INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

On some Structured Inverse Eigenvalue Problems

Robert Erra, Bernard Philippe

N° 2604

Juin 1995

PROGRAMME 6

Rapport de recherche

ISSN 0249-6399

On some Structured Inverse Eigenvalue Problems

Robert Erra*, Bernard Philippe**

Programme 6 — Calcul scientifique, modélisation et logiciel numérique
Projet ALADIN

Rapport de recherche n° 2604 — Juin 1995 — 21 pages

Abstract: This work deals with various finite algorithms that solve two special *Structured Inverse Eigenvalue Problem* (SIEP).

The first problem we consider is the *Jacobi Inverse Eigenvalue Problem* (JIEP): given some constraints on two sets of real, find a Jacobi matrix J (real symmetric tridiagonal with positive nondiagonal entries) that admits as spectrum and principal subspectrum the two given sets. Two classes of finite algorithms are considered. The polynomial algorithm is based on a special Euclid-Sturm algorithm (Householder's terminology) which has been rediscovered several times. The matrix algorithm is a symmetric Lanczos algorithm with a special initial vector. Some characterization of the matrix insures the equivalence of the two algorithms in exact arithmetic.

The results of the symmetric situation are extended to the non-symmetric case: this is the second SIEP which is considered : the *Tridiagonal Inverse Eigenvalue Problem* (TIEP). Possible breakdowns may occur in the polynomial algorithm as it may happen with the non-symmetric Lanczos algorithm. The connection between the two algorithms exhibits a similarity transformation from the classical Frobenius companion matrix to the tridiagonal matrix.

This result is used to illustrate the fact that, when computing the eigenvalues of a matrix, the non-symmetric Lanczos Algorithm can lead to a slow convergence, even for a symmetric matrix since an outer eigenvalue of the tridiagonal matrix of order $n-1$ can be arbitrarily far from the spectrum of the original matrix.

Key-words: Jacobi matrix, Euclid algorithm, Sturm sequence, Lanczos algorithm, inverse eigenvalue problems.

(Résumé : tsvp)

*ESIEA 9 rue Vesale, 75005 PARIS FRANCE
**INRIA/IRISA Campus de Beaulieu, 35042 RENNES Cedex FRANCE philippe@irisa.fr

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu, 35042 RENNES Cedex (France)
Téléphone : (33) 99 84 71 00 – Télécopie : (33) 99 84 71 71

Problèmes inverses de valeurs propres

Résumé : Cet article présente différents algorithmes directs qui résolvent deux problèmes inverses de valeurs propres structurés. Le premier problème est le problème inverse de Jacobi : étant donné quelques contraintes sur deux ensembles de réels, construire une matrice de Jacobi qui admette comme spectre et spectre principal d'ordre immédiatement inférieur les deux ensembles de nombres. On étudie deux classes d'algorithmes : un algorithme de polynômes souvent redécouvert et l'algorithme de Lanczos symétrique avec un vecteur initial approprié.

La situation précédente est ensuite étendue au cas non symétrique : elle correspond au deuxième problème inverse. Dans ce cas l'algorithme de polynômes peut s'arrêter avant la fin, comme cela peut arriver avec l'algorithme de Lanczos non symétrique. Dans le cas sans échec, on exhibe une transformation de similitude entre la matrice tridiagonale et la matrice compagnon de Frobenius qui relie les deux algorithmes. Une conséquence de ce dernier résultat permet de montrer que l'algorithme de Lanczos non symétrique utilisé pour calculer les valeurs propres d'une matrice d'ordre n peut mener à une matrice d'ordre $n-1$ dont une valeur propre est arbitrairement loin du spectre de la matrice initiale.

Mots-clé : Matrice de Jacobi, algorithme d'Euclide, suite de Sturm, algorithme de Lanczos, problèmes inverses de valeurs propres.

1 Introduction

In the last centuries, the computation or localisation of some (or all) the roots of a polynomial has received a considerable attention. Another related problem has emerged and, progressively, became of fundamental interest for the community of numerical analysis: the computation or localisation of some (or all) of the eigenvalues of a matrix.

More precisely, for every matrix algorithm we can look at its polynomial counterpart. For instance, Bernoulli's method (1729) for finding the largest root of an algebraic equation $p(x) = 0$ is equivalent to the Von Mises-Geiringer method (1928) applied on the Frobenius companion matrix of $p(x)$ which computes its largest root by the power method. However the translation is not always obvious.

Two algorithms have been the object of a great deal of work: *the Routh's Algorithm* (1877), based on the Sturm theorem (1835), which computes the number of roots of a polynomial in a given interval and: *the Lanczos's Algorithm* (1950) which tridiagonalises a matrix. The purpose of this work is to point out and make clear the formal connections existing between these two famous algorithms and to give a relevant bibliography.

The link becomes clear with a modified version of Routh's Algorithm which we call *Routh-Lanczos algorithm* and for which we present a brief history. Given a polynomial $p(x)$ with real and simple roots, these algorithms build a Jacobi matrix (ie. a real irreducible symmetric tridiagonal matrix) whose characteristic polynomial is $p(x)$. For sake of simplicity we will only consider polynomials with real coefficients.

The fact that we impose the *structure* of the matrix solution of problems (2.1) and (2.2) explains the denomination *structured inverse eigenvalue problems* due to Boley and Golub [BG87] that we have declined in JIEP, studied in sect. (2), BIEP and TIEP, studied in sect. (4).

Notations

In this work $J = [\beta_k, \alpha_k, \gamma_{k+1}]$ denotes the tridiagonal matrix

$$J = \begin{pmatrix} \alpha_1 & \gamma_2 & & & & \\ \beta_2 & \alpha_2 & \gamma_3 & & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & \beta_{n-1} & \alpha_{n-1} & \gamma_n \\ & & & & \beta_n & \alpha_n \end{pmatrix}.$$

When J is real symmetric, we always suppose that we have $\beta_i = \gamma_i \geq 0$.

If $p(x)$ is a polynomial with real coefficients $p(x) = \sum_{i=0}^{n} a_i x^i$ the classical Frobenius's companion matrix associed to p is:

$$C(p) = \begin{pmatrix} 0 & 1 & & & 0 \\ 0 & 0 & 1 & & 0 \\ 0 & 0 & \ddots & & 0 \\ & \ddots & \ddots & \ddots & \vdots \\ -\frac{a_0}{a_n} & -\frac{a_1}{a_n} & \cdots & & -\frac{a_{n-1}}{a_n} \end{pmatrix} \quad (1)$$

For every matrix J of order n, we call *upper principal submatrix* the matrix of order $n-1$ obtained by deleting the last row and the last column of J. Similarly the matrix of order $n-1$ obtained by deleting the first row and the first column of J is called the *lower principal submatrix*, and the spectra of these matrices are called *principal subspectra.* I_n is the identity matrix of order n.

