Mathematics in signal processing

Based on the proceedings of a conference organized by the Institute of Mathematics and its Applications on Mathematics in Signal Processing, held at University of Bath in September 1985

Edited by

T. S. DURRANI
University of Strathclyde, Scotland

J. B. ABBISS
Spectron Development Labs, Inc., USA

J. E. HUDSON
University of Technology, Loughborough

R. N. MADAN
Office of Naval Research, USA

J. G. McWHIRTER
Royal Signals and Radar Establishment

T. A. MOORE
British Broadcasting Corporation

CLARENDON PRESS · OXFORD · 1987

De Moor B., Vandewalle J. *An adaptive singular value decomposition algorithm based on generalized Chebyshev recursion.* Proc. of the Conference on Mathematics in Signal Processing, p.607-635 , Univ. of Bath, Bath (GB). 17-19 September 1985 (Eds. T.S.Durrani, J.B.Abbiss, J.E.Hudson, R.N. Madan, J.G.McWhirter, T.A.Moore), Clarendon Press, Oxford, 1987.

AN ADAPTIVE SINGULAR VALUE DECOMPOSITION ALGORITHM BASED ON GENERALIZED CHEBYSHEV RECURSIONS

B. De Moor* and J. Vandewalle
(ESAT, Electrical Engineering Department,
Katholieke Universiteit Leuven, Belgium)

ABSTRACT

In this paper, two algorithms for singular value decomposition are studied in a unifying framework: the power method and the Chebyshev iteration method.

They allow to compute adaptively the partial SVD (dominant, smallest or intermediate triplets) of matrices that are slowly time varying and perturbed by noise. The only routines used are matrix vector multiplication and orthogonalization procedures. Some original convergence expressions are derived and convergence rates of both methods are compared. Some backward error theorems permit the control of the convergence level with respect to the noise level.

1. INTRODUCTION: SCOPE OF THE PAPER

Over the past few years, the singular value decomposition of matrices has become a very reliable and numerically robust tool in the analysis and design of algorithms in numerical linear algebra and is used, as an efficient matrix decomposition method, in the solution of many numerical problems (linear equations, identification and realization, model reduction, algorithm analysis, image and signal processing, ...).

In this paper the power method for the singular value decomposition is studied followed by a modification of this classical technique, which is based upon the properties of Chebyshev polynomials. It will be demonstrated that this algorithm is very well suited for a special class of applications, where other algorithms are likely to perform less efficiently. A geometric and numerical convergence analysis of both power and Chebyshev method will be presented

* Sponsored by the IWONL.

in a unifying approach, since it will be demonstrated that the Chebyshev method is nothing else than the power method, applied to an extended matrix. This leads to completely similar convergence expressions for both methods. Furthermore, some attention will be paid to a backward error analysis based upon the Rayleigh quotient. This will permit the estimation of the necessary convergence level of the approximating vector in the iteration.

Finally, it should be mentioned that the scope of this paper is not to provide a general, complete algorithm for the SVD.

This paper only summarizes some simple but original results and some new viewpoints on rather classical techniques.

Also in [De Moor, 1984] some similar results are obtained, and demonstrated in more detail, but only for the power method.

2. THE SINGULAR VALUE DECOMPOSITION

In this section, the main theoretic and well known results on the singular value decomposition are summarized for notational convenience. Also, some historical background is provided together with an overview of existing algorithms. Finally, we summarize the main specifications for the adaptive SVD strategy to be developed here.

2.1 The Autonne-Eckart-Young theorem (restricted to real matrices)

Every real rectangular mxn matrix A can be decomposed as

$$\underset{mxn}{A} = \underset{mxn}{U} \;.\; \underset{mxn}{\Sigma} \;.\; \underset{mxn}{V^t}$$

where U & V are orthogonal $U^t.U = I_m = U.U^t$ and $V^t.V = I_n = V.V^t$ and Σ is diagonal rectangular: $\Sigma = \text{diag}\,(\sigma_1, \sigma_2, \ldots, \sigma_r, 0, \ldots 0)$ where by convention $\sigma_1 \geqslant \sigma_2 \geqslant \ldots \geqslant \sigma_r > 0$ and r = algebraic rank of A.

The columns u_i (v_i) of U (V) are the left (right) singular vectors and the real positive numbers σ_i are the singular values.

From the theorem it follows immediately that:

1. Every matrix A can be written as a direct sum of r rank one matrices (dyadic decomposition)

$$A_{m\times n} = \sum_{i=1}^{r} u_i . \sigma_i . v_i^t$$

2. The singular values squared are the eigenvalues of $A.A^t$ and $(A^t.A)$ because $A.A^t = U.\Sigma . \Sigma^t . U^t$ and hence $(A.A^t). u_i = \sigma_i^2.u_i$.

These facts will be used in the development of the Chebyshev algorithm. Of course, there are many more extremely interesting properties of the SVD, that can be found in excellent textbooks e.g. [Golub (1984), Hansen (1985)].

2.2 Algorithms for the SVD

Of course, it is straight forward to compute the SVD of A via the eigendecomposition of $A.A^t$ or $A^t.A$. However, the explicit computation of $A.A^t$ & $A^t.A$ introduces numerical errors that halve the numerical precision achievable with this technique, compared to operating with the matrix A itself. Especially, the small singular values are sensitive to the implicit squaring (Klema & Laub, 1978).

One of the most reliable and efficient algorithms for the SVD of an arbitrary matrix is of course that of Golub (1965). It uses orthonormal transformations to bidiagonalize the matrix, whereafter the QR iteration is used to compute the singular values.

However, this algorithm is very costly with respect to memory and computational requirements ($O(n^3)$) and since it operates on the matrix entries itself, it deletes completely a possible structure of the matrix (Hankel, Toeplitz, Sparse...). However, no adaptive version has been developed where a priori information can be used (for instance SVD of a previous time step).

These drawbacks are not shared by the Lanczos algorithm, which was studied extensively in [Paige (1972)] and is nowadays considered as a very promising algorithm for large, structured and preferable sparse eigenvalue problems.

Another class of algorithms is the Jacobi methods, an approach that will probably lead to parallel computer implementation [Hari, 1984]. In this paper we give some promising results for another algorithm that may possibly perform better in a number of applications, that are characterized by the following specifications.

