows, if necessary, to ensure that the multiplier is bounded by one. In the basic serial implementation, we introduce zeros in the first column in the order $(n,1)$, $(n-1,1)$, $\dots$, $(2,1)$, and then work from the bottom to the middle of each of the remaining columns in turn. This strategy avoids the sequential search for a pivot of partial pivoting and allows row operations to be overlapped. I explain the relation of this method to another parallel Gaussian elimination algorithm mentioned by Wilkinson in [10, pp. 236–237] and Gallivan et al. [4], and discuss the stability, making use of results of Sorensen [7] and Trefethen and Schreiber [9].

An excellent example of how modifying a point algorithm for parallel computation can worsen the stability is block LU factorization. I quantify the instability of block LU factorization for general matrices and explain why it is unconditionally stable only for matrices that are block diagonally dominant by columns [2].

I outline a complexity argument that suggests we need to look beyond elimination methods to obtain efficient linear equations solvers on massively parallel machines. A classic algorithm in the NC complexity class (polylogarithmic run time on a parallel machine) is one devised by Csanky (1976); it turns out to be very unstable. A better NC algorithm is Newton's method for the inverse, first considered by Schulz in 1933. I discuss its practical speed of convergence and its stability.

Finally, I indicate the possible courses of action when a method is found to be unstable. We can apply fixed precision iterative refinement if the instability is not too severe. Or we can redo the computation with a slower but more stable method—a strategy proposed by Demmel [1].

*Department of Mathematics, University of Manchester, Manchester, M13 9PL, England (na.nhigham@na-net.ornl.gov).

References

[1] James W. Demmel. Trading off parallelism and numerical stability. Technical Report UCB/CSD-92-702, Computer Science Division, University of Calfornia, Berkeley, 1992.

[2] James W. Demmel, Nicholas J. Higham, and Robert S. Schreiber. Block *LU* factorization. Numerical Analysis Report No. 207, University of Manchester, England, February 1992. Submitted to Journal of Numerical Linear Algebra with Applications.

[3] Jeremy J. Du Croz and Nicholas J. Higham. Stability of methods for matrix inversion. *IMA Journal of Numerical Analysis*, 12:1–19, 1992.

[4] K. A. Gallivan, R. J. Plemmons, and A. H. Sameh. Parallel algorithms for dense linear algebra computations. *SIAM Review*, 32:54–135, 1990.

[5] Nicholas J. Higham and Alex Pothen. Stability of the partitioned inverse method for parallel solution of sparse triangular systems. Numerical Analysis Report No. 222, University of Manchester, England, October 1992.

[6] A. H. Sameh and R. P. Brent. Solving triangular systems on a parallel computer. *SIAM J. Numer. Anal.*, 14:1101–1113, 1977.

[7] D. C. Sorensen. Analysis of pairwise pivoting in Gaussian elimination. *IEEE Trans. Comput.*, C-34:274–278, 1985.

[8] V. Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:354–356, 1969.

[9] L. N. Trefethen and R. S. Schreiber. Average-case stability of Gaussian elimination. *SIAM J. Matrix Anal. Appl.*, 11(3):335–360, 1990.

[10] J. H. Wilkinson. *The Algebraic Eigenvalue Problem.* Oxford University Press, 1965.

Semi-circulant solvers and boundary corrections for first-order pde

S. Holmgren[†] and K. Otto[†]

Department of Scientific Computing, Uppsala University
Box 120, S-751 04 Uppsala
Sweden

November 1992

Abstract

We consider solving systems of equations arising from time-dependent and time-independent first-order linear pde.

$$Bu = b$$

The matrix B is a discretization of a system of n_c pde in 2D using a five-point operator on an $m_1 \times m_2$-grid. The bandwidth of the matrix is $2n_c m_1$, but the number of nonzero elements is only $\mathcal{O}(m_2 m_1)$. Thus, the memory and arithmetic requirements for solving the systems using Gaussian elimination are prohibitive. To store the LU-factors, $\mathcal{O}(m_2 m_1^2)$ memory positions are required. The factorization requires $\mathcal{O}(m_2 m_1^3)$ a.o. and each backsubstitution $\mathcal{O}(m_2 m_1^2)$ a.o.

Previously we have developed an iterative solution procedure based on a CG-like iterative method combined with *semi-circulant* preconditioners [1]. A semi-circulant preconditioner can be considered as a fast direct solver ($\mathcal{O}(m_2 m_1 \log_2 m_1)$ arithmetic complexity) for a pde problem closely related to the original one.

$$Mu = c$$

In the semi-circulant matrix M two approximations are introduced. The coefficients in the differential operator are approximated by constants in the space direction associated with m_1, and the Dirichlet and outflow boundary conditions in the same space direction are replaced by periodic boundary conditions. For problems with constant coefficients in the m_1-direction we obtain.

$$\begin{aligned} B &= M + SV^T \\ V &= \operatorname{diag}(v_1, \ldots, v_{m_2}) \\ S &= \operatorname{diag}(s, \ldots, s) \\ s &= \begin{bmatrix} I_{(n_c)} & 0 \\ 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ 0 & I_{(n_c)} \end{bmatrix} \end{aligned} \tag{1}$$

The matrix SV^T is of rank $2n_c m_2$ and contains the corrections of the boundary conditions.

Often the main problems in constructing and analyzing numerical methods for first-order pde are caused by the numerical boundary conditions required at the outflow boundaries. Semi-circulant preconditioners have proved to be efficient, and for some model problems with a B according to eq. (1) we have argued that the number of iterations required are independent of problem size [2].

We now construct a memory efficient solver for linear systems with a coefficient matrix B according to eq. (1). The solver is based on the Fourier technique used for the semi-circulant preconditioners. The boundary corrections SV^T are resolved by using the Sherman-Morrison-Woodbury formula.

$$\begin{aligned} B^{-1} &= (M + SV^T)^{-1} = (I - M^{-1}SP^{-1}V^T)M^{-1} \\ P &= I + V^T M^{-1} S \end{aligned} \tag{2}$$

There are several ways of exploiting this factorization, none of which requires more than $\mathcal{O}(m_2 m_1 + m_2^2)$ memory positions. When computing $u = B^{-1}b$ there are at least three alternatives.

i, First form the $2n_c m_2 \times 2n_c m_2$-matrix P by essentially performing $2n_c m_2$ semi-circulant solves. Then LU-factorize P. The total cost for this is $\mathcal{O}(m_2^2 m_1 \log_2 m_1 + m_2^3)$. The substitution then basically consists of two semi-circulant solves and a backsubstitution for P at a cost of $\mathcal{O}(m_2 m_1 \log_2 m_1 + m_2^2)$. This gives a direct solver possibly useful for time-dependent problems, where the factorization could be performed *once* prior to the time-marching.

ii, If it is considered too expensive to form P, then approximate P^{-1} by some matrix H^{-1} which is cheaper to form. Then use C as a preconditioner in an iteration, where $C^{-1} = (I - M^{-1}SH^{-1}V^T)M^{-1}$.

iii, When we compute $B^{-1}b$ using eq. (2) we need P^{-1} acting on a vector, $x = P^{-1}z$. If we solve $Px = z$ for x using a CG-like method P need not be formed, since only multiplications with $P = I + V^T M^{-1} S$ are required. This strategy is similar to the partitioned matrix method in domain decomposition.