For a matrix A we note $A^P = \mathcal{I}_n \cdot A^T \cdot \mathcal{I}_n$ where

$$\mathcal{I}_n = \begin{pmatrix} 0 & \cdots & & 0 & 1 \\ 0 & \cdots & 0 & 1 & 0 \\ \vdots & & & & \\ 0 & 1 & 0 & \cdots & \\ 1 & 0 & \cdots & & 0 \end{pmatrix} \quad (2)$$

$\mathcal{I}_n$ is the *reversal* matrix of order n, and A^P is the *pertransposition* of A. A n-th order matrix A such that $A^P = \mathcal{I}_n \cdot A^T \cdot \mathcal{I}_n = A$ is called a persymmetric matrix.

2 Jacobi Inverse Eigenvalue Problems

Discrete Inverse Eigenvalue Problems have first been proposed by Downing and Householder [DH56] (see [Fri77, FNO87] for newer references).

We limit our attention, in this section, to the following situations which we define as the *upper Jacobi Inverse Eigenvalue Problems* (an upper JIEP), that has been raised by Hochstadt ([Hoc67]):

Problem 2.1 *Let $\Lambda = \{\lambda_i\}_{i=1\cdots n}$ and $\Omega = \{\omega_i\}_{i=1\cdots n-1}$ be two sets of real, satisfying the strict interlacing property*

$$\lambda_1 < \omega_1 < \lambda_2 < \omega_2 \cdots < \omega_{n-1} < \lambda_n \quad (3)$$

Find a Jacobi matrix J_n such that:

$$sp(J_n) = \{\lambda_i\}_{i=1\cdots n} \quad and \quad sp(J_{n-1}) = \{\omega_i\}_{i=1\cdots n-1} \quad (4)$$

where $sp(A)$ denotes the spectrum of A and J_{n-1} the upper principal submatrix of order $n-1$ of J_n.

Problem 2.2 *Let $p_n(x)$ and $p_{n-1}(x)$ be two monic polynomials of degree n and $n-1$ respectively which are assumed to have strictly interlaced real roots.*

Find a Jacobi matrix J_n such that

$$det(xI_n - J_n) = p_n(x) \quad and \quad det(xI_{n-1} - J_{n-1}) = p_{n-1}(x) \tag{5}$$

where J_{n-1} is the upper principal submatrix of order $n-1$ of J_n .

We can also define the lower (JIEP), where J_{n-1} is replaced by the lower principal submatrix of order $n-1$ of J_n. We must point out that, if J_n is a solution of the upper (JIEP) with data Λ and Ω then $J_n^P = \mathcal{I} * J_n * \mathcal{I}$ is a solution of the lower (JIEP) with data Λ and Ω, see also [BG78].

Another related problem is the following *persymmetric* (JIEP):

Problem 2.3 *Let $\Lambda = \{\lambda_i\}_{i=1\cdots n}$ be a set of real increasing. Find a persymmetric Jacobi matrix J_n having Λ as spectrum.*

Problems (2.1) and (2.3) come first from the discretisation of continuous inverse Sturm-Liouville problems. The problem of computing Gauss quadrature formula gives also (SIEPs): for example computing the Gauss quadrature formula of order $2n$, knowing the Gauss quadrature formula of order n can be interpreted as the following

Problem 2.4 *Let J_n be a Jacobi matrix of order n and let Λ_{2n} be a set of $2n$ reals $\{\lambda_i\}_{i=1\cdots 2n}$ strictly increasing. Find a Jacobi matrix J_{2n} having Λ_{2n} as spectrum and whose principal upper submatrix of order n is J_n (problem DD in [BG87]).*

References and other exemples of (SIEPs) can be found in [GK60, Hal76, BG87]. These problems can be seen as particular cases of the more general following problem: the *Banded Inverse Eigenvalue Problem* (BIEP): i.e the problem of the reconstruction of a symmetric p-banded $n \times n$ matrix from the spectral data with additional interlacing conditions. See [BK81, BG87, MH81] for the case $2p+1 < n$ and [Fri79] for the special case $2p+1 = n$ (reconstruction of the complete symmetric matrix) with methods different from those that we will discuss here.

Hald ([Hal76]) proved that the first problem is well-posed :

Theorem 2.1 *Let J and $\tilde{J}$ be solutions of Problem (2.1) with respective data*

$$\lambda_1 < \omega_1 < \lambda_2 < \omega_2 \cdots < \omega_{n-1} < \lambda_n$$

and

$$\tilde{\lambda_1} < \tilde{\omega_1} < \tilde{\lambda_2} < \tilde{\omega_2} \cdots < \tilde{\omega_{n-1}} < \tilde{\lambda_n}$$

There exists a constant K such that

$$||J - \tilde{J}||_E \leq K[\sum_{i=1}^{n}(\lambda_i - \tilde{\lambda_i})^2 + \sum_{i=1}^{n}(\omega_j - \tilde{\omega_j})^2]^{\ 1/2} \tag{6}$$

where $\| \, \|_E$ is the Frobenius norm. The constant K is bounded in terms of n and ν_0 ($K = O(n, \nu_0)$) defined by:

$$\nu_0 = \delta_0 \sqrt{\frac{\epsilon_0}{2\delta_0 + \epsilon_0}}$$

where

$$\epsilon_0 = \frac{1}{d} \min_{j,k}(|\lambda_j - \omega_k|, |\tilde{\lambda_j} - \tilde{\omega_k}|)$$

and

$$\delta_0 = \frac{1}{2d} \min_{j \neq k}(|\lambda_j - \lambda_k|, |\omega_j - \omega_k|, |\tilde{\lambda_j} - \tilde{\lambda_k}|, |\tilde{\omega_j} - \tilde{\omega_k}|)$$

with

$$d = max(\lambda_n, \tilde{\lambda_n}) - min(\lambda_1, \tilde{\lambda_1}).$$

Obviously, Problems (2.1) and (2.2) are equivalent and, therefore this theorem implies that Problem (2.2) is well-posed as well.