2.3 Specifications

1 Partial SVD

In many applications, only few singular values of the singular spectrum are needed. Let us mention some examples:

-In the solution of overdetermined sets of linear equations, corrupted by white noise, the solution vector is given by the right singular vector, corresponding to the smallest singular value (De Moor, 1985 b). In Pisarenko's line spectrum estimation method, the minimal eigenvector of the symmetric covariance matrix is needed (Furhmann, 1984).

-In linear system theory, a minimal realization is obtained via the SVD of a block Hankel matrix. Here only some few dominant triplets are needed, (Kung, Zeiger Mc Ewen) similarly in image compression applications and algebraic image restoration techniques (Shlien, 1982).

-In order to extract a fetal ECG, noisy measurements are projected on line onto a subspace spanned by an intermediate singular vector of the measurement matrix, that can be shown to correspond to the fetal heart, where the dominant singular values correspond to the mother heart (Vanderschoot, 1983). Other applications where intermediate eigen-and singularvalues are needed are mentioned in Jensen (1972). In all those examples, only a partial SVD is needed.

2 Large and/or structured matrices

There are a lot of applications where the matrix dimensions are large (say a few hundred) and the matrix itself can be highly structured ((block)Hankel, - Toeplitz, ...). To avoid excessive memory and computation requirements, one should not operate on the matrix entries itself, as the Golub algorithm does, hence deleting the possible structure.

For those cases, it is much more convenient to access the data indirectly, via a matrix vector multiplication subroutine, which does not alter the matrix entries.

This matrix-vector access concept will also be exploited in the following specification.

3 Adaptive computation of SVD

Two types of time varying sequences of matrices are possible:

-Adding a new column or row and omitting one (for instance, when new data become available and old data are to be neglected).

-Another possibility is time variance element-wise when all elements change slowly in each timestep.

For those cases the SVD's on two subsequent timesteps are related and hence, the SVD on timestep T could be used as initial guess for the SVD on T + 1 in an iterative algorithm.

The algorithm that will be described here is perfectly well suited for this, since it uses a matrix vector subroutine and since it is iterative. Golub's algorithm makes no use of an a priori available first initial guess.

4 Convergence level control

Measured data are always corrupted by noise. Also the matrix singular value decomposition is perturbed by the noise although the singular values are themselves perfectly well conditioned (Golub, 1983). But, since the data are noisy, a full 7 or 14 (or more) digit accuracy is of no use and in most cases only a limited number of accurate digits will suffice, especially when only few singular vectors are to be computed. In this paper, some original results are obtained that permit one to control the level of convergence. Together with the backward error analysis, this is a powerful tool in the analysis of the SVD of noisy matrices.

3. THE CLASSICAL POWER METHOD

Because the Chebyshev algorithm is related to the power method (in Wilkinson it is considered as an acceleration of the power method), we first present briefly some results on the power method for the SVD. More details can be found in [De Moor, 1985 a].

3.1 Algorithm

Suppose already p-1 singular triplets have been computed : (u_i, σ_i, v_i), i = 1, (p-1) let e_p^k, s_p^k, f_p^k be the k-th iteration approximation for u_p, σ_p, v_p; let e_p^O be an initial guess for u_p, then step k in the iteration for the triplet is:

step k

```
1.  Orthonormalize $e_p^{k-1}$ with respect to $(u_1 \ldots u_{p-1})$ so that
        $||e_p^{k-1}|| = 1$ (Deflation step)

2.  Compute
        $f_p^k = A^t.e_p^k$
        $s_p^k = ||A^t.e_p^k||$
        $f_p^k = f_p^k/s_p^k$

3.  Test for convergence $||f_p^k - f_p^{k-1}|| < TOL$
    If so, convergence in triplet p
    If not, go to 4

4.  Compute   $e_p^k = A.f_p^k \;/\; ||A.f_p^k||$

5.  Set k = k+1, go to 1.
```

The properties of this simple iteration scheme and possible refinements are extensively studied in De Moor (1983). (Omission of normalization, orthogonalization, acceleration algorithm, optimal initial guesses, deflation techniques, ...).

Before proving the convergence of the iteration scheme, we first need the following

Definition: The internal coordinates of an iteration vector $e_p^k (f_p^k)$ are the coordinates of $e_p^k (f_p^k)$ in the left (right) singular base, given by $U^t.e_p^k (V^t.f_p^k)$.

3.2 *Convergence*

3.2.1 *Infinite precision arithmetic*

Theorem: If e_p^O is not orthogonal to u_p then

$$\lim_{k\to\infty} e_p^k = u_p, \quad \lim_{k\to\infty} f_p^k = v_p, \quad \lim_{k\to\infty} s_p^k = \sigma_p.$$

Proof: Suppose in the algorithm, each orthogonalization and normalization step is omitted, then

$$e_p^k = \underbrace{A.A^t \;..........\; A.A^t}_{k \text{ times } A.A^t}.e_p^O$$

and $A.A^t = U.(\Sigma.\Sigma^t)U^t$ where $U^t.U = I$

so that $(U^t e_p^k) = (\Sigma.\Sigma^t)^k.\ (U^t.e_p^O)$

Here, $U^t e_p^k$ and $U^t.e_p^O$ are the internal coordinates of e_p^k and e_p^O.

Since e_p^O is orthonormalized to $(u_1, \dots u_{p-1})$, the internal coordinates have the form

$$U^t.e_p^O = (0, \dots, 0, g_p, g_{p+1}, \dots g_n)^t$$

$$U^t.e_p^k = (0, \dots, 0, \left(\sigma_p\right)^{2k} g_p, \left(\sigma_{p+1}\right)^{2k} g_{p+1} \dots)^t$$

and after normalizing and taking the limit and with the generic assumption that $\sigma_p > \sigma_{p+1} > \dots > \sigma_m$ we have for the i-th component

$$\lim_{k\to\infty} \frac{\sigma_i^{2k} g_i}{\sqrt{\sigma_p^{4k} g_p^2 + \sigma_{p+1}^{4k} g_{p+1}^2 + \dots}} = \delta_{ip}$$

where δ_{ip} is the Kronecker delta.

Hence $$\lim_{k\to\infty} \frac{U^t.e_p^k}{||U^t e_p^k||} = (0, \dots 0, \underset{\substack{\uparrow \\ p}}{1}, 0 \dots 0)^t$$

but these are nothing else than the internal coordinates of u_p.

The same technique can be used in proving the convergence of f_p^k to v_p and s_p^k to σ_p.