For problems with variable coefficients in both space directions (not fitting into this framework) the methods above could be used as preconditioners. We use these strategies for some scalar ($n_c = 1$) model problems involving wave propagation. We also apply them to a driven cavity problem solving the Navier-Stokes' equations employing an operator splitting. We compare the performance and memory requirements of the various solution methods.

To analyze the convergence properties of the CG-like methods for the coefficient matrix $M^{-1}B$ or P, the eigensystems of these matrices are examined. For a model problem with coefficient matrix $B = I_{(m_2)} \otimes B_1 + B_2 \otimes I_{(m_1)}$ a thorough analysis has been carried out [2]. In short the result is:

i, $M^{-1}B$ has only $2m_2$ eigenvalues $\{1 + \mu_{1,k}, 1 + \mu_{2,k}\}_{k=1}^{m_2}$ separated from unity.

ii, $M^{-1}B$ has a nonsingular eigenvector matrix $W = (V_2 \otimes I_{(m_1)})\operatorname{diag}(U_1, \ldots, U_{m_2})$, where V_2 is the eigenvector matrix of B_2. For every fixed $k = 1, \ldots, m_2$ all but the first and last columns of U_k are mutually orthonormal. The first and last columns $U_k(:,1)$, $U_k(:,m_1)$ are associated with $\mu_{1,k}$ and $\mu_{2,k}$.

This is intimately related to the eigensystem of P.

i, P has eigenvalues $\{1 + \mu_{1,k}, 1 + \mu_{2,k}\}_{k=1}^{m_2}$, i.e. the same as the nontrivial eigenvalues of $M^{-1}B$.

ii, P has an eigenvector matrix $W_P = V^T W S = V_2 \circ Z$, where $z_{jk} = v_j^T [U_k(:,1), U_k(:,m_1)]$ and $\circ$ denotes the Hadamard product.

Often W_P is less ill-conditioned than W. However, the number of iterations required for convergence is almost the same for both $M^{-1}B$ and P. This once again shows that the convergence of CG-like methods is not completely determined by the condition number of the eigenvector matrix.

[†] This work was supported by the Swedish National Board for Industrial and Technical Development (NUTEK)

References

[1] S. HOLMGREN, K. OTTO,
Iterative solution methods and preconditioners for block-tridiagonal systems of equations,
SIAM J. Matrix Anal. Appl., 13 (1992), pp. 863-886.

[2] S. HOLMGREN, K. OTTO,
Semi-circulant preconditioners for first-order pde,
Submitted to SIAM J. Matrix Anal. Appl.

Iterative Solution of Linear Systems with Low Displacement Rank by Preconditioned Conjugate Gradient-Type Algorithms

Thomas Huckle
Institut für Angewandte Mathematik
Universität Würzburg, F.R.G.

Extended Abstract

In recent years. there has been considerable interest in the iterative solution of systems of linear equations with Toeplitz coefficient matrices by preconditioned conjugate gradient algorithms. Conjugate gradient schemes involve the coefficient matrix of a linear system only in the form of matrix-vector products. Therefore, they are very well suited for the solution of Toeplitz systems, since matrix-vector products with Toeplitz matrices can be computed efficiently using FFTs. Furthermore, appropriately chosen circulant or skew-circulant matrices yield powerful preconditioners for Toeplitz systems.

Toeplitz matrices are a special case of more general families of structured matrices classified by their displacement rank. More precisely, Toeplitz matrices are of displacement rank two. Other classes of non-Toeplitz matrices of low displacement rank arise in important applications. For example, the positive definite coefficient matrices of the normal equations corresponding to Toeplitz least-squares problems are non-Toeplitz matrices of displacement rank four. In this talk, we propose and analyze the use of circulant preconditioners for the iterative solution of systems of linear equations $Ax = b$ with coefficient matrices A of low displacement rank. In the case of symmetric positive definite A, these preconditioners are combined with the classical conjugate gradient algorithm. We also study the case of nonsymmetric linear systems. Here we use the quasi-minimal-residual method (QMR) and its transpose-free variant, the TFQMR algorithm, for the solution of $Ax = b$.

Many recent papers consider the special case of linear equations with a Toeplitz matrix $T_n(t_{n-1},...,t_1;t_0;t_{-1},...,t_{1-n}) := (t_{i-j})_{i,j=1}^n$ connected with an l_1-sequence $(t_j)_{j=-\infty}^{\infty}$, $\sum_{j=-\infty}^{\infty}|t_j| \leq M < \infty$. The Toeplitz structure can be generalized in the following way. Let us denote a lower Toeplitz matrix L_n connected with an l_1-sequence $(l_j)_{j=0}^{\infty}$ by $L_n(l_0,...,l_{n-1}) := T_n(l_{n-1},...,l_1;l_0;0,...,0)$, and an upper Toeplitz matrix connected with an l_1-sequence $(u_j)_{j=0}^{\infty}$ by $U_n(u_0,...,u_{n-1}) := T_n(0,...,0;u_0;u_1,...,u_{n-1})$. Then a displacement rank matrix A of positive displacement rank $\alpha_+ = k$ is given by

$$A_n = \sum_{j=1}^{k} L_n^{(j)} U_n^{(j)}$$

with lower and upper Toeplitz matrices. Similarly a matrix with negative displacement rank $\alpha_- = k$ can be defined by reversing L and U. Note that Toeplitz matrices have

displacement rank two. Here, we will consider only matrices with positive displacement rank; the other case can be treated in the same way. Furthermore, we assume that all occurring Toeplitz matrices are connected with l_1-sequences.

For preconditioning we use circulant matrices, i.e., Toeplitz matrices of the form

$$C_n(c_0,...,c_{n-1}) := T_n(c_1,...,c_{n-1};c_0;c_1,...,c_{n-1}) = F_n^H \Lambda F_n$$

with a diagonal matrix Λ and the Fourier matrix

$$F_n := \frac{1}{\sqrt{n}}\left(w^{-jk}\right)_{j,k=0}^{n-1} \ , \ \ w := exp(2\pi i/n) \ .$$

For the following, we also need skew-circulant matrices

$$S_n(s_0,...,s_{n-1}) := T_n(-s_1,...,-s_{n-1};s_0;s_1,...,s_{n-1}) = \Omega F_n^H \Lambda F_n \bar{\Omega}$$

with Λ diagonal and $\Omega := \mathrm{diag}(1,\sigma,...,\sigma^{n-1})$, $\sigma := exp(\pi i/n)$.

The so-called optimal circulant preconditioner C for a given matrix A that minimizes $A - C$ in the Frobenius norm is given by $C := F_n^H \, \mathrm{diag}(F_n A F_n^H)\, F_n$. This leads to two different ways to obtain circulant preconditioners for a displacement rank matrix A_n. One possibility is to compute the optimal circulant approximations $C_L^{(j)}$ and $C_U^{(j)}$ to the Toeplitz matrices $L_n^{(j)}$ and $U_n^{(j)}$ in order to define the circulant preconditioner $C_{LU} := \sum_{j=1}^{k} C_L^{(j)} C_U^{(j)}$. For A_n a Hermitian matrix we use the Hermitian part of C_{LU} as preconditioner. On the other hand one can try to determine the true optimal circulant Frobenius norm approximation C_A to A_n. In this case, one has to compute $\mathrm{diag}(F_n LU F_n^H)$. This can be done by using the well known partitioning of a Toeplitz matrix T in the sum of a circulant and a skew-circulant matrix. If the lower and upper Toeplitz matrices in A_n are connected with l_1-sequences then we show that these two preconditioners are asymptotically equivalent in the spectral norm, and that the eigenvalues of $A_n - C_n$ are clustered around 0 if n tends to infinity. Thus, if for example the matrices A_n and A_n^{-1} are Hermitian uniformly positive definite then the number of iterations of the preconditioned conjugate gradient method for computing $A_n^{-1}b$ to a given accuracy is bounded independent of n.