3 Existing algorithms for JIEP

For sake of completeness, we first outline two families of algorithms that can be used to solve (JIEP), namely the Lanczos algorithm and the Euclid-Sturm algorithm with a special emphasis on Routh's algorithm. We point out that we only consider *direct or finite algorithms*, i.e algorithms that terminate in a finite number of steps. See [FNO87] for a survey of iterative algorithms for inverse eigenvalue problems.

3.1 Euclid-Sturm Algorithms

The integer Euclid's algorithm is one of the oldest "nontrivial" algorithms. For a full description see [Knu81] and for a survey see [Bar74].

Definition 3.1 *Let f and g be two polynomials of degree n and m respectively. We call a* generalised Euclid algorithm *([Knu81]), any algorithm based on the following recursion:*

$$\begin{cases} p_n = f \quad and \quad p_{n-1} = g, \\ \delta_k \cdot p_k = q_k \cdot p_{k-1} + sign(\nu_k) \cdot \gamma_k \cdot p_{k-2} \quad for \quad k = n, \ n-1, \cdots \end{cases} \tag{7}$$

where $(sign(\nu_k) \cdot \gamma_k \cdot p_{k-2})$ is the remainder of the polynomial division of $(\delta_k \cdot p_k)$ by p_{k-1}.

It is often used for computing a G.C.D of f and g: if $p_{k-2} \equiv 0$ then p_{k-1} is a G.C.D of f and g. Traditionnally, the sequence $\{p_n, \ p_{n-1}, \cdots\}$ is called a *polynomial remainder sequence* (p.r.s). We remark that the terms of the remainder sequence are unique up to a scalar multiplication.

Classical choices for δ_k, $sign(\nu_k)$ and γ_k are:

1. $\delta_k = 1$, $\nu_k = 1$ and $\gamma_k = 1$ which corresponds to the classical Euclid algorithm.
2. $\delta_k \neq 1$, $sign(\nu_k) = 1$, and $\gamma_k = 1$, for polynomials over a unique factorisation domain (polynomials with integer coefficients for example).
3. $\delta_k = 1$, $\nu_k = -1$, and $\gamma_k = 1$ which corresponds to the classical Sturm algorithm ([Stu35]). If p_n and p_{n-1} satisfy the assumptions of Problem (2.2), then the p.r.s obtained stops with the constant polynomial $p_0 \equiv 1$ and is a Sturm sequence [Jac74].
4. $\delta_k = 1$, $\nu_k = -1$ and γ_k chosen in order to obtain a monic polynomial for p_{k-2}.

The latter case has been proposed several times (with minor modifications) :

- Schwarz can be considered as the first who proposed an algorithm that builds a tridiagonal (eventually complex) matrix admitting f as characteristic polynomial and g as the characteristic polynomial of the upper principal submatrix. His procedure can be expressed in the formalism of (7), with a special choice of g [Sch56], called *the alternant*, g is defined by $g(z) = 1/2 \cdot [f(z) - (-1)^n f^*(-z)]$, where f^* is a polynomial whose coefficients are the conjugate of the coefficients of f. This work follows Wall's work who studied connections between infinite Jacobi matrices, submatrices of finite order of such matrices and the associated continued fractions, see chapters IX, X, XI, XII in ([Wal48]).
- Collins proposed it as an attempt to keep small the magnitude of the coefficients in the polynomial G.C.D. computation [Col67]. He chose $g = f'$ which is not a monic polynomial and $\nu_k = 1$. This algorithm is often used in computer algebra systems [Mig91].
- Householder studied the same recurrence and provided relationships between the coefficients of the p.r.s. [Hou74].
- Hald seems to be the first to have introduced it for solving Problem (2.1). He proved that, by taking p_n and p_{n-1} satisfying the assumptions of Problem (2.2), the p.r.s obtained is still a Sturm sequence but with monic polynomials [Hal76].
- Ben-Or and al [BOFKT88, BOT90], following Collins's idea, used that algorithm for computing the roots of a given polynomial which has only real roots. They also propose a parallel version of their algorithm.
- More recently, Schmeisser has rediscovered Hald's algorithm, see [Sch93]. He proposed this algorithm to solve a problem posed by Fiedler at the *International Colloqium on Applications of Mathematics* in Hamburg in july 1990 (see also [Fie90]): *Given a monic polynomial $p(x)$ with only real roots, construct a real symmetric matrix for which $p(x)$ is the caracteristic polynomial.*

Householder calls *Euclid-Sturm algorithms* the algorithms based on the fourth recursion. We propose to call Euclid-Sturm algorithms all algorithms based on the third and fourth recursions.

For completeness, we must point out that Rutishauser's QD algorithm, proposed in 1954, can be used to compute a tridiagonal matrix J_n of order n which satisfies

$$det(x \cdot I_n - J_n) = p(x)$$

when $p(x)$ is a polynomial of degree n with distinct roots. This algorithm can be viewed as a special case (and at the *origin*) of the matrix LR algorithm.

Wendroff ([Wen61]) has studied a variant of problem (2.2): *the construction of a sequence of orthogonal polynomials* $\{g_0, g_1, \cdot, g_{n-1}, g_n\}$, *given* $g_{n-1} = p_{n-1}$ *and* $g_n = p_n$. He proved that under the assumptions of problem (2.2) this problem has a unique solution, which is a Sturm sequence, but his demonstration doesn't use explicitly an Euclid-Sturm scheme and he doesn't mention problem (2.1). We can point out that problem (2.3) has been studied by Hald ([Hal76]), and has initially been studied by Gantmacher-Krein in the book ([GK60]) published in German in 1960 (but published in russian in 1950).

A well-known Euclid-Sturm Algorithm is the Routh's Algorithm. Routh stated his famous algorithm to compute the number of roots of a real polynomial, which lie in the left-half complex plane. We follow here the presentation of Barnett and Siljak's extensive survey [BS77] which was published for the centennial of the algorithm.

Given two polynomials f and g such that:

$$f(x) = \alpha_0 \cdot x^n + \alpha_1 \cdot x^{n-1} + \cdots + \alpha_n \tag{8}$$

$$g(x) = \beta_0 x^{n-1} + \beta_1 x^{n-2} + \cdots + \beta_{n-1} \tag{9}$$

with $\alpha_0 \neq 0$ and $\beta_0 \neq 0$.