3.2.1 *Finite precision arithmetic*

Finite precision arithmetic is always characterized by the so called machine precision $\varepsilon_m = \sup \{\varepsilon | fl(1+\varepsilon) = 1\}$ where fl(.) denotes the floating point operator (Wilkinson 1965). The effect of finite machine precision on the addition of two real numbers a,b can be seen from $(a+b) = a(1+\frac{b}{a})$. If $\frac{b}{a} < \varepsilon_m$, then $fl\ (1+\frac{b}{a})=1$ and $fl(a+b) = a$. Hence, if two numbers a and b differ more than $(-\log_{10} \varepsilon_m)$ decimal digits, addition has no effect.

This finite precision causes the internal coordinates of e_p^0 to be of the form $fl(U^t.e_p^k) = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_{p-1}, g_p, g_{p+1}, \ldots g_m)$ where the ε_i are of the order of magnitude of ε_m.

Now, if we suppose that the orthogonalization keeps the first p-1 internal coordinates small $(\sim\varepsilon_m)$, then it is not difficult to see that numerically, geometric convergence occurs when the p-th internal coordinate dominates all others by more than $[-\log \varepsilon_m]$ decimal digits. Since for a "good" initial guess for the p-th left singular vector u_p, the internal coordinates satisfy $g_p > g_{p+1} > g_i$, $i = p+2,m$ convergence occurs if

$$\log_{10} \frac{|\sigma_p^{2k} g_p|}{|\sigma_{p+1}^{2k} g_{p+1}|} > -\log_{10}\varepsilon_m$$

or after

$$k = \frac{-\log_{10}\ \varepsilon_m - \log_{10} \left[\frac{g_p}{|g_{p+1}|}\right]}{2\log_{10} \left[\frac{\sigma_p}{\sigma_{p+1}}\right]}$$

matrix vector multiplications with $A.A^t$. under the assumption that the first (p-1) are kept to machine precision level with orthogonalization. The derived formula is of course very interesting since it shows clearly the three different effects that influence the number of iterations.

1. $\log_{10} \frac{\sigma_p}{\sigma_{p+1}}$: the larger the gap between the computed value σ_p and the subsequent one σ_{p+1}, the faster the convergence. For clustered singular values, $\sigma_p/\sigma_{p+1} \approx 1$, convergence will be very slow. For those cases of slow convergence, acceleration algorithms have been developed and described in (De Moor, 1983).

2. $\log_{10}(g_p/g_{p+1})$: the better the quality of the initial guess, the smaller the number of iterations. For a very good initial vector, the internal coordinates of e_p^0 satisfy $g_p >> g_{p+1}$.

3. $(-\log_{10} \varepsilon_m)$: This quantity measures the number of desired correct digits in the iteration vector. Hence if for instance only t correct digits are necessary, one puts $-\log_{10} 10^{-t} = t$ instead of $(-\log \varepsilon_m)$, which is equivalent to a full machine precision result.

   Hence, the numerator $[-\log t - \log (g_p/g_{p+1})]$ measures the number of digits that have to be modified in order to obtain the singular vector up to the desired precision $t = -\log_{10} 10^{-t}$ decimal digits.

   It is important to mention that the factors $\log g\frac{g_p}{p+1}$ and $\log \frac{\sigma_p}{\sigma_{p+1}}$ can be estimated dynamically from the convergence pattern (De Moor, 1983).

Using this numerical analysis in internal coordinates, not only the number of iterations for t digit precision can be estimated during the iteration. In [De Moor, 1984], one can find results on the determination of the optimal moment for fixing the initial guess for the next triplet and on the estimation of the critical deflation-orthogonalisation frequency.

4. THE CHEBYSHEV ALGORITHM

4.1 The Chebyshev polynomials

Orthogonal polynomials in general, and Chebyshev polynomials in particular, play an important role in numerical linear algebra (Stiefel (1955), Fox (1968), Rivlin (1976), ...).

The application of Chebyshev polynomials in the eigenvalue problem is studied for the symmetric case in Parlett (1980), Stewart (1969) and for the non symmetric case in Wilkinson (1965), Saad (1982, 1984) and Manteuffel (1977). These references show that the idea of using Chebyshev polynomials in order to accelerate iterative eigenvalue algorithms is certainly not new.

Chebyshev polynomials belong to the class of polynomials that are orthogonal with respect to a given density function (Stiefel, 1955). They can be defined in several ways (Rivlin, 1976) but we only need here some important, well known properties.

4.1.1 Recurrence relation

All orthogonal polynomials satisfy a three term recurrence relation. For the Chebyshev polynomials, this becomes

$$T_0(x) = 1$$

$$T_1(x) = x$$

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

4.1.2 Behaviour of the Chebyshev polynomials

An important property is $\forall\, n,\ x \in [-1,1] : |T_n(x)| \leqslant 1$

$x \notin [-1,1] : |T_n(x)| \geqslant 1$

This follows of course from the definition

$$T_n(x) = \cos(n \operatorname{Bgcos} x), \ |x| \leqslant 1$$

$$T_n(x) = \cosh(n \operatorname{Bgcosh} x), \ |x| \geqslant 1$$

This means that in $[-1,1]$, the Chebyshev polynomials show an equiripple oscillation with turning points and zeros symmetrically disposed about x=0.

For $x \notin [-1,1]$, the behaviour is characterized by

$$|T_{n+1}(x)| > |T_n(x)|$$

$$|T_n(x + \Delta x)| > |T_n(x)|, \quad (\Delta x > 0)$$

and by the following original theorem

Theorem

For $d_1 > d_0 > 1$ the Chebyshev polynomials satisfy

$$\frac{T_{n+1}(d_1) - T_{n+1}(d_0)}{T_n(d_1) - T_n(d_0)} > 1$$

Proof

Can be easily proved by induction.

The following theorem is extremely important for the SVD algorithm that will be developed.
The proof can be found in Rivlin (1976).

Theorem

The Chebyshev polynomial $2^{1-n}.T_n(s)$ is always the largest possible polynomial of degree n with leading coefficient 1, outside $[-1,1]$. For every $p_n(x)$, polynomial of degree n with leading coeff. 1 and if $M = \max |p_n(x)|$, $x \in [-1,1]$, then for $|y| > 1$

$$|p_n(y)| \leqslant M.2^{1-n}|T_n(y)|$$

It will turn out further on that this theorem shows that the Chebyshev algorithm is the best possible strategy.