Finally, we present results of numerical experiments with the proposed circulant preconditioners for various symmetric and nonsymmetric linear systems of low displacement rank.

This is joint work Roland W. Freund from AT& T Bell Laboratories.

RELATIVE PERTURBATION TECHNIQUES FOR EIGENVALUE AND SINGULAR VALUE PROBLEMS

ILSE C.F. IPSEN*

Abstract. We present techniques for deriving relative perturbation results for singular values and vectors, as well as for eigenvalues and vectors.

We consider the class of perturbations δA where $A+\delta A$ is congruent to the original matrix A, i.e. $A+\delta A = D^T A D$ for some non-singular D. This includes component-wise relative perturbations, as well perturbations that amount to elimination of off-diagonal blocks (deflation).

The relative perturbation results derived by means of our techniques can be used as deflation and convergence criteria in algorithms for solving singular value and eigenvalue problems of dense or banded matrices. Not only do they guarantee high accuracy, they also enhance efficiency because the potential for break-up into smaller subproblems is recognised as early as possible. This in turn makes our perturbation results highly suitable for the design of accurate parallel divide-and-conquer algorithms, in particular as a means to estimate and control load balance.

We briefly describe our main results. Let M be a real, symmetric matrix and $M+\delta M = D^T M D$ for some non-singular matrix D. If λ_i and λ_i' are the respective ith eigenvalues of M and $M+\delta M$ then

$$|\lambda_i - \lambda_i'| \leq \|\delta M\|$$

is the traditional absolute error bound. It provides useful information only about the relative accuracy of the largest eigenvalues. In contrast, we derive the relative error bound

$$|\lambda_i - \lambda_i'| \leq |\lambda_i|\,\gamma,$$

where $\gamma = \|D^T D - I\|$ and the norm is the two-norm. Because this bound on the relative accuracy is the same for all eigenvalues, it can provide realistic information about the relative accuracy of the smallest eigenvalues.

Furthermore, if

$$M w_i = \lambda_i w_i \quad \text{and} \quad (M+\delta M) w_i' = \lambda_i' w_i',$$

and if θ_i is the angle between the eigenspaces spanned by w_i and w_i' then the traditional result

$$|\sin\theta_i| \leq \frac{\|\delta M\|}{\text{gap}_i - \|\delta M\|}$$

says that the amplification of $\|\delta M\|$ in the bound is inversely proportional to the absolute difference

$$\text{gap}_i = \min_{j\neq i} |\lambda_i(M) - \lambda_j(M)|.$$

[*] This represents joint work with Stanley C. Eisenstat

Again, this bound is too pessimistic for eigenvectors associated with the smallest eigenvalues. In contrast, we derive the bound

$$|\sin\theta_i| \leq \frac{\delta}{\rho_i - \gamma} + \beta,$$

where

$$\rho_i = \min_{j\neq i} |\lambda_i - \lambda_j|/|\lambda_i|$$

is the relative gap and

$$\beta = \|D^{-1}\|\,\|D - I\|, \qquad \delta = \|D^T D\|\,\|D^{-T} D^{-1} - I\|.$$

This bound provides realistic information for all eigenvalues that are well-separated in relation to their magnitude.

Similar results hold for singular values and singular vectors. Let B be a real, possibly rectangular, matrix and $B+\delta B = D_L B D_R$ for non-singular D_L and D_R. If σ_i and σ_i' are the respective ith singular values of B and $B+\delta B$ then we prove

$$\frac{\sigma_i}{\|D_L^{-1}\|\,\|D_R^{-1}\|} \leq \sigma_i' \leq \sigma_i\,\|D_L\|\,\|D_R\|.$$

Also, if

$$B u_i = \sigma_i v_i, \quad \text{and} \quad (B+\delta B) u_i' = \sigma_i' v_i',$$

and if θ_i^u and θ_i^v are the respective angles between the spaces spanned by u_i and u_i', and by v_i and v_i' then we prove

$$\max\{\sin\theta_i^u, \sin\theta_i^v\} \leq 2\left(\frac{\delta}{\rho_i - \gamma} + \beta\right).$$

Most surprisingly, all of the above relative perturbation results can be derived by means of the traditional absolute perturbation results.

We show that the following known results for the class of component-wise relative perturbations represent special cases of our results above:

- bounds on the relative errors in the singular values of bi-diagonal matrices (Demmel & Kahan 90; Deift, Demmel, Li & Tomei 91)
- bounds on the relative errors in the singular values of bi-acyclic matrices (Demmel & Gragg 92)
- relative bounds on the angles between singular vectors of bi-diagonal matrices (Barlow & Demmel 90; Deift, Demmel, Li & Tomei 91)

In addition, we derive a relative bound on the singular vector angles for bi-acyclic matrices.

For the class of perturbations that amount to elimination of an off-diagonal block (deflation) we prove a bound on the relative error in the singular values. If B is a real, possibly rectangular matrix, and $B+\delta B = BD$ for

$$D = \begin{pmatrix} I_k & X \\ & I_{n-k} \end{pmatrix},$$

where I_m is the identity matrix of order m, then

$$|\sigma_i' - \sigma_i| \leq \sigma_i\,\|X\|.$$

The following known results are special cases of this bound:

- bounds on the relative errors in the singular values of bi-diagonal matrices (Demmel & Kahan 90; Deift, Demmel, Li & Tomei 91; Fernando & Parlett 92)
- bounds on the relative errors in the smallest singular values of block triangular matrices (Chandrasekaran & Ipsen 92; Mathias & Stewart 92)

In addition, we derive a bound on the relative error in the singular values of deflated block-triangular matrices. If

$$B = \begin{pmatrix} B_{11} & B_{12} \\ & B_{22} \end{pmatrix} \quad \text{and} \quad B + \delta B = \begin{pmatrix} B_{11} & \\ & B_{22} \end{pmatrix},$$

where B_{11} or B_{22} is non-singular, then

$$|\sigma_i - \sigma_i'| \leq \sigma_i \frac{\|B_{12}\|}{\max\{\sigma_{min}(B_{11}), \sigma_{min}(B_{22})\}}.$$

This means, high relative accuracy of the singular values is preserved after elimination of a 'small' off-diagonal block if at at least one of the diagonal blocks is well-conditioned.