Then the Routh Array is the set of rows:

$$\begin{array}{lllll} r_{0,1} & r_{0,2} & \cdots & & r_{0,n+1} \\ r_{1,1} & r_{1,2} & \cdots & r_{0,n} & \\ r_{2,1} & \cdots & & & \\ \cdots & \cdots & & & \end{array}$$

where $r_{0,j} = \alpha_{j-1}$ and $r_{1,j} = \beta_{j-1}$ for corresponding j. The array is generated by the rule:

$$r_{i,j} = \frac{-\begin{vmatrix} r_{i-2,1} & r_{i-2,j+1} \\ r_{i-1,1} & r_{i-1,j+1} \end{vmatrix}}{r_{i-1,1}} \tag{10}$$

This is the so-called determinantal form of the Routh's algorithm. It avoids the need of a polynomial division. The relation between the Sturm sequence $f_0(x), f_1(x), \cdots, f_n(x)$, obtained from (7) which correponds to the second case for δ_k and γ_k, and the first column of the Routh's array is the following :

- $f_0(x) = f(x)$
- $f_1(x) = g(x)$
- $f_i(x) = \delta_i \cdot (\sum_{j=0}^{n-i} r_{2\cdot i-1,m-i+j-1} \cdot x^j)$, where $\delta_i = sign(\prod_1^{2\cdot i-2} r_{j,1})$.

See also [Gan59] and [BS77] for newer references.

From the Routh's array one can construct the Schwarz matrix, which is a tridiagonal real matrix, but is generally non symmetric:

$$S = \begin{pmatrix} 0 & 1 & 0 & & & 0 \\ -r_n & 0 & 1 & & & \\ & -r_{n-1} & 0 & 1 & & \\ & & \ddots & \ddots & \ddots & 0 \\ & & & -r_3 & 0 & 1 \\ & & & & -r_2 & -r_1 \end{pmatrix} \tag{11}$$

where $r_1 = r_{11}$, $r_2 = r_{21}$ and $r_i = r_{i1}/r_{i-2,1}$ for $i > 2$, see [Sch56]. A similar result is implicit in Wall's book. The tridiagonal matrix R, similar to S, defined by

$$R = \begin{pmatrix} 0 & -(r_n)^{1/2} & 0 & & & 0 \\ (r_n)^{1/2} & 0 & -(r_{n-1})^{1/2} & & & \\ & (r_{n-1})^{1/2} & 0 & & & \\ & & \ddots & \ddots & \ddots & 0 \\ & & & & 0 & -r_2^{1/2} \\ & & & & r_2^{1/2} & -r_1 \end{pmatrix} \tag{12}$$

where the square roots can be complex, is called the Routh's matrix, see [BS77]. The number of positive terms in the sequence r_1, r_1r_2, $\cdots, r_1r_2\cdots r_n$ is equal to the number of roots of $p(x)$ with negative real parts [BS77].

3.2 The symmetric Routh-Lanczos algorithm

To solve Problem (2.2), one can compute the p.r.s. by the *fourth* case of recurrence presented in subsection (3.1). Hald proved then that all the quotient polynomials are monic and of degree 1 and all the coefficients γ_k are positive [Hal76]. Then the recurrence can be written :

$$p_k(x) = (x - \alpha_k)p_{k-1}(x) - {\beta_k}^2 p_{k-2}(x) \quad k = n, \cdots, 2 \tag{13}$$

which is the well-known three-term recurrence for computing the characteristic polynomial of the Jacobi matrix $J_n = [\beta_k,\ \alpha_k,\ \beta_{k+1}]$, but *computed in a reverse way.* We note that the polynomials $p_0(x), \cdots, p_n(x)$ are called *Lanczos Polynomials* by Householder in [Hou64].

The obtained tridiagonal matrix J_n

$$J_n = \begin{pmatrix} \alpha_1 & \beta_2 & & & & \\ \beta_2 & \alpha_2 & \beta_3 & & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & \beta_{n-1} & \alpha_{n-1} & \beta_n \\ & & & & \beta_n & \alpha_n \end{pmatrix}.$$

which is, by construction, a Jacobi matrix and is solution of Problem (2.2). We call this procedure *the symmetric Routh-Lanczos Algorithm.* J_n is characterized by the following :

Theorem 3.1 *(Hald) Let $\{\lambda_i\}_{i=1\cdots n}$ and $\{\omega_i\}_{i=1\cdots n-1}$ be two sets of real, satisfying the strict interlacing property 3. There exists a unique Jacobi matrix of order n such that its eigenvalues are $\{\lambda_i\}_{1,n}$ and the eigenvalues of J_{n-1} are $\{\omega_i\}_{1,n-1}$.*

Proof : See [Hal76] for a complete proof. □

Hochstadt ([Hoc67, Hoc74]) proved that there exists *at most* one solution to problem (2.1) and Gary-Wilson ([GW76]) proved that there exists *at least* one solution. Hald ([Hal76]) seems to be the first who explicitely proved that problem (2.1) has a unique solution (using an Euclid-Sturm scheme).

3.3 The Lanczos algorithm

The algorithm, which was called p-q algorithm by Lanczos ([Lan50]), is well known, frequently used and has been thoroughly studied. It has been initially proposed by Lanczos *circa* 1950 for computing the minimal polynomial of a matrix.

The use of Lanczos Algorithm to solve Problem (2.1) has been first proposed in [BG78] and [BG77] (see [BG87] for more information and references). The goal is to choose the starting vector from the given eigenvalues data :

Proposition 3.1 *Let $\{\lambda_i\}_{i=1\cdots n}$, $\{\omega_i\}_{i=1\cdots n-1}$ be two sets of strictly interlaced real. If Δ is the diagonal matrix of which diagonal entries are $\{\lambda_i\}_{i=1\cdots n}$ and if the vector $q = (q_1, \cdots, q_n)^T$ is defined by:*

$$q_k{}^2 = \frac{\prod_{j=1}^{n-1}(\omega_j - \lambda_k)}{\prod_{j=1,\ j\neq k}^{n}(\lambda_j - \lambda_k)} \tag{14}$$

then, the symmetric Lanczos Algorithm applied on Δ, with a starting vector q satisfying (14), generates the Jacobi matrix $[\beta_k, \alpha_k, \beta_{k+1}]$ which is solution of Problem (2.1).

A straight application of Hald's Theorem [Hal76] insures that, in exact arithmetic, the Jacobi matrix obtained here is equal to the matrix $J_n^P = \mathcal{I} \cdot J_n \cdot \mathcal{I}$ obtained in the previous section with the Routh-Lanczos algorithm. The fact that we must use the pertransposition of J_n comes from the fact that the Routh-Lanczos Algorithm solves an upper (JIEP) while the symmetric Lanczos Algorithm used in 3.1 solves a lower (JIEP).