4.1.3 *Asymptotic behaviour for large n*

For large n, the following expressions hold (Parlett, 1980)

$$T_n(1 + 2\varepsilon) \sim \frac{1}{2}(1+2\sqrt{\varepsilon} + 2\varepsilon)^n \text{ for } 0 \leqslant \varepsilon \leqslant 0.1$$

$$\sim \frac{1}{2} e^{(2n\sqrt{\varepsilon})} \text{ for } n\sqrt{\varepsilon} > 1$$

It is this rapid growth of $T_n(1+2.\varepsilon)$ for small ε that makes the Chebyshev polynomials attractive for computation of the SVD.

4.2 The Chebyshev algorithm

4.2.1 Computation of the dominant triplet

Suppose one wants to compute the dominant triplet (u_1, σ_1, v_1) of A (m<n) with $\sigma_1(A) > 1$ and $0 < \sigma_i(A) < 1$ $i=2, m$ and the SVD of $A = U.\Sigma.V^t$.

Let e_1^0 be an initial guess for u_1 and define $e_1^k = T_k(A.A^t).e_1^0$
Since polynomials are analytic functions one has
$T_k(A.A^t) = U.T_k(\Sigma.\Sigma^t).U^t$ and $U^t.e_1^k = T_k(\Sigma.\Sigma^t).(U^t.e_1^0)$

$(U^t e_1^k)$ and $(U^t e_1^0)$ are the internal coordinates of the k-th iteration vector e_1^k and the initial guess e_1^0.

Let $U^t e_1^0 = (g_1, g_2, \ldots, g_m)$ then

$$U^t e_1^k = (T_k(\sigma_1^2)g_1,\ T_k(\sigma_2^2)g_2, \ldots, T_k(\sigma_m^2)g_m)$$

Because of the conditions on the singular values of A and the properties of the Chebyshev polynomials, one has

$$|T_{k+1}(\sigma_1^2)| > |T_k(\sigma_1^2)| > 1$$

$$|T_k(\sigma_i^2)| \leq 1 \qquad i=2,m$$

Hence, the first internal coordinate grows for increasing k while all other coordinates are bounded by 1.

In the next section the convergence will be proved analytically but first, the vector iteration scheme is deduced.

By the recurrence relation of the Chebyshev polynomials

$$T_k(AA^t) = 2(A.A^t)T_{k-1}(AA^t) - T_{k-2}(AA^t)$$

so that

$$\boxed{\begin{array}{l} e_1^k = 2\ A.(A^t e_1^{k-1}) - e_1^{k-2} \\ \text{with } e_1^1 = A.(A^t.e_1^0) \end{array}}$$

This is the fundamental iteration scheme of the SVD Chebyshev iteration method.

4.2.2 Convergence

Under the assumption $\sigma_1(A) > 1$ and $0 < \sigma_i(A) < 1$, $i=2,m$ numerical convergence of $t=(-\log \varepsilon)$ digits will be reached if the first internal coordinate dominates the other by t decimal digits.
Hence, numerical convergence occurs if

$$\log \frac{|T_k(\sigma_1^2)\ g_1|}{\max\limits_{i=2,m} |T_k(\sigma_i^2) g_i|} = -\log \varepsilon$$

Since $|T_k(\sigma_i^2) g_i| < 1$, $i=2,m$ this becomes for the worst case

$$\log |T_k(\sigma_1^2)|$$

This equation determines the number of iterations k (multiplications with $A.A^t$) necessary for numerical convergence of $t=-\log \varepsilon$(decimal digits). Of course, σ_1 is unknown, hence $T_k\ (\sigma_1^2)$, but the number of multiplications can be estimated by considering the asymptotic convergence behaviour as follows:

In iteration step k, the number of correct decimal digits is $\log |T_k(\sigma_1^2).g_1/\max\limits_{i=2,m} g_i|$.

In iteration step k+1, this becomes $\log |T_{k+1}(\sigma_1^2)|.g_1/\max\limits_{i=2,m} g_i|$

Hence, from k to k+1 the gain in number of correct digits is

$$\log |T_{k+1}(\sigma_1^2)| - \log|T_k(\sigma_1^2)| = \log \frac{|T_{k+1}(\sigma_1^2)|}{|T_k(\sigma_1^2)|}$$

For "large" k, we now prove the following

<u>Theorem</u>

If $d > 1$, then $\lim\limits_{k \longrightarrow \infty} \frac{T_{k+1}(d)}{T_k(d)} = d + \sqrt{d^2 - 1} = L$

Proof

From $T_{k+1}(d) = 2dT_k(d) - T_{k-1}(d)$ it follows that

$$\frac{T_{k+1}(d)}{T_k(d)} = 2d - \frac{T_{k-1}(d)}{T_k(d)}$$

But under very general conditions

$$L = \lim_{k \longrightarrow \infty} \frac{T_{k+1}(d)}{T_k(d)} = \left[\lim_{k \longrightarrow \infty} \frac{T_k(d)}{T_{k-1}(d)} \right]^{-1}$$

and hence $L = 2d - \frac{1}{L}$ or $L^2 - 2\,dL + 1 = 0$ with solutions $L = d \pm \sqrt{d^2 - 1}$

But since for $d > 1$, $|T_{k+1}(d)| > |T_k(d)|$ also $L > 1$ and hence $L = d + \sqrt{d^2 - 1}$ is the only acceptable solution.

From this theorem, it follows that for each iteration with $A.A^t$, $\log(\sigma_1^2 + \sqrt{\sigma_1^4 - 1})$ decimal digits converge.

From section 3.2, it then follows that the number of matrix multiplications with AA^t is approximated by

$$k = \frac{-\log \varepsilon - \log(g_1/\max_{i=2,m} g_i)}{\log(\sigma_1^2 + \sqrt{\sigma_1^4 - 1})}$$

The expression in the numerator is asymptotically correct. One can show numerically that it is an overestimation of the true value

$$\left[\frac{T_{k+1}(\sigma_1^2)}{T_k(\sigma_1^2)} \right]$$

which however, converges rather fast to the correct limit.

The interpretation of the convergence expression is quite similar to the case of the power method. Only the numerator is changed into the expression $(\log(\sigma_1^2 + \sqrt{\sigma_1^4 - 1})$, which is the number of left multiplications with $A.A^t$ (Chebyshev iteration) per decimal digit of convergence. As we shall derive further on, this factor can be estimated from the convergence pattern. To conclude this section, let us mention that some similar expression of the form $d+ \sqrt{d^2-1}$ in the convergence behaviour can be found in a very interesting paper on Chebyshev iteration in Saad (1984).