A Direct Method for Reordering Eigenvalues in the Generalized Real Schur Form of a Regular Matrix Pair (A, B)

Bo Kågström
Institute of Information Processing
University of Umeå
S-901 87 Umeå, Sweden

In this talk we will present a *direct* method for reordering eigenvalues in the generalized real Schur form of a regular matrix pair (A, B) [5]. The method performs an orthogonal equivalence transformation of the real matrix pair (A, B), where A is upper quasi-triangular and B upper triangular. (This form can be computed by an orthogonal equivalence transformation using the nZ algorithm.) A quasi-triangular matrix is triangular with possible 2×2 blocks along the diagonal. In the generalized Schur form the 2×2 blocks correspond to pairs of complex conjugate eigenvalues of the real pencil $A - \lambda B$. The real eigenvalues are given by the ratios of the diagonal entries of A and B corresponding to 1×1 diagonal blocks in the generalized Schur form. So the problem of reordering eigenvalues is equivalent to swap 1×1 and 2×2 diagonal blocks along the diagonal of (A, B). Let (A_{11}, B_{11}) and (A_{22}, B_{22}) be matrix pairs of size $m \times m$ and $n \times n$, respectively, where $m, n = 1$ or 2. We want to find orthogonal $(m+n) \times (m+n)$ matrices m and n such that

$$P^T \left(\begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} - \lambda \begin{bmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{bmatrix} \right) Q = \begin{bmatrix} \hat{A}_{22} & \hat{A}_{12} \\ 0 & \hat{A}_{11} \end{bmatrix} - \lambda \begin{bmatrix} \hat{B}_{22} & \hat{B}_{12} \\ 0 & \hat{B}_{11} \end{bmatrix}, \qquad (0.1)$$

where (A_{ii}, B_{ii}) and $(\hat{A}_{ii}, \hat{B}_{ii})$ for $i = 1, 2$ are equivalent matrix pairs with the same eigenvalues but their positions are exchanged (swapped) along the block diagonal of (A, B). Each swap comprises solving for (L, R) in the generalized Sylvester equation [7]:

$$\begin{aligned} A_{11} \cdot R - L \cdot A_{22} &= -A_{12} \\ B_{11} \cdot R - L \cdot B_{22} &= -B_{12} \end{aligned} . \qquad (0.2)$$

Further, m and n are determined from computing orthogonal basis for certain eigenspaces (involving L and R) of the regular pencil in ().

The numerical stability of the direct reordering method will be discussed. Since the method is based on solving of a sequence of generalized Sylvester equations we will also present a perturbation analysis of the generalized Sylvester equation itself, including perturbation and error bounds and an expression for the normwise relative backward error of an approximate solution [6].

We have developed reliable and robust software in the LAPACK-style [1], with guaranteed backward stability, which implements the direct reordering method. The error analysis and numerical experiments show the following characteristics of the direct reordering method:

- It is numerically stable and accurate except for "extremely" ill-conditioned problems. Typically, these problems are related to ill-conditioned, large-normed solutions of the associated generalized Sylvester equation.
- The numerical stability can be guaranteed and controlled by computing the size of the backward error and rejecting a swap if it exceeds a certain threshold. The user can choose between a "weak" or a "strong" stability criterion.
- We can expect "large" changes in individual eigenvalues for ill-conditioned $Ax = \lambda Bx$ problems even if the backward error after the swapping is at the level of machine precision.

  Examples include defective eigenvalues, notably at infinity. This type of inherited ill-conditioning cannot be "cured" by any reordering method. However, one possible remedy is to start to deflate the infinite eigenvalues with a staircase type of algorithm (e.g., see [2, 3]) and then perform the required reordering of the finite eigenvalues. The placement of the infinite cluster could be made either to the $(1, 1)$-block or to the $(2, 2)$-block.

Our practical algorithm will be compared experimentally with an iterative method based on the QZ iteration [9]. Besides, we have implemented a block solver of the generalized Sylvester equation with Dif^{-1} estimators based on the work in [7]. These routines form the basis for practical estimation of condition numbers for the generalized eigenvalue problem $Ax = \lambda Bx$, where (A, B) is a regular matrix pair [4, 2, 3, 8], which also will be illustrated.

This is joint work with Peter Poromaa, University of Umeå.

References

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen. *LAPACK Users' Guide.* Society for Industrial and Applied Mathematics, Philadelphia, 1992.

[2] J. Demmel and B. Kågström. The generalized Schur decomposition of an arbitrary pencil $A - \lambda B$: robust software with error bounds and applications. Part I: Theory and Algorithms. *To appear in ACM Trans. Math. Software.* Also published as Report UMINF-91.22, 1991.

[3] J. Demmel and B. Kågström. The generalized Schur decomposition of an arbitrary pencil $A - \lambda B$: robust software with error bounds and applications. Part II: Software and Applications. *To appear in ACM Trans. Math. Software.* Also published as Report UMINF-91.23, 1991.

[4] J. Demmel and B. Kågström. Computing stable eigendecompositions of matrix pencils. *Lin. Alg. Appl.*, 88/89:139–186, April 1987.

[5] B. Kågström. A Direct Method for Reordering Eigenvalues in the Generalized Real Schur Form of a Regular Matrix Pair (A,B). Report UMINF-92.18, Institute of Information Processing, University of Umeå, S-901 87 Umeå, Sweden, 1992.

[6] B. Kågström. A Perturbation Analysis of the Generalized Sylvester Equation. Report UMINF-92.17, Institute of Information Processing, University of Umeå, S-901 87 Umeå, Sweden, 1992.

[7] B. Kågström and L. Westin. Generalized Schur methods with condition estimators for solving the generalized Sylvester equation. *IEEE Trans. Autom. Contr.*, 34(4):745–751, 1989.

[8] G. W. Stewart and J.-G. Sun. *Matrix Perturbation Theory.* Academic Press, New York, 1990.

[9] P. Van Dooren. ALGORITHM 590: DSUBSP and EXCHQZ: Fortran routines for computing deflating subspaces with specified spectrum. *ACM Trans. Math. Software*, 8:376–382, 1982. and (Corrections) Vol. 4, No. 4, p 787, 1983.

THE K-CONDITION NUMBER AND ITS USE FOR THE CONSTRUCTING OF PARALLELIZABLE PRECONDITIONED CONJUGATE GRADIENT SOLVERS

I.E.Kaporin (Russia)

1. Let H be an SPD matrix of order n. Define the K-condition number as

$$K(H) = (trH/n)^n/detH .$$

Qualitatively, this matrix function $K(H)$ has much in common with the standard condition number,

$$C(H) = \lambda_{max}(H)/\lambda_{min}(H) ,$$

and can be efficiently used instead of $C(H)$ for many purposes.

2. In particular, see [2,3], for the number of iterations k of the conjugate gradient (CG) method applied to the solution of $Hx = b$, the estimate

$$k \le \lceil \log_2 \frac{K(H)}{\varepsilon} \rceil$$

holds, which can be more appropriate than the standard one,

$$k \le \lceil \frac{1}{2} \sqrt{C(H)} \log_e \frac{2}{\varepsilon} \rceil ,$$

when choosing a preconditioning strategy or even for estimating the actual CG iteration number.

3. Hence, the choice of a preconditioning can be subjected to the requirement of the reduction of $K(H)$ rather than $C(H)$. Here we concentrate on the case of left preconditioned matrix

$$H = G^TG\ A,$$

where A is the original SPD coefficient matrix and G is a sparse lower triangular Incomplete Inverse Cholesky (IIC) factor of A. If the sparsity pattern of G is fixed, a simple formula for G that minimizes $K(H)$ exists [1-3], as opposed to the case of mininizing $C(H)$ [5], when no computationally feasible procedures are available.

For FD- or FE-type sparse matrices A quite appropriate IIC sparsity patterns can be obtained using special reorderings, such as Domain-Decomposition/Multicolor ones [2]. In such cases, $nz(G + G^T)$ can be much smaller than $nz(G^TG)$ which is essential for overall efficiency of IIC-CG algorithms, see S.5 below.