Such a result can be used for proving Scott's result about the fact that even symmetric Lanczos can converge slowly in the sense of the Kaniel-Paige-Saad's theory [Sco79] :

Corollary 3.1 *Let A be a real symmetric matrix of order n, with $sp(A) = \{\lambda_1, \lambda_2, \cdots, \lambda_n\}$. For every set of reals $\{\omega_1, \cdots, \omega_{n-1}\}$ which strictly interlaces $sp(A)$, there exists a real vector p_0 such that, if J_n is the Jacobi matrix obtained by the symmetric Lanczos algorithm with starting vector p_0, and J_{n-1} it's principal submatrix, then we have*

$$sp(J_{n-1}) = \{\omega_1, \cdots, \omega_2\}$$

In particular we can choose p_0 such that

$$\omega_i = \frac{\lambda_i + \lambda_{i+1}}{2} \quad \forall i = 1, \cdots, n-1$$

3.4 Other Algorithms for solving a (JIEP)

We have seen that constructing a Jacobi matrix is equivalent to construct a sequence of orthogonal polynomials. We must point out that the two oldest known procedures for generating orthogonal polynomials, are the *Stieltjes* procedure ([Sti84]) and the *Chebyschev* procedure ([Che58, Che59]). Both algorithms compute a sequence of orthogonal polynomials with a *forward process*, i.e they compute p_k before p_{k+1}. The Stieltjes procedure has been studied by Gautschi [Gau82, Gau85], and the Chebyshev procedure has been also advocated by Gautschi [Gau82] and by Gutknecht and Gragg ([GG94]) ; the two algorithms have a $O(n^2)$ complexity . Both algorithms can be used to solve a (JIEP).

In [Rei91] Reichel compares the Stieltjes procedure and an algorithm proposed by Gragg and Harrod ([GH84]) for solving a (JIEP). This last algorithm belongs to another class of algorithms that can be used to solve a JIEP: they are based on a modification of a Rutishauser's algorithm ([Rut63]). For example, the Rutishauser-Kahan-Pal-Walker algorithm, proposed by Gragg and Harrod ([GH84, BG87]), consists in applying a sequence of carefully chosen orthogonal plane rotations in a given order onto an arrowheaded matrix (also called diagonal bordered matrix). A similar algorithm has been first proposed by Biegler-Konig ([BK81]). These algorithms are $O(n^3)$ algorithms and, evidently, they compute, in exact arithmetic, the same Jacobi matrix J_n constructed by the Routh-Lanczos and the Lanczos algorithms.

Finally, we must point out that an algorithm of the same class has been recently proposed by Ammar and Gragg ([AG91]) for solving a (BIEP). It uses an efficient pattern of rotations which provides a stable $O(n^2)$ algorithm. It seems to be the first *numerically stable* algorithm that can solve a (JIEP) in $O(n^2)$ arithmetic operations.

4 General polynomials

4.1 Complete and incomplete p.r.s.

An interesting question arises when polynomials are not anymore assumed to have real interlacing roots.

Problem (2.2) becomes

Problem 4.1 *Let $p_n(x)$ and $p_{n-1}(x)$ be two real monic polynomials of degree n and $n-1$ respectively.*

Find a tridiagonal matrix J_n such that

$$det(xI_n - J_n) = p_n(x) \quad and \quad det(xI_{n-1} - J_{n-1}) = p_{n-1}(x) \tag{15}$$

where J_{n-1} is the principal submatrix of order $n-1$ of J_n.

We call this second *structured inverse eigenvalue problem* a (TIEP) (for *Tridiagonal Inverse Eigenvalue Problem*).

We can still try to compute the p.r.s. $\{p_k(x)\}_{k=n,\cdots,0}$ with the Routh-Lanczos algorithm which corresponds to the fourth choice of Section 3.1 :

$$p_k = (x - \alpha_k)p_{k-1}(x) - \gamma_k p_{k-2}(x) \quad k = n, \cdots, 2 \tag{16}$$

In contrast to (JIEP), the process (7) computing the p.r.s. can stop before reaching the remainder of degree one. Before exhibiting such situations, let us consider the following definition :

Definition 4.1 *Let $p_n(x)$ and $p_{n-1}(x)$ be the two starting polynomials. We say that the p.r.s is complete when the degree of polynomial p_k is equal to k and the last computed polynomial is of degree 1.*

A necessary condition to have a complete p.r.s is to start with two relative prime polymomials p_n and p_{n-1}, since the last remainder of the sequence is their monic G.C.D. If the polynomials are not prime the algorithm stops after having computed a multiple of the G.C.D.(p_n,p_{n-1}). This situation is the first case where an *incomplete p.r.s* is computed instead of the complete p.r.s expected.

But $G.C.D.(p_n, p_{n-1}) = 1$ is not a sufficient condition to insure a complete p.r.s. For example : the sequence defined by $p_n(x) = x^n - 1$ and $p_{n-1} = x^{n-1}$ gives $p_{n-2}(x) = 1$ and the process stops immediately for any $n > 2$. This situation is the second case of computation of an incomplete p.r.s. This fact was known during the 19-th century but it's not clear if Sturm knew that his algorithm could suffer from breakdown (for further references, see [KN81], english translation of a work published in Russian in 1936).

The preceeding example shows that Problem 4.1 contrasts drastically with the JIEP. We can point out the fact that the matrix $C(p_3)$, when $p_3(x) = x^3 - 1$ is exactly the example chosen in [PTL85] to show that Nonsymmetrix Lanczos Algorithms may breakdown.

Similarly to the non-symmetric Lanczos Algorithm, we call *happy breakdown* the first case and *serious breakdown* the second. It is as hard to overcome this situation by choosing another polynomial p_{n-1} as it is for the selection of a "good" pair of initial vectors in the Lanczos process.

4.2 A non-symmetric Routh-Lanczos Algorithm

If the recurrence (16) gives a complete p.r.s, it allows us to define the tridiagonal matrix

$$J = \begin{pmatrix} \alpha_1 & 1 & & & & \\ \gamma_2 & \alpha_2 & 1 & & & \\ & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & \gamma_{n-1} & \alpha_{n-1} & 1 \\ & & & & \gamma_n & \alpha_n \end{pmatrix} \tag{17}$$

which is, by construction, solution of (4.1). This is *the non-symmetric Routh-Lanczos algorithm.*

We establish now the connection between this algorithm and the non-symmetric Lanczos algorithm.