4.2.3 Spectrum transformation

Till now, we only considered the case where $\sigma_1(A) > 1$, $0 < \sigma_i(A) < 1$, $i = 2,m$ and proved that in this case, the algorithm converges to the dominant triplet. In order to compute the dominant, smallest or some intermediate triplets, of a general matrix, some more information is needed in the form of upper- and lower bounds on the desired singular value(s).

4.2.3.1 Dominant triplet.

Suppose one knows real numbers a and b such that $\sigma_2^2 \leqslant a \leqslant \sigma_1^2$ and $0 \leqslant b \leqslant \sigma_m^2$.

Then the linear function

$$f\ (x) = \frac{2x - (a+b)}{a-b}$$

satisfies $f(a) = 1,\ f(b) = -1$

$$f(\sigma_1^2) > 1,\ |f(\sigma_i^2)| \leqslant 1 \quad i = 2,m$$

such that

$$\begin{aligned} f(A.A^t) &= \frac{2}{a-b}\ A.A^t - \frac{a+b}{a-b}\ I \\ &= V \left[\frac{2}{a-b}\ \Sigma.\ \Sigma^t - \frac{a+b}{a-b}\ I\right] . V^t \\ &= U.D.V^t \end{aligned}$$

is a matrix satisfying the conditions of §4.2.1, with eigenvalues

$$|d_1| = \left| \frac{2\sigma_1^2 - (a+b)}{a-b} \right| > 1 \text{ and } |d_i| = \left| \frac{2\sigma_i^2 - (a+b)}{a-b} \right| \leqslant 1$$

i=2,m.

Only the singular values (eigenvalues of $A.A^t$) are changed but the singular vectors are conserved.

This means that the following Chebyshev matrix vector iteration scheme will converge to the dominant singular triplet.

Iteration scheme

1. Determine a,b
2. Choose e_1^0
   compute $e_1^1 = A.(A.^t.e_1^0)$

Step k

3. $e_1^{k+1} = \frac{4}{a-b} A.(A^t e_1^k) - 2\,\frac{a+b}{a-b}\, e_1^k - e_1^{k-1}$
4. Test for convergence

$$\left\| \frac{e_1^{k+1}}{\|e_1^{k+1}\|} - \frac{e_1^k}{\|e_1^k\|} \right\| < \text{TOL}$$

If so, stop.
If not, go back to 3

or alternatively

4' Estimation of quantities in convergence expression.

If conv., stop
If not go to 3

Of course, the convergence rate is now proportional to

$$1/\log\left(d_1 + \sqrt{d_1^2 - 1}\,\right)$$

where $d_1 = \frac{2\sigma_1^2 - (a+b)}{a-b} > 1$

Now, from the properties of the Chebyshev polynomials, converge is optimal if $|d_1 - 1|$ is as large as possible. It is not difficult to prove that this corresponds to optimal choices of a and b as $a = \sigma_2^2$ and $b = \sigma_m^2$.

Hence, the best possible convergence is determined by the ratio $K_{12} = \sigma_2^2 / \sigma_1^2$ and $K_{2m} = \sigma_m^2 / \sigma_2^2$.

The smaller K_{12} and the larger K_{2m}, the better the convergence rate. From the theorem of optimality of Chebyshev polynomials, it follows that this strategy is the best possible using only matrix polynomials.

4.3.2 Smallest triplet

The necessary bounds a and b must now satisfy : $a > \sigma_1^2$

$$\sigma_m^2 < b < \sigma_{m-1}^2$$

and the spectrum transformation function is here

$$f(A.A^t) = \frac{2.AA^t}{b-a} - \frac{a+b}{b-a} I$$

since

$$f(a) = -1 \qquad f(\sigma_m^2) > 1$$

$$f(b) = 1 \qquad |f(\sigma_i^2)| < 1 \qquad i = 1, (m-1)$$

The Chebyshev iteration now becomes

$$e_m^{k+1} = \frac{4}{b-a} A.(A^t e_m^k) - 2 \frac{a+b}{b-a} e_m^k - e_m^{k-1}$$

and the convergence rate is determined by

$1/ \log (d_m + \sqrt{d_m^2 - 1})$ iterations per decimal digit with

$$d_m = [\ \sigma_m^2 - (a+b)\ ] / (b-a)$$

Optimal convergence condition occurs for $a = \sigma_1^2$, and $b = \sigma_{m-1}^2$.

4.2.3.3 Intermediate triplet (u_i, σ_i, v_i).

Now, 4 bounds are needed for triplet i

$$A > \sigma_1^2 \quad \text{(with } 1 < i < m)$$

$$B < \sigma_m^2$$

$$\sigma_{i-1}^2 < b < \sigma_i^2 < a < \sigma_{i+1}^2$$

and define $M_a = A-a$, $M_b = b-B$, $D = a-b$, $M = \max\ (M_a, M_b)$

The following strategy is optimal for the symmetric case where $M_a = M_b = M$.

Instead of a linear spectrum transformation spectrum, now a parabola is used

$$f(x) = px^2 + qx + t \text{ with } p = \frac{-2}{M(M+d)}, q = -(a+b)\ p$$

$$t = 1 + ab.p$$

For more details about the deviation, omitted here for brevity, we refer to De Moor (1985a).

The Chebyshev iteration now becomes

$$e_i^{k+1} = 2\ p\ A\ [\ A^t\ (A(A^t e_i^k) + \frac{q}{p}\ e_i^k)\ ] + 2te_i^k - e_i^{k-1}$$

Each iteration now demands 4 matrix vector multiplications.

Optimal convergence occurs when $\sigma_i^2 = \frac{a+b}{2}$ and then

$$d_i = f(\sigma_i^2) = 1 + \frac{1}{2(\frac{M}{D})\ (\frac{M}{D} + 1)}$$

while the optimal convergence factor is then: $1/\log\ (d_i + \sqrt{d_i^2 - 1})$ iterations per decimal digit.

From this it follows that, the smaller M/D, the faster will be the convergence. Unfortunately the M/D ratio is usually large in practical situations. A possible strategy then would consist in allowing more iteration vectors and hence perform a kind of subspace iteration (De Moor 1985a).

In the case where the interval D is located asymmetrically ($M_a \ll M_b$ or vice versa), the ratio M/D can be "preprocessed" in order to obtain attractive convergences rates (De Moor 1985a).