4. Since the resulting preconditioned CG methods employ only simple vector and matrix-vector operations, several families of highly parallel solution procedures are essentially defined.

5. Several results concerning the performance limitations of the entire family of the "explicit" iterative methods are also presented.

Theorem [4]. *Let $J_i^{(k)}$ be the subset of indices of nonzero entries of the vector $(HA)^k e_i$, where e_i is the i^{th} unit vector, A is an SPD matrix, and the diagonal of HA is nonzero. Then the lower bound*

$$\min_{\substack{\deg P \le k \\ P(0) = 1}} \| P(HA) \|_A^2 \ \ge\ 1 - e_i^T(A_i^{(k)})^{-1}e_i / e_i^TA^{-1}e_i$$

holds, where $A_i^{(k)}$ is obtained from A by replacing all its rows and columns having indices not contained in $J_i^{(k)}$ by those of the identity matrix.

Using this theorem with several test matrices A, one can obtain some non-trivial lower bounds on the number of iterations and/or condition numbers for such families of methods as SOR- and SSOR-like ones, as well as for CG-like methods using sparse preconditioner H. In particular, several hypotheses presented in [5] can be proven using these results. Moreover, this theorem shows that explicitly preconditioned CG methods may perform satisfactorily only with rather filled preconditioners H, which fact explains the importance of using the factored form $H = G^TG$.

References

1. Kaporin I.E. A preconditioned conjugate-gradient method for solving discrete analogs of differential problems. Diff. Equat., 1990, v.26, no.7, pp.897-906.
2. Kaporin,I.E. New convergence results and preconditioning strategies for the conjugate gradient method. To appear in J. of Numer. Lin. Alg. with Appls., 1992.
3. Kaporin, I.E. Two-side explicitly preconditioned conjugate gradient method for the solution of nonsymmetric linear systems. To appear in Int. J. of Computer Math., 1992.
4. Kaporin, I.E. Efficiency bounds for several iterative schemes. Moscow, 1986. (Russian)(An unpublished manuscript.)26p.
5. Greenbaum, A., and G.H. Rodrigue. Optimal preconditioners of a given sparsity pattern. BIT, 1989, v.29, pp.610-634.

Large Structured Problems

Linda Kaufman
AT&T Bell Laboratories
Murray Hill, NJ 07974

Much of the work in numerical linear algebra for large systems concentrates on the zero structure of the system. There are some examples, notably multigrid and fast direct methods for separable partial differential equations, where asking from whence the problem came and exploiting its algebraic structure has paid off handsomely. In this talk we will explore some additional examples of this type.

The first example is a linear least squares problem coming from system identification in which each column can be written as a tensor product of 2 vectors. Thus a problem with $m \times p$ rows and n columns actually comes from two matrices one having m rows and the other p rows. By doing QR decompositions on these 2 matrices independently and then forming the original system one can reduce the problem to one in which the only the first i^2 elements of the ith column are nonzero.

A similar problem arises in separable nonlinear least squares with multiple right hand sides. Golub and Leveque have shown for that problem one can solve a linear least squares with a Jacobian that has special structure. Each right hand side corresponds to a block of rows and the jth column of the ith block has the form $M_j y_i$ where each M_j usually has only 1 or 2 nonzero columns. One can use this structure to quickly obtain a QR factorization of the whole matrix. The trick involves concatenating all the nonzero columns of all the M_j matrices to form a dense matrix G. If each M_j has only one nonzero column then by doing a QR decomposition of G and one of Y whose ith column is y_i, then we are back to the problem defined in the previous paragraph. Slightly more complicated M_j matrices will lead to slightly more complicated final matrices but they will usually be almost triangular. Note that all this structure can be exploited if one chooses to form the normal equations.

A nonnegative linear least squares problem arises in the reconstruction problem in positron emission tomography. Typically there are millions of nonzero elements in the matrix but fortunately it is composed of repeated submatrices that decreases the storage and affords the use of parallel computation. One can attack the problem with a preconditioned conjugate gradient algorithm where the preconditioner gives an approximate distance of the variable to the bound. The preconditioner allows one to travel further without hitting a constraint in the space of the larger variables, which physically are the most important. Because the problem lies on a grid the techniques of multigridding and adaptive gridding in pdes are applicable but they must be applied with caution because the solution has steep fronts and because the noise in the data tends to produce a noisy solution.

A New Approach to Condition Estimation

Charles Kenney

Efficient estimation of matrix norms has long been a central problem in condition theory, especially for situations where

1) The matrix M in question is not known explicitly or is too expensive to compute directly.

2) Products of the form Mv where v is a vector can be formed relatively easily.

3) Transpose products of the form $M^T w$ where w is a vector can also be formed relatively easily.

As a familiar example, if M is the inverse of a matrix A, then we may only know M implicitly through the LU factorization of A but the matrix-vector products in 2) and 3) can be found by using the LU factors to solve either $Ax = v$ or $A^T y = w$. In a more general setting, we may wish to measure the sensitivity of a function F that maps matrices into matrices by estimating the norm of the linearization DF of F about a matrix X. In this case, M is the Kronecker form of DF and the matrix-vector products of the form $Mv = vec(DF(Z))$ can be estimated via

$$DF(Z) = (F(X + \delta Z) - F(X))/\delta \; + \; O(\delta) \, , \qquad (1)$$

where δ is "small" and $v = vec(Z)$. If F is smooth and maps square matrices into square matrices of equal dimension, then the transpose products $M^T w$ can be formed by replacing X by X^T as described in [1].

If the three conditions listed above are met, the norm of M can be efficiently estimated via the power method: given an initial vector v_0 define $w_1 = Mv_0$ and $v_1 = M^T w_1$. Unless the initial vector is poorly chosen the 2-norm ratio $\|v_1\|/\|w_1\|$ provides a good estimate of the 2-norm of M. Better estimates can be obtained by restarting the power iteration with v_0 replaced by v_1. (See [2] for a statistical analysis of the performance of the power method for random initial vectors.)

Unfortunately, the transpose step is the Achilles' heel of the power method for the problem of estimating the sensitivity of functions that map between spaces of different dimensions. As an illustration, if F maps $I\!R^n$ into $I\!R$ then DF is just the gradient of F and products Mv in 2) are easily approximated by (1). However, the transpose step 3) requires knowing each component of the gradient. Difference estimates of the individual entries of the gradient would require n extra function evaluations and hence the transpose step is impractical.

Because of this problem, a new form of condition estimation has been developed [3] which drops the transpose requirement 3) and only assumes that matrix products of the form Mv in 2) can be obtained at a reasonable cost. (By (1) this cost should be no more than the cost of one extra function evaluation, and for many problems is considerably less, especially if the function can be evaluated via a Newton method.)

Somewhat surprisingly the statistical theory associated with the norms of vectors of the form Mv for random vectors v can be worked out in great detail and a rather complete theory derived that predicts the accuracy of norm estimates for M from just a few matrix-vector products.