We obtained a result, which has been suggested by a similar relation between the Frobenius companion matrix and the *Comrade Matrix* in [Bar75], and a Pury and Weygandt's result which states that a Frobenius companion matrix C is similar to the Routh canonical tridiagonal matrix R (12) through a tranformation matrix which is triangular[BS77].

If $p_n(x)$ and p_{n-1} are polynomials with real coefficients, we denote by T the lower triangular matrice T, composed of the coefficients of the p.r.s. $p_0(x), \cdots, p_{n-1}(x)$ produced by the Routh-Lanczos algorithm when there is no breakdown:

$$T = \begin{pmatrix} 1 & 0 & 0 & \cdots & & 0 \\ a_0^{[1]} & 1 & 0 & \cdots & & 0 \\ a_0^{[2]} & a_1^{[2]} & 1 & 0 & 0 & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_0^{[n-2]} & \cdots & & & 1 & 0 \\ a_0^{[n-1]} & a_1^{[n-1]} & & \cdots & a_{n-2}^{[n-1]} & 1 \end{pmatrix} \tag{18}$$

where

$$p_k(x) = x^k + \sum_{j=0}^{k-1} a_j^{[k]} \cdot x^j \quad \forall k = 0, \cdots, n$$

Let $J = [\gamma_k, \alpha_k, 1]$ be the tridiagonal matrix obtained in (17).

We have the following

Theorem 4.1 *Let $p_n(x)$ and p_{n-1} be polynomials of degree n with real coefficients. If the Routh-Lanczos's Algorithm produces a complete p.r.s, the matrices J and T defined as above are related by the following similarity transformation:*

$$J = TCT^{-1} \tag{19}$$

where C is the companion matrix $C(p_n)$.

Proof : We follow the proof given in [Bar75] for a similar result. Let $\{x_j\}_{j=1,n}$ be a set of n distinct and nonzero real numbers. The matrix $\mathcal{X}$, defined by $\mathcal{X}_{ij} = {x_j}^{i-1}$, is therefore a nonsingular Vandermonde matrix.

We define, for any nonzero real x, the vector

$$\mathcal{P}(x) = \begin{pmatrix} p_0(x) \\ p_1(x) \\ \vdots \\ p_{n-1}(x) \end{pmatrix} \tag{20}$$

A direct computation gives:

$$J\mathcal{P}(x) = x\mathcal{P}(x) + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -p_n(x) \end{pmatrix} \tag{21}$$

and

$$\mathcal{P}(x) = TX \tag{22}$$

where $X = (1, x, x^2, \cdots, x^{n-1})^t$. Similarly,

$$C(p_n)X = xX + \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -p_n(x) \end{pmatrix}$$

and therefore from (21)

$$TC(p_n)X = x\mathcal{P}(x) + T\begin{pmatrix} 0 \\ \vdots \\ 0 \\ -p_n(x) \end{pmatrix} = J\mathcal{P}(x) = JTX$$

since $(0, \cdots, 0, p_n(x))^t$ is invariant under T.

This is true for any nonzero real x, and hence:

$$JT\mathcal{X} = TC\mathcal{X} \tag{23}$$

which ends the proof. □

To describe the connections of the Routh-Hurwitz algorithm with the Lanczos algorithm, we consider here the version of the non-symmetric Lanczos Algorithm which produces the real tridiagonal matrix with entries set to one on the upper diagonal.

Corollary 4.1 *The relation (19) shows that the matrix J is also obtained by applying the Lanczos algorithm on the matrix C when taking as left and right initial vectors respectively the first row of T and the first column of T^{-1} and for which no breakdown occurs.*

4.3 Ill convergence for the Lanczos algorithm

We can state a result for the non-symmetric case which is much worse than for its symmetric version :

Theorem 4.2 *Let A be a real matrix with real eigenvalues $\{\lambda_i\}_{i=1,\cdots,n}$. For every real $\omega \neq 0$ such that $\omega < \lambda_1$ (or $\omega > \lambda_n$), there exist two initial vectors p_0 and q_0 such that, if J_n is the real tridiagonal matrix obtained by the non-symmetric Lanczos Algorithm (with initial vectors p_0 q_0), we have*

$$\omega \in sp(J_{n-1})$$

where J_{n-1} is the principal submatrix of J_n

Let us first state a lemma.

Lemma 4.1 *Let p and q be two real monic polynomials. We assume that*

- *p is of degree n and its roots $\{\lambda_i\}_{i=1,\cdots,n}$, are simple and real,*
- *q is of degree $n-1$ and its roots $\{\omega_i\}_{i=1,\cdots,n-1}$ are simple and real,*
- *$\omega_1 < \lambda_1 < \omega_2 < \cdots < \omega_{n-1} < \lambda_{n-1} < \lambda_n$ (we call this property the* shifted *interlacing assumption).*

Then, if r is defined as the third polynomial in the sequence obtained by the Routh-Lanczos algorithm applied on (p, q), i.e

$$p(x) = (x-a)q(x) - cr(x) \quad \text{with} \quad \text{degree}(r) \leq n-2$$

then we have

i) r is of degree $n-2$ and its roots are simple and real *ii) the roots $\{\nu_i\}_{1,n-2}$ of r strictly interlace the roots of q.*

Proof : A computation similar to [Hal76] (pp. 67-68) insures that

$$c = \sum_{j=1}^{n-1} \left[(\lambda_{j+1} - \omega_j) \times \sum_{k=1}^{j} (\omega_k - \lambda_k) \right]$$

which proves that c is negative and cannot vanish. Moreover, since the polynomials p and $-cr$ are coincident on $\{\omega_i\}_{1,n-1}$ and since $\{p(\omega_j)\}_{1,j-1}$ is an alternate sequence, the roots of r strictly interlace the roots of q. □

Proof (theorem)

Let us define the polynomials $p_n(x) = det(xI - A)$ and $p_{n-1}(x) = \prod_{i=1}^{n-1}(x - \omega_i)$ where $\{\omega_i\}_{1,n-1}$ is any set of reals satisfying the shifted interlacing assumption:

$$\omega_1 < \lambda_1 < \omega_2 < \cdots < \omega_{n-1} < \lambda_{n-1} < \lambda_n$$

and where $\omega_1 = \omega$. Therefore by applying the previous lemma, we can claim that the first step of the Routh-Lanczos algorithm defines a polynomial p_{n-2} of which roots strictly interleave the set $\{\omega_i\}_{1,n-1}$. We are now in the situation where the Routh-Lanczos algorithm with the polynomials p_{n-1} and p_{n-2} computes a complete p.r.s $p_{n-1}, p_{n-2}, \cdots, p_1, p_0$ of order $n-1$ which is a Sturm sequence. We notice that the p.r.s $p_n, p_{n-1}, \cdots, p_1, p_0$ of order $n-1$ is also complete, but is not a Sturm sequence. Let J_n the obtained tridiagonal matrix and T the triangular matrix containing the coefficients of the polynomials of the sequence. Theorem (4.1) insures that the matrices satisfy $J_n = TC(p_n)T^{-1}$.