To conclude this section, let us illustrate the computation of an intermediate triplet by a simple example.

Let $A.A^t$ = diag (10;9;6;4;1) and given bounds B=0, A=11, a=8, b=5 then p=-0.05; q=0.65- t=-1.

Then $f(AA^t)$ = diag (0.5; 0.8; 1.1; 0.8; -0.4) and hence there are needed

$$\frac{1}{\log\,(d_1 + \sqrt{d_1^{\,2} - 1})} = \frac{1}{\log\,(1.1 + \sqrt{(1.1)^2 - 1})} = 5.2$$

iterations per desired digit of precision. This is equivalent with 5.2 x 4 = 21 matrix vector multiplications per digit.

4.3 *Relation Chebyshev-power iteration*

4.3.1 *Number of iterations*

For identical initial guess vector and machine precision, it follows from the convergence expression that the ratio of matrix-multiplications of the dominant triplet for the Chebyshev method compared to the power method, is given by

$$S = \frac{{}^{n}(\text{Chebyshev})}{{}^{n}(\text{power method})} = \frac{2\,\log\,(\sigma_1/\sigma_2)}{\log\,(d_1 + \sqrt{d_1^{\,2} - 1})}$$

where $d_1 = \dfrac{2\sigma_1^2 - (a+b)}{-b + a}$ with $b<\sigma_m^2$ and $\sigma_2^2 < a < \sigma_1^2$.

When S < 1, then the Chebyshev method is faster than the power method.

Now take b=0, $\alpha = a/\sigma_2^2$ and $r = \sigma_1/\sigma_2$

then $$S = \frac{\log r^2}{\log\left(2\,(r^2/\alpha) - 1 + \sqrt{(2r^2/\alpha - 1)^2 - 1}\right)}$$

This convergence speed ratio is plotted in Fig. 1 for different values of α and r.

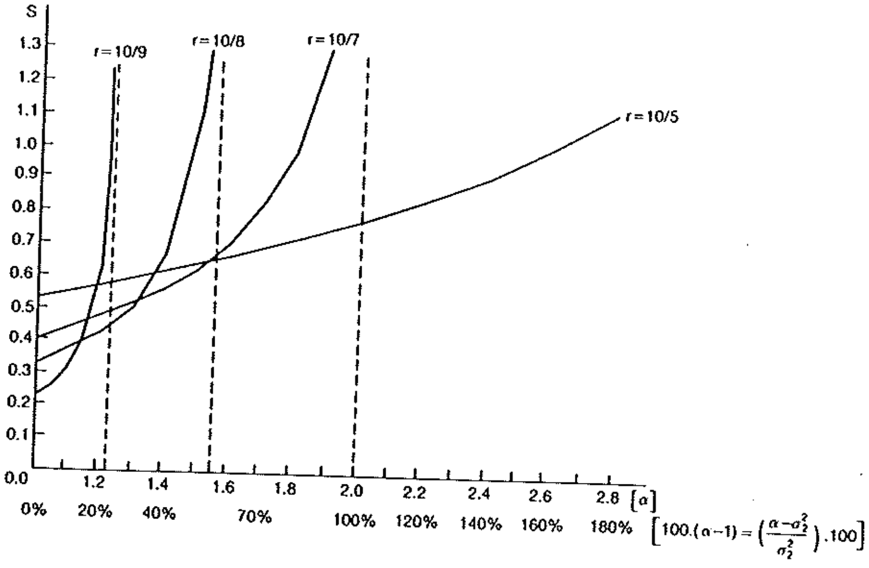

Fig. 1 r is the singular value ratio. For values of S smaller than 1, the Chebyshev method is faster than the power method. For small values of r, it is necessary to estimate the lower bound a of σ_1 accurately (optimally $a=\sigma_2$).

From this it immediately follows that the closer the singular values are ($r \longrightarrow 1$) the larger the gain in speed of the Chebyshev iteration in comparison with the power method but of course, the more difficult it will be to estimate a satisfying $\sigma_2^2 < a < \sigma_1^2$. For correct and optimal estimation of a $(= \sigma_2^2)$, S, as a function of r is given in Fig. 2.

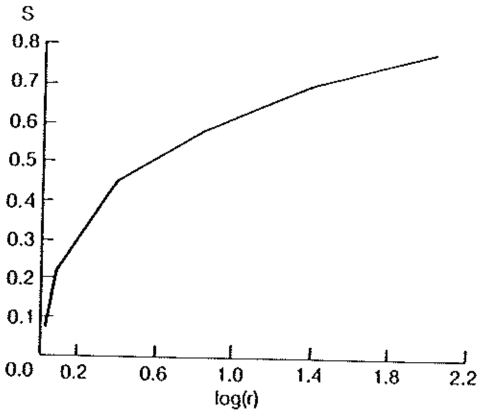

Fig. 2 If the bound a is estimated accurately, the Chebyshev method is almost always faster than the power method. The more the singular values are clustered, the larger the difference in iterations between Chebyshev and power method, but the more difficult it is to estimate a.

From this it is clear that, the more clustered σ_1 and σ_2 are, (log r ⟶ 0), the better is the convergence rate of the Chebyshev iteration compared to the power method.

In some forthcoming work, the application of acceleration algorithms will be investigated for the case of clustered singular values (see De Moor 1985 a).

4.3.2 Relation with power method

The following statement considerably simplifies the study of the sometimes rather complicated looking Chebyshev schemes.

Theorem

The Chebyshev iteration scheme

$$e_1{}^1 = f(A.A^t)e_1{}^0$$

$$e_1{}^{n+1} = 2\ f(AA^t)e_1{}^n - e_1{}^{n-1}$$

is completely equivalent with the eigenvalue power method applied to the matrix

$$B = \begin{bmatrix} -I & 2\ f(AA^t) \\ -2f(AA^t) & 4\ [\ f(AA^t)\ ]^2\ -\ I \end{bmatrix}$$

with initial vector $\begin{bmatrix} e_1^{\,0} \\ e_1^{\,1} \end{bmatrix}$

Proof

From $e_1^{\,n+1} = 2\ f(AA^t)e_1^{\,n} - e_1^{\,n-1}$

it follows that

$$e_1^{\,n+2} = 2\ f(AA^t)e_1^{\,n+1} - e_1^{\,n}$$

$$= [\ 4\ [\ f(AA^t)\ ]^2 - \ I\ ]e_1^{\,n} - 2\ f(AA^t)\ e_1^{\,n-1}.$$

which can be written as

$$\begin{bmatrix} e_1^{\,n+1} \\ e_1^{\,n+2} \end{bmatrix} = \begin{bmatrix} -I & 2f(AA^t) \\ -2f(AA^t) & 4f(AA^t)^2-I \end{bmatrix} \begin{bmatrix} e_1^{\,n-1} \\ e_1^{\,n} \end{bmatrix}$$