This theory is based on the distribution of inner products between a fixed vector $v \in I\!R^n$ and randomly selected unit vectors z. The random variable $\zeta = |v^T z|$ is easy to work with analytically because ζ^2 has a beta distribution. In particular, the expected value of ζ is equal to $E_n \, \|v\|$ where E_n is a constant depending only on the dimension n. From this an exact expression can be derived for the probability that ζ / E_n lies within a given factor ω of $\|v\|$. This probability is given approximately by the expression

$$\Pr\left(\frac{\|v\|}{\omega} \le \frac{\zeta}{E_n} \le \omega \, \|v\| \right) \approx 1 - \frac{2}{\pi \omega} \, . \qquad (2)$$

By taking more than one inner product, say $v^T z_1, \ldots, v^T z_m$, we obtain an m^{th} order estimate for $\|v\|$. That is, the probability of a bad estimate (off by more than a factor ω) is less than a constant divided by ω^m. For example, with two inner products the chance of a bad estimate is approximately $\frac{\pi}{4\omega^2}$. Thus only a few inner products are needed to render the possibility of a bad estimate for the norm of v very small indeed.

This procedure can be extended to estimate the Frobenius norm of a matrix M with just a few matrix-vector products Mz_1, $Mz_2, \ldots$ The basic idea here is that each entry of the product Mz is an inner product between a row of M and z, so the preceding theory can be applied. Combining the estimates of the norms of the rows of M gives an estimate of the Frobenius norm of M, but the analysis of how well this estimate approximates $\|M\|_F$ is not as easy as in the vector case. A conjecture is presented to the effect that the probability of a bad estimate in the matrix case is no worse than the probability of a bad estimate in the vector case. This is strongly supported by numerical evidence and a conservative form of the conjecture is proved in [4]. The material in this talk represents joint work with Alan Laub and Thorkell Gudmundsson.

References:

[1] Kenney, C.S., and A.J. Laub, "Condition Estimates for Matrix Functions," *SIAM J. Matrix Anal. Appl.*, 10(1989), pp.191–209.

[2] Kuczyński, J., and H. Woźniakowski, "Estimating the Largest Eigenvalue by the Power and Lanczos Algorithms with a Random Start," *SIAM J. Matrix Anal. Appl.*, 13(1992), pp.1094–1122.

[3] Kenney, C.S., and A.J. Laub, "Small-Sample Statistical Condition Estimates for General Matrix Functions," *SIAM J. Sci. Comp.*, to appear, 1993.

[4] Gudmundsson, T., C.S. Kenney, and A.J. Laub, "Small-Sample Statistical Estimates for Matrix Norms," *SIAM J. Matrix Anal. Appl.*, submitted, Jan. 1993.

Restarted Arnoldi Procedure and Eigenvalue Translation Technique for Solving Large Sparse Automatic Control Problems

S.A. Kharchenko
Russian Academy of Sciences & Elegant Mathematics, Inc. (USA)
E-mail: kharchen@sms.ccas.msk.su

In this talk we consider the problem of constructing efficient and numerically stable algorithms for solving Eigenvalue Assignment problems with sparse large-scale matrices: for a given square matrix $A \in \mathcal{R}^{n \times n}$ and a given multiple input $b \in \mathcal{R}^{n \times \ell}$ find a feedback matrix $f \in \mathcal{R}^{n \times \ell}$ such that the matrix $A^T + bf^T$ has all its eigenvalues in the left-half plane. Existing algorithms for solving this problem are not suitable for sparse large-scale matrices since they require either the computation of the exact eigenspaces of the matrix associated with eigenvalues to be assigned or the reduction of the entire matrix to the block Hessenberg form which fully destroys the sparsity structure. Moreover, the larger is the size of the problem to be solved the more unstable are the existing algorithms.

We consider the following algebraic reformulation of the original problem: find a feedback matrix f such, that the matrix $A^T + bf^T$ has all its eigenvalues in the desired complex domain K of the complex plane. To solve this problem we exploit eigenvalue translations [1] which deal with only approximations to the eigenspaces corresponding to the eigenvalues to be assigned computed at several restarted Arnoldi cycles. It enables us to preserve automatically the sparsity structure of the original matrix since in this case the Eigenvalue Assignment problem is reduced to a sequence of Eigenvalue Assignment problems of smaller sizes exploiting only a procedure for multiplying a matrix by a vector. In order to maintain the numerical stability during calculations we construct similarity transformation of the transformed matrix after every restarted Arnoldi cycle. This transformation preserves the spectrum, assigned by the Arnoldi procedure, minimizes the largest singular value of the transformed matrix and can be computed without destroying the sparsity structure of the original matrix since only a small number of multiplications of the matrices A and A^T by a vector is required.

The suggested algorithm for the single-input case $\ell = 1$ can be described as follows. Let A_i^T be the iteration matrix at the i-th global iteration of the algorithm and b_i be the corresponding single input while $A_1^T = A^T$ and $b_1 = b$. In order to construct feedback vector f_i we perform k iterations of the Arnoldi procedure with the matrix A_i and initial vector $v_1 = b_i/\|b_i\|$. Thus we have the equalities:

$$\left\{ \begin{array}{lcl} A_i V_k & = & V_k H_k + \beta v_{k+1} e_k^T, \\ V_{k+1}^T V_{k+1} & = & I_{k+1}, \end{array} \right. \qquad (1)$$

where $V_k = \{v_1, ..., v_k\}$. From matrix equalities (1) we can compute approximations to the eigenpairs of the matrix A_i and norm of the corresponding eigenresiduals. If the spectrum of the matrix H_k lies in the prescribed convex domain K of the complex plane than, by the approximation properties of the Arnoldi procedure, we stop global iterations of the algorithm.

The feedback vector f_i we seek in the form

$$f_i = X_U w. \qquad (2)$$

Here $X_U \in \mathcal{R}^{n \times m}$ is a real matrix which is a unitary transformed matrix of approximations to eigenvectors from the eigenpairs to be assigned while $w \in \mathcal{R}^m$ is to be chosen to assign required eigenvalues. We emphasize that we do not require to assign all eigenvalues at the current global iteration of the algorithm.

Controllability conditions for a feedback vector f_i from (2) have the form

$$\| r_j(A_i) \| < \frac{\delta_j}{2} \mid ctg\, \widehat{b_i, x_j} \mid, \qquad (3)$$

where $r_j(A_i)$ are the eigenresiduals of the eigenpairs to be assigned, x_j are approximations to the corresponding eigenvectors while δ_j are some positive coefficients depending on the stability properties of the eigenspaces to be assigned.

To show that we actually assign eigenvalues of the matrix A_i^T by the transformation $A_i^T + b_i f_i^T$ we prove the numerical stability of eigenvalue assignments, i.e. when $\| r_j(A_i) \|$ tend to zero for every eigenpair from the group to be assigned then the radii of the circles containing assigned and perturbed eigenvalues also tend to zero. It means that the algorithm gives a solution to the algebraic formulation of the Eigenvalue Assignment problem.

Iterations of the presented algorithm without stabilization step may be unstable due to an increase of the largest singular value of the matrix after every transformation. We construct the similarity transformation of the form $I_n + u_i v_i^T$ in order to minimize the maximal singular value of the iteration matrix at the next global iteration of the algorithm

$$A_{i+1}^T = (I_n + u_i v_i^T)(A_i^T + b_i f_i^T)(I_n + u_i v_i^T)^{-1}.$$

The single-input vector b_{i+1} at the next global iteration of the algorithm will be of the form

$$b_{i+1} = (I_n + u_i v_i^T) b_i.$$

Obviously the solution f to the original Eigenvalue Assignment problem can be quite easily restored from the solution to the transformed Eigenvalue Assignment problem.