Let X be a matrix such that $XAX^{-1} = C(p_n)$. The last two equalities imply that $J_n = (TX)A(TX)^{-1}$.

Now, we can conclude by defining the two vectors p_0 and q_0 equal to, respectively, the first column of TX and the first row of $(TX)^{-1}$ as the starting vectors for the Lanczos process. □

This result illustrates that, when computing the eigenvalues of a matrix, the non-symmetric Lanczos Algorithm can lead to a slow convergence, even for a symmetric matrix since a peripherical eigenvalue of the tridiagonal matrix of order $n-1$ can be arbitrarily far from the spectrum of the original matrix. This poor behavior of the non-symmetric Lanczos algorithm is not shared by the Arnoldi algorithm since for the latter the field of values of the Hessenberg matrices are always included in the field of values of the original matrix. Despite of that, Lanczos algorithms are still attractive for their very cheap computational cost.

5 Parametric Representations of a Jacobi matrix

All the previous results lead to the problem of *parametric representations of a Jacobi matrix* and, more generally, of a tridiagonal matrix. This problem is defined and studied in [Par92]. We summarise here some of the results.

Theorem 5.1 *Let $J_n = [\beta_k, \alpha_k, \beta_{k+1}]$ a real Jacobi matrix of order n.*

1. *Given the n eigenvalues $\{\lambda_i\}_{i=1\cdots n}$ of J_n and the $n-1$ eigenvalues $\{\omega_i\}_{i=1\cdots n-1}$, of J_{n-1}, principal submatrix of J_n (or those of any principal submatrix or order $n-1$), if the eigenvalues of J_{n-1} interlace those of J_n, then J_n is uniquely determinated.*

2. *Given the polynomials $p_n = det(x \cdot I - J_n)$ and $p_{n-1} = det(xI_{n-1} - J_{n-1})$, if the roots of p_{n-1} interlace strictly the roots of p_n, then J_n is uniquely determinated.*

3. *Let B a matrix similar to the Jacobi matrix J_n defined by*

$$Q^{-1}BQ = J_n.$$

   Then Q and J_n are determinated, to within diagonal scaling by the first (or last) column of Q and the first (or last) row of Q^{-1}.

4. *Given the n eigenvalues $\{\lambda_i\}_{i=1J\cdots n}$ of J_n and the last row of its orthogonal matrix of eigenvectors, then J_n is uniquely determinated.*

5. *Given the n eigenvalues $\{\lambda_i\}_{i=1\cdots n}$ of J_n and the $n-1$ values $\{e_1^T J_n^k e_1\}_{k=1}^{n-1}$, where $e_1^T = (1, 0, \cdots, 0)^T$, then J_n is uniquely determinated.*

The first three determinations are consequences of the previous sections, the fourth determination is cited in [HP99], and the last is a special case of theorem 1 in [MH81]. We can point out the fact that the five results involve $2n-1$ free parameters by considering appropriate normalisations.

6 Conclusion

This paper deals with two classes of finite algorithms that solve a special structured inverse eigenvalue problem(SIEP) : given some constraints on a set of real eigenvalues, find a Jacobi matrix J (real symmetric tridiagonal) that admits eigenvalues satisfying the constraints. We have called JIEP (Jacobi Inverse Eigenvalue Problem) the case where the two sets of given eigenvalues satisfy the interlacing property (3) and TIEP (Tridiagonal Inverse Eigenvalue Problem) the problem obtained when the interlacing property is relaxed.

The two classes of finite algorithms that have been considered are:

- a polynomial algorithm, based on a special Euclid-Sturm algorithm (Householder's terminology) which has been rediscovered several times.

- a matrix algorithm, which is a symmetric Lanczos algorithm with a special choice of the starting vector.

Hald's theorem insures the equivalence of the two algorithms in exact arithmetic.

We have extended the results of the symmetric situation to the non-symmetric case by considering general real polynomials. Possible breakdowns may occur in the polynomial algorithm as it may happen with the non-symmetric Lanczos algorithm. The connection between the two algorithms exhibits a similarity transformation from the Frobenius matrix to the tridiagonal matrix.

This result has been used to illustrate the fact that, when computing the eigenvalues of a matrix, the non-symmetric Lanczos Algorithm can lead to a slow convergence, even on a symmetric matrix.

Acknowledgement

The authors would like to thank Dr M.H. Gutknecht for the reading of a preliminary version of this work and for some references.

References

[AG91] G. Ammar and W.B. Gragg. $o(n^2)$ reduction algorithms for the construction of a band matrix form spectral data. *Siam J. Matrix Anal. Appl.*, 12:426–431, 1991.

[Bar74] S. Barnett. A new look at classical algorithms for polynomial resultant and gcd calculation. *SIAM Rev.*, 16:193–206, 1974.

[Bar75] S. Barnett. A companion matrix analogue for orthogonal polynomials. *Lin. Alg. Appli.*, 12:197–208, 1975.

[BG77] D.L. Boley and G.H. Golub. Inverse eigenvalue problems for band matrices. *Lecture Notes in Mathematics*, 1977. (Berlin:Springer).

[BG78] C. De Boor and G.H. Golub. The numerically stable reconstruction of a jacobi matrix from spectral data. *Lin. Alg and Appl.*, 21:245–260, 1978.

[BG87] D.L. Boley and G.H. Golub. A survey of matrix inverse eigenvalue problems. *Inverse Problems*, 3:595–622, 1987.

[BK81] F. W. Biegler-Konig. Construction of the band matrices from spectral data. *Lin. Alg. and Apll.*, 40:79–84, 1981.

[BOFKT88] M. Ben-Or, M. Feig, D. Kozen, and P. Tiwari. A fast parallel algorithm for determining all roots of a polynomial with real roots. *SIAM J. Comput.*, 17:1081–1092, 1988.