This is nothing else than the eigenvalue power iteration method. Moreover, the eigenvalue spectrum of this matrix is easily obtained as follows:

Since $f(AA^t) = Vf(\Sigma\ \Sigma^t)V^t$ the eigenvalues of B follow from

$$B = \begin{bmatrix} V & O \\ O & V \end{bmatrix} . \begin{bmatrix} -I & 2f(\Sigma\ \Sigma^t) \\ -2f(\Sigma\ \Sigma^t) & 4[\ f(\Sigma\ \Sigma^t)\ ]^2\ -\ I \end{bmatrix} \begin{bmatrix} V^t & O \\ O & V^t \end{bmatrix}$$

$$= U.D.U^t$$

and by putting det $(D-\Lambda) = 0$ it is easily computed that

$$\Lambda = 2[\, f(\Sigma\ \Sigma^t)\,]^2 - I \pm 2f(\Sigma\ \Sigma^t)\sqrt{f(\Sigma\ \Sigma^t)^2 - I}$$

Now put $f(\sigma_1^2) = d_i$ then the eigenvalues are real for $d_i > 1$ and imaginary for $d_i < 1$.

Suppose for simplicity of explanation that only $d_1 > 1$ and all other $d_i < 1$.

Then

- for $d_1 > 1$ $\lambda_1 = 2d_1^2 - 1 + 2\, d_1 \sqrt{d_1^2 - 1}$

$$\lambda_{-1} = 2d_1^2 - 1 - 2d_1 \sqrt{d_1^2 - 1}$$

and one can prove that

$$\lambda_1 > 1 \text{ and } 0 < |\, \lambda_{-1} | < 1$$

and that $\lambda_{-1} = \dfrac{1}{\lambda_1}$

- for $d_i < 1$

$$\lambda_i = 2d_i^2 - 1 + 2d_i j\sqrt{1-d_i^2} \qquad \text{where } j^2 = -1$$

$$\lambda_{-1} = 2d_i^2 - 1 - 2d_i j\sqrt{1-d_i^2}$$

and $|\lambda_i| = |\, \lambda_{-i}| = 1$

The desired part of the spectrum ($d_i > 1$) is mapped on the real axis and split into two eigenvalues $\lambda_i > 1$ and $\lambda_{-i} = \dfrac{1}{\lambda_i}$ for each singular value and the rest of the spectrum is mapped on the unit circle !!

The convergence pattern can now be predicted with the techniques of §3 for the power method.

The internal coordinate corresponding to the dominant eigenvalue λ_i will grow every iteration by a factor (worst case)

$$\log \frac{\lambda_1}{\max\limits_{i>2} |\lambda_{\pm i}|} = \log [\ 2d_1^{\ 2} -1+2d_1\sqrt{d_1^{\ 2} - 1}\ ]$$

But one step of the eigenvalue power method corresponds to 2 Chebyshev recursions. Hence one Chebyshev recursion step is equivalent to a growth determined by

$$\begin{aligned}[\ 2d_1^{\ 2}-1 + 2d_1\sqrt{d_1^{\ 2} - 1})^{\frac{1}{2}} &= ((d_1 + \sqrt{d_1^{\ 2} - 1})^2)^{\frac{1}{2}} \\ &= d_1 + \sqrt{d_1^{\ 2} - 1}\end{aligned}$$

which is precisely the quantity computed asymptotically in §4.4.

5. BACKWARD ERROR MODIFICATION

In many practical cases, the matrix A for which the SVD is desired, is perturbed by noise and uncertainties in the data. It is a well known result that the sinaular values are very well conditioned [Golub, 1983] and that the condition of the singular vectors depends on the relative spread of the singular values.

The methods we discuss here, apply at every moment in the iteration, to an approximation of both left and right singular vectors.

Since obviously it is useless to require a higher accuracy in the results than can be guaranteed by the data, some tools must be constructed in order to check the relative accuracy to the data accuracy level. The theorems developed further on are straightforward generalizations of those in [Parlett 1980].

Let A be mxn, x an m-vector approximation to the left singular vector u_k, and y an n-vector approximation to the right singular vector v_k with $\|\ x\ \| = \|\ y\ \| = 1$.

Define the generalized Rayleigh quotient as $\rho = x^t.A.y = y^t.A^t.x$. It can be proved that ρ is the best possible approximation for σ_k, using x and y only.

Define the residues $\underset{nx1}{r_x} = A^t.x - \rho.y$ and $\underset{mx1}{r_y} = A.y - \rho.x$

then the following theorems have very practical implications.

Theorem

The residues r_x and r_y are orthogonal to the approximations x and y.

$$r_x^t.y = 0 \quad r_y^t.x = 0$$

Proof : Trivial

Theorem Backward error modification.

The vectors x and y are left and right singular vectors of a modified matrix A - M with corresponding singular value s where

$$M = x.r_x^t + r_y.y^t$$

Proof it is easy to show that

$$(A - M)y = s .x$$
$$(A^t - M^t).x = s .y$$

Theorem The singular value decomposition of M is given as

$$M = \begin{bmatrix} x & \dfrac{r_y}{||r_y||} \end{bmatrix} \begin{bmatrix} ||r_x|| & 0 \\ 0 & ||r_y|| \end{bmatrix} \begin{bmatrix} r_x^t/||r_x|| \\ y^t \end{bmatrix}$$

Proof Trivial

From this, the norms of M are given as $||\ M\ || = \max\{ ||r_x||, ||r_y||]$ and $||M||_F^2 = ||r_x||^2 + ||r_y||^2$.

Now, one can prove that for two matrices A,B with singular values a_i and b_i, $|a_i - b_i|<||A - B||$. Then put B=A-M, which has a singular value s, $|\sigma_k - s|<||A - (A-M)|| = ||M|| = \max (||r_x||,|r_y||)$. This means that the error in the approximation of a singular value is bounded by the norm of the backward error matrix M.

If this norm $||M||$ becomes smaller than the estimated uncertainty level in one singular value, further iteration is useless.