As an application of the suggested algorithm consider now the problem of finding a feedback stabilization f for the second order differential equation

$$M_N \ddot{q}_N + G_N \dot{q}_N + K_N q_N = B_N u$$

with the first order realization

$$\dot{x} = \begin{pmatrix} 0 & I \\ -M_N^{-1} K_N & -M_N^{-1} G_N \end{pmatrix} x + \begin{pmatrix} 0 \\ -M_N^{-1} B_N \end{pmatrix} u.$$

This problem can be solved directly by our algorithm. In this case we need no inverting of the matrix M_N since in our algorithm only a procedure for multiplying a matrix by a vector is needed. At every global iteration of the algorithm we must only solve systems with matrices M_N and M_N^T.

The results of numerical experiments with large sparse matrices are presented which can not be reproduced by the existing algorithms.

References

[1] S.A. Kharchenko and A.Yu. Yeremin, *Eigenvalue translation based preconditioners for the GMRES(k) method*, Research Report EM-RR-2/92, Elegant Mathematics, Inc.(USA), 1992.

[2] S.A. Kharchenko, A.Yu. Yeremin and N.L. Zamarashkin, *Restarted Arnoldi Procedure and Eigenvalue Translation Technique for Solving Large Sparse Automatic Control Problems*, Research Report EM-RR-8/93, Elegant Mathematics, Inc.(USA), 1993.

Computational Kernels for Iterative Methods

Recent work here and elsewhere has focused on the establishment of a set of computational kernels for solving sparse linear systems by iterative methods. Many iterative algorithms can be decomposed into a relatively small set of basic computational operations. Since these are the most computationally intensive parts of the code, it is possible to develop efficient and portable implementations of iterative algorithms by writing them in terms of these basic building blocks. Such computational kernels are particularly advantageous for developing software for use on various high performance computers. The development of parallelizable computational kernels is particularly complicated. Several different researchers have begun to write and test software for computational kernels. These approaches will be compared and contrasted.

David R. Kincaid
Center for Numerical Analysis
University of Texas at Austin
Austin, TX 78713–8510

Incomplete Block SSOR Preconditionings for p-Adaptive Three-Dimensional FE Systems

L.Kolotilina
Steklov Mathematical Institute
Fontanka 27, 191011 St.Petersburg, Russia
E-mail: liko@lomi.spb.su

The p-version of the Finite Element Method (FEM) is considered to be a promising approach to solving many problems of structural mechanics. To reduce the size of the resulting linear system ensuring the desired accuracy various p-adaptive strategies of constructing a FE approximation are exploited.

A symmetric positive definite coefficient matrix of the resulting linear system can be naturally presented in the following two-by-two block form

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \tag{1}$$

where the first diagonal block A_{11} corresponds to the "old" degrees of freedom while the second block A_{22} corresponds to the "new" degrees of freedom resulting from a p-refinement.

In the three-dimensional case the matrix A may be quite ill-conditioned, relatively densely populated (depending on p), and the second block A_{22} has in general an irregular sparsity pattern. However, the considered block partitioning of A possesses two algebraic properties which are natural to be exploited when constructing a preconditioner for A:

(1) the "new" block A_{22} is well-conditioned and

(2) the matrix $A_{11}^{-1}(A_{11} - A_{12}A_{22}^{-1}A_{21})$ is well-conditioned.

The second property ensures that the block SSOR matrix corresponding to the block partitioning (1) would be a good preconditioner for A. However, this approach is not practically feasible because of the solution of linear systems with A_{11} and A_{22} it requires.

We propose to apply instead an incomplete block SSOR preconditioning [2], where the matrices A_{11} and A_{22} are replaced by some approximations $\tilde{A}_{11}$ and $\tilde{A}_{22}$ to them.

The approximation $\tilde{A}_{22}$ to A_{22} is constructed using the diagonally compensated reduction of positive off-diagonal entries [1] taking into account

property (1). The construction of $\tilde{A}_{11}$ is essentially based on the results in [2], where block SSOR preconditionings corresponding to a superelement partitioning of the original FE mesh for a 3D high-order FE system are considered.

The described preconditionings are analyzed theoretically and provided with the results of numerical experiments for 3D linear elasticity problems for orthotropic materials.

References

[1] O. Axelsson and L. Kolotilina. Diagonally compensated reduction and related preconditioning methods. *J. of Numer. Linear Algebra with Appl.*, to appear.

[2] L.Yu. Kolotilina, I.E. Kaporin and A.Yu. Yeremin. Block SSOR preconditionings for for high order FE systems II: Incomplete BSSOR preconditionings. *Linear Algebra Appl.*, 154-156:647–674, 1991.

Image Processing and Compression Using Wavelets

C.-C. Jay Kuo

Department of Electrical Engineering-Systems
University of Southern California
Los Angeles, California 90089-2564
Phone: (213) 740-4658, Fax: (213) 740-4651
E-mail: cckuo@sipi.usc.edu

The wavelet and wavelet packet transforms provide a new multiresolution signal analysis tool, and have received a lot of attention recently. The effective application of this new tool to engineering problems requires insights into the nature of the problems and novel ideas in processing the data. I will focus on the application of wavelets to image processing problems in this talk. Basic concepts of wavelet theory will be first reviewed. Then, I will use three examples to illustrate the advantages provided by the new wavelet approach, i.e. image compression, texture analysis and planar curve representation as detailed below.

1. Image Compression

Image compression methods based on a multiresolution approach have been studied over the last ten years. The major advantage of the multiresolution approach is that it provides a graceful degradation between image quality and compression ratio and is hence suitable for progressive transmission. The first multiresolution compression method, which is usually known as the Laplacian pyramid scheme, was proposed by Burt and Anderson. The basic idea is to decomposition an image into a low resolution image by lowpass filtering and a detailed image which is the difference of the original image and the low resolution image. By recursively performing the decomposition for the lower resolution images, we obtain a sequence of detailed images of different resolutions which can be encoded separately. With recently developed wavelet theory, the application of wavelet transform to image data compression has been considered by many researchers.

We present a new method for image compression based on a modified wavelet transform called the full wavelet transform (FWT) in this research. With the FWT, we first apply the two-scale wavelet decomposition to the original image and obtain 4 subimages. Then, we apply the two-scale wavelet decomposition to all 4 decomposed subimages and obtain 16 subsubimages. The procedure is performed recursively until a desired level is reached. Thus, an image is decomposed into small blocks of the same size via FWT, where each block corresponds to a particular frequency band (or channel) whereas each transform coefficient in the blocks corresponds to a local spatial region in the original image. We observe experimentally that energy compaction is achieved in both the spatial and frequency domains via FWT. The energy compaction property can be effectively utilized to achieve high image compression ratio while preserving good image quality.

The relationship between our proposed algorithm and three other popular compression schemes, i.e. the DCT (Discrete Cosine Transform), PWT (Pyramidal Wavelet Transform), and SBC (SubBand Coding) schemes will be discussed. The performance of these algorithms will be compared.

