Università di Pisa

Dipartimento di Informatica

Technical Report : TR-96-13

A Fast Algorithm for Hankel Matrices Represented in Orthogonal Polynomial Bases

L. Gemignani

May 16, 1996

ADDR: Corso Italia 40,56125 Pisa,Italy. TEL: +39-50-887111. FAX: +39-50-887226

A Fast Algorithm for Hankel Matrices Represented in Orthogonal Polynomial Bases

Luca Gemignani *

Dipartimento di Informatica, Università di Pisa
Corso Italia 40, 56125 Pisa, Italy.

ABSTRACT

Consider a $n \times n$ lower triangular matrix L whose $(i+1)$-st row is defined by the coefficients of the real polynomial $p_i(x)$ of degree i such that $\{p_i(x)\}$ is a set of orthogonal polynomials satisfying a standard three-term recurrence relation. If H is a $n \times n$ real Hankel matrix with nonsingular leading principal submatrices, then $\widehat{H} = LHL^T$ will be referred as a strongly nonsingular Hankel matrix with respect to the orthogonal polynomial basis $\{p_i(x)\}$. In this paper we develop an efficient $O(n^2)$ algorithm for the solution of a system of linear equations with a real symmetric coefficient matrix $\widehat{H}$ which is a Hankel matrix with respect to a suitable orthogonal polynomial basis. We then apply our method to the solution of a weighted finite moment problem.

---

1. INTRODUCTION

In this article we present a new efficient algorithm for the solution of a linear system with a real symmetric