As an example, consider the uncertainty level in a matrix A which is perturbed by white noise with standard deviation σ. Then the uncertainty level on one singular value is proportional to $\max(m,n).\sigma^2$ (De Moor 1985b). Hence iteration can stop when $||M|| < \max(m,n)\sigma^2$. However, this backward error modification expression has to be linked with the convergence expressions derived earlier.

6. AN ADAPTIVE STRATEGY AND PERSPECTIVES

In this section, we briefly summarize some ideas and perspectives to implement the Chebyshev iteration scheme in an adaptive environment. The data are not modified during computation and may be accessible for instance only in secondary memory.

The data themselves are assumed to be slowly time varying either in a way that at each time step a new line of information is added or such that minor changes occur elementwise. Moreover, some noise level characteristics are assumed to be known. The initial guess for each triplet can be obtained from the SVD of a previous time step or from a previous obtained triplet. The number of iterations for convergence is determined with respect to the desired accuracy level, that in its turn is determined by the noise level.

In the near future, the adaptive estimation of optimal singular value bounds is to be studied, for which interesting ideas can already be found in the work of Saad (1984) and Manteuffel (1977). The operation count is to be optimized by acceleration algorithms and optimal orthogonalization (deflation) strategies (De Moor 1985a).

A main problem of course is the resolution of closely spaced singular values. But also other methods such as the Lanczos method, where the relative spread of singular values equally well determines the convergence rate (Parlett, 1980), suffer from this problem.

In the case of the Lanczos algorithm some block versions have been developed for this purpose (Golub, 1981), and extensions of our Chebyshev method to the subspace iteration case are straight-forward (Parlett, 1980). Moreover, there is a large parallelism between this Chebyshev iteration and the Lanczos method. Both use only matrix vectors subroutines, need initial guesses to start up the iteration, need some deflation techniques such as selective orthogonalization, and convergence is governed by the condition and spread of the singular spectrum.

However, a direct comparison is not straight-forward and it would be very interesting to set up one.

7. CONCLUSIONS

In this paper two SVD algorithms, the power method and the Chebyshev iteration method, were studied in a unifying framework. Convergence expressions were obtained and it was proven that the Chebyshev method is also a power method. Comparisons in convergence speed were made. It was demonstrated that the typical applications for these algorithms are those where the SVD of a slowly time varying sequence of matrices is to be computed. The matrices are large and/or structured, and accessible only via matrix vector products. Knowledge of the characteristics of the noise on the data can be used to estimate the number of desired digits in the results or to follow the convergence via the backward error modification technique.

REFERENCES

1. De Moor B., Vandewalle, J., Staar, J. and Lauwers, J., (1984), An adaptive singular value decomposition algorithm and its applications to adaptive realization. Proc. of the 6th Intern. Conference on Analysis and Optimization of systems, Nice.

2. De Moor B., (1985a) Some new results for adaptive singular value decomposition algorithms. Int. report, ESAT, K.U. Leuven.

3. De Moor B., (1985b) Overdetermined sets of linear equations: trade-off between uncertainty and number of solutions. Int. report, ESAT, K.U. Leuven.

4. Fox, L. and Parker, I., (1968) Chebyshev polynomials in numerical analysis. Oxford Univ. Press, London.

5. Fuhrmann, D.R. and Liu, B., (1984) An iterative algorithm for locating the minimal eigenvector of a symmetric matrix, Proc. of the 1984 ISCAS conference.

6. Golub, G.H. and Kahan, W., (1965) Calculating the singular values and pseudo-inverse of a matrix. *Siam J. Num. Anal.* 2, 205-24.

7. Golub, G.H., Luk, F.T. and Overton, M.L., (1981) A block Lanczos method for computing the singular values and corresponding singular vectors of a matrix. ACM. Transactions on mathematical Software, Vol. 7, No. 2.

8. Golub, G.H. and Van Loan, C.F., (1983) Matrix computations. Johns Hopkins University Press, Baltimore.

9. Hansen, P.C., (1984) SVD, Theory and applications. rep. NI-84-05, Techn. Univ. Denmark.

10. Hari, V. and Veselic, K., (1981) On Jacobi methods for singular value decomposition. Int. report of department of mathematics, University of Zagreb, Yugoslavia.

11. Jensen, P.S., (1972) The solution of large symmetric eigenproblems by sectioning. *Siam. J. Num. Anal.*, vol. 9, No. 4.

12. Klema, V.C. and Laub, A.J., (1980) The singular value decomposition: its computation and some applications. *IEEE Trans. Aut. Contr.*,

13. Kung, S.Y., (1978) A new identification and model reduction algorithm via singular value decomposition. Proc. 12th Asitomar Conf. Circ. Syst.,

14. Manteuffel, T., (1977) The Chebyshev iteration for nonsymmetric linear systems. *Num. math.* 28, pp. 307-327.

15. Paige, C., (1972) Computational variants of the Lanczos method for the eigenproblem. *J. Inst. math. Appl.* 10.

16. Parlett, B., (1980) The symmetric eigenvalue problem. Prentice Hall Series in Computational mathematics. Prentice Hall Inc., Englewood Cliffs, N.J. 07632.

17. Rivlin, T.J., (1976) The Chebyshev polynomials. J. Wiley and Sons, New York.

18. Saad, Y., (1984) Chebyshev acceleration techniques for solving nonsymmetric eigenvalues problems. Math. of Comput., Vol. 42, No. 166, p. 567-588.

19. Schlien, S., (1982) A method for computing the partial singular value decomposition. *IEEE Trans. on pattern analysis and machine intelligence*, vol. PAMI 4, No. 6.

20. Stiefel, E.L., (1955) Kernel polynomials in linear algebra and their numerical application. *Appl. Math. Ser. Nat. Bur. Stand*, 49, 1-22.

21. Vanderschoot, J. and Vandewalle, J. et al (1984) Extraction of weak bioelectrical signal by means of singular value decomposition. Proc. of 6th Internat. Conf. on Analysis and Optimization of Systems, Nice.

22. Veselic, K., (1976) A convergent Jacobi method for solving the eigenproblem of arbitrary real matrices. *Num. Math.* 25, 179-184.

23. Wilkinson, (1965) The algebraic eigenvalue problem. Clarendon Press, Oxford.

24. Zeiger, H.P. and Mc Ewen, A.J., (1974) Approximate linear realizations of given dimension via Ho's algorithm. *IEEE Trans. Aut. Contr.*, Vol. AC-19, pp. 153.