[BOT90] M. Ben-Or and P. Tiwari. Simple algorithms for approximating all roots of a polynomial with real roots. *J. of Complex.*, 6:417–442, 1990.

[BS77] S. Barnett and D.D Siljak. Routh's algorithm, a centennial survey. *SIAM Review*, 19:472–489, 1977.

[Che58] P. Chebyshev. Sur les fractions continues. *J. Math. Pures Appl. S ri. II*, 3:289–323, 1858.

[Che59] P. Chebyshev. Sur l'interpolation par la méthode des moindres carrés. *Mém. Acad. Imp. des Sci. St. Petersbourg, série 7*, 1:1–24, 1859.

[Col67] G. E. Collins. Subresultants and reduced polynomial remainder sequence and determinants. *J. ACM*, 14:128–142, 1967.

[DH56] A.C. Downing and A.S Householder. Some inverse characteristic value problem. *J. ACM*, 3:203–207, 1956.

[Fie90] M. Fiedler. Expressing a polynomial as the caracteristic polynomial of a symetric matrice. *Lin. Alg. and Appl.*, 141:265–270, 1990.

[FNO87] S. Friedland, J. Nocedal, and M.L Overton. The formulation and analysis of numerical methods for inverse eigenvalue problems. *Siam J. Num. Anal.*, 24(3):634–667, 1987.

[Fri77] S. Friedland. Inverse eigenvalue problem. *Lin. Alg. and Applic.*, 17:15–51, 1977.

[Fri79] S. Friedland. The reconstruction of a symmetric matrix from the spectral data. *J. of Math. Anal. and Appl.*, 71:412–422, 1979.

[Gan59] Gantmacher. *La théorie des matrices*, volume 2. Dunod, Paris, 1959.

[Gau82] W. Gautschi. On generating orthogonal polynomials. *Siam J. Sc. Stat. Comput.*, 3:289–317, 1982.

[Gau85] W. Gautschi. Orthogonal polynomials-constructive theory and applications. *J. Comp. Apll. Math.*, 12:61–76, 1985.

[GG94] M. H. Gutknecht and W. B. Gragg. Stable look-ahead versions of the euclidean and chebyshev algorithms. Technical report, IPS-ETH Zurich, March 1994.

[GH84] W.B. Gragg and W.J. Harrod. The numerically stable reconstruction of jacobi matrices from spectral data. *Num. Math.*, 44:317–335, 1984.

[GK60] F. R. Gantmacher and M.G. Krein. *Oszillationsmatrizen, Oszillationskerne und kleine Schwingungen mecanischer Systeme*. Akademie-Verlag, Berlin, 1960.

[GW76] L.J. Gray and D.G. Wilson. Construction of a jacobi matrix from spectral data. *Lin. Alg. and Appl.*, 14:131–134, 1976.

[Hal76] O. Hald. Inverse eigenvalue problems for jacobi matrix. *Lin. Alg. Applic.*, 14:63–85, 1976.

[Hoc67] H. Hochsdtadt. On some inverse problems in matrix theory. *Archiv. der Math.*, 18:201–207, 1967.

[Hoc74] H. Hochsdtadt. On the construction of a jacobi matrix from spectral data. *Lin. Alg. Applic.*, 8:435–446, 1974.

[Hou64] Householder. *The theory of matrices in numerical analysis.* Blaisdell, 1964.

[Hou74] A.S. Householder. Bigradiants and the euclid-sturm algorithm. *Siam Review*, 16(2):207–213, 1974.

[HP99] R.O. Hill and B.N. Parlett. Refined interlacing properties. *Siam J.*, 99:239–247, 1999.

[Jac74] Jacobson. *Basic Algebra I*, volume 2. Freeman, San Fransisco, 1974.

[KN81] M. Krein and M. Naimark. The method of symmetric and hermitian forms in the theory of the separation of the roots of algebraic equations. *Lin. and Multil. Alg.*, 10:265–308, 1981.

[Knu81] Donald E. Knuth. *Seminumerical Algorithms*, volume 2 of *The Art of Computer Programming.* Addison-Wesley, Reading, Massachusetts, second edition, 10 January 1981.

[Lan50] C. Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Res. Nat. Bur. of Stand.*, 45:255–281, 1950.

[MH81] M. Mattis and H. Hochstadt. On the construction of band symmetric matrices from spectral data. *Lin. Alg. and App.*, 38:109–119, 1981.

[Mig91] M. Mignotte. *Mathematics for Computer Algebra.* Springer-Verlag, 1991.

[Par92] B.N. Parlett. Reduction to tridiagonal form and minimal realization. *Siam J. MAtrix Anal. and Appl.*, 13(2):567–597, 1992.

[PTL85] Parlett, Taylor, and Liu. A look ahead lanczos algorithm for unsymmetric matrices. *Math. Comp.*, 44(169):105–124, 1985.

[Rei91] L. Reichel. Fast qr decomposition of vandermonde-like matrices and polynomial least squares approximation. *Siam J. Matrix Anal. Appl.*, 12(3):552–564, 1991.

[Rut63] H. Rutishauser. On jacobi rotation patterns. In *Experimental Arithmetic, High Speed Computing and Mathematics*, Providence, 1963. Proc. Symp. Appl. Math. 15 Amer. Math. Soc.

[Sch56] H.R. Schwarz. Ein verfahren zur stabilitatsfrage bei matrizen-eigenwerte-problem. *Z. Angw. Math. Phys.*, 7:473–500, 1956.

[Sch93] G. Schmeisser. A real symmetric tridiagonal matrix with a given characteristic polynomial. *Lin. and Mult. Alg.*, 193:11–18, 1993.

[Sco79] D.S. Scott. How to make the lanczos algorithm converge slowly. *Math. Comp.*, 33(145):239–247, 1979.

[Sti84] T. J. Stieltjes. Quelques recherches sur la théorie des quadratures dites mécaniques. *Ann. Sci. Ecole Norma. Paris Séri. 3*, 1:409–426, 1884.

[Stu35] J. Sturm. Mémoires présentés par divers savants. pages 271+, 1835.

[Wal48] H.S. Wall. *Analytic theory of Continued Fractions.* Chelsea, N.Y, 1948.

[Wen61] B. Wendroff. On orthogonal polynomials. *Proc. Amer. Math. Soc.*, 12:554–555, 1961.

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 46 avenue Félix Viallet, 38031 GRENOBLE Cedex 1
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
ISSN 0249-6399