2. Texture Classification and Segmentation

Textures provide important characteristics for surface or object identification from aerial or satellite photographs. Texture segmentation is important in applications, say, distinguishing the boundaries of different surfaces such as land, sea, forest, farms with agriculture, etc. Most texture classification and segmentation algorithms have been traditionally developed by considering the statistical property of image pixels in a local region. Motivated by evidences from physiology and psychophysics, a recent approach to characterizing textured images is to decompose images into several channels with different spatial frequencies and orientations and analyze the properties of each channel. The approach is generally known as the multichannel texture analysis. The resulting methods, which include the Gabor filter, the multichannel filter-bank decomposition, and the wavelet packet transform, often outperform traditional methods.

In this presentation, we will review the multichannel texture classification algorithms, point out their relationship and make a thorough performance comparison. The methods compared include the DCT, DST, DHT, Gabor filters, Laws filters, wavelet and wavelet packet transforms. The robustness of the algorithms with respect to noise and their discriminant capability of similar textures will also be illustrated by experiments. We will also propose an algorithm using both the wavelet packet transform and the hierarchical fuzzy clustering technique for texture segmentation.

3. Multiscale Planar Curve Descriptor

Effective representation of planar curves is crucial for shape description and recognition and has many applications in image analysis and understanding. Traditional planar curve descriptors include methods based on the Fourier transform and the scale-space filtering approach. We have recently developed a multiscale descriptor which extracts components of curves by using the biorthogonal wavelet transform. With this descriptor, we can decompose a curve into components of different scales so that coarser scale components carry basic information while finer scale components contain detailed information. The biorthogonal wavelet transform restores smooth curves more effectively than traditional compactly-supported orthogonal wavelet transforms such as the well known Haar or Daubechies bases. We show that the reconstructing filters and their duals, the analysis filters, form a perfect reconstruction digital filter banks. Thus, the proposed new filter bank provides a lossless multiscale decomposition/reconstruction scheme to represent planar curves. Since the decomposed data provide a hierarchical representation of planar curves, they can be effectively used in hierarchical matching and object recognition, progressive rendering of graphics, and motion detection.

We will analyze the performance of various wavelet bases by examining their regularity and vanishing moments. We will illustrate two different applications of the proposed wavelet curve descriptor, i.e. character recognition and fast deformation of curves. The performance of the new descriptor is compared to that of other types of descriptors such as descriptors based on the Fourier transform and the scale-space filtering approach in numerical experiments.

Results on Nested Classical Iterative Methods

Paul Lanzkron
Department of Computer Science
Duke University

We consider the solution of algebraic linear systems of the form

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

where A is a matrix and x and b are vectors. Relaxation methods for solving this problem are defined by splitting the matrix A as $A = M - N$, and solving the system

$$Mx_{k+1} = b + Nx_k \tag{2}$$

at each step k, from some initial guess x_0. Generally M is chosen to be easily invertible. We will consider the situation where M is not necessarily easy to invert. This situation might arise in block Gauss-Seidel where the diagonal blocks become very large. In this case we will consider solving the system $Mv = g$ by an iterative method, which might also require the solution of a system of equations done by an iterative method. We call this a nested iterative method.

Two-level iterative methods were first investigated by Nichols [?]. Two-level iterative methods are distinguished from nested iterative methods because the former does not consider that at each step of an iterative solution of $Mv = g$ the system

$$Fv_{j+1} = g + Gv_j, \tag{3}$$

where $M = F - G$, could also be solved iteratively. We call equation (3) the inner iteration and equation (2) the outer iteration. Nichols showed that if the inner iteration were convergent and the outer iteration were convergent then there was some finite number, p, such that if $P > p$ inner iterations were done at each outer iteration then the whole iteration would converge to the solution of (1).

The iterative methods considered by Nichols did not account for a large class of problems arising from Gauss-Seidel type iterations. If A is split as $A = D - L - U$ and the iteration

$$Dx_{k+1} = b + Ux_k + Lx_{k+1} \tag{4}$$

is performed, the splitting is given by $M = D - L$ and $N = U$. Note, however that to solve (4) by an iterative method the matrix D is split and not the matrix M.

We derive results for the case when A is an M-matrix. To derive these results we demonstrate the itertion matrix for the Gauss-Seidel iteration described above. We will also show that this iteration implies an induced splitting of A. This induced splitting of A under the conditions that $A = M - U$, where $M = D - L$, and $D - L$ are convergent regular splittings and $D = F - G$ is a convergent weak regular splitting is itself a weak regular splitting [?]. We then recall that a weak regular splitting of an M-matrix is convergent. Under these same conditions it can also be shown that increasing the number of inner iterations does not decrease the convergence rate of the iterative method.

These results showed that if A were an M-matrix the iteration would converge for any fixed number of inner iterations. We then showed that under somewhat tighter restrictions the iteration would converge even if the number of inner iterations were changed at every

outer iteration. We can also show that if A is an M-matrix both multiplicitive and additive Schwarz iterations converge if the inner iteration comes from a weak regular splitting. This extends results of [?].

It is well known that the iteration

$$x_{k+1} = b + \mu_0 Tx_k + \mu_1 x_k + \mu_2 x_{k-1}, \tag{5}$$

where T is a $n \times n$ matrix and μ_i are scalars, converges if $T \geq 0$ and

$$\frac{1 - \mu_1 - \mu_2}{\mu_0} > \rho(T),$$

$\rho(T)$ the spectral radius of T and μ_0, μ_1, and $\mu_2 > 0$. We give an alternative proof of this by deriving an iteration matrix for the method and showing that the induced splitting is convergent. We give this as an illustration of the power of induced iteration matrices.

The results discussed to this point require the outer iteration to be a regular splitting. It is well know that SOR does not give a regular splitting. Thus the results presented above do not apply to the case when the outer iteration is SOR and the inner iteration comes from a weak regular splitting. We have partially extended the classical results[?, ?], but have not completely solved the problem. We will present these results.

We will conclude the talk with a discussion of practical results from nested iterative methods. An interesting observation we made was that determining the optimal number of inner iterations is very complex if nothing is known about the system. We found, however, that near optimal (and sometimes suboptimal) results can be obtained if the number of inner iterations to be done for the next few, say 5, outer iterations was chosen randomly. We conjecture that this result may be due to the iteration matrix changing at every fifth iteration, and that therefore the residual vector does not converge to the eigenvector associated with the largest eigenvalue of a particular iteration matrix.

These practical results will also include some discussion of our implementation of nested iterative methods on a BBN butterfly, and on a TMC CM5 currently at Duke University.

@article nichols, author = "Nancy K. Nichols",title="On the Convergence of Two-Stage Iterative Processes for Solving Linear Equations", volume=10,pages="460-469",journal="SIAM Journal on Numerical Analysis",year=1973 @book ortega-rheinboldt, author = "J.M. Ortega and W.C. Rheinboldt", title = "Iterative Solution of Nonlinear Equations in several variables",publisher="Academic Press", address="New York and London",year=1970

@article rodrigue, author = Garry Rodrigue and Kang LiShan and Liu Yu-Hui, title = Convergence and Comparison Analysis of Some Numerical Schwarz Methods, journal =Numerische Matematik, volume = 56, year=1989, pages=123-138

@article kulisch, author = U. Kulisch, title = Über reguläre Zerlegungen von Matrizen und einige Anwendungen, journal =Numerische Matematik, volume = 11, year=1968, pages=444-449

@book varga, author ="Richard S. Varga", title ="Matrix Iterative Analysis", publisher="Prentice-Hall",address="Englewood Cliffs, New Jersey", year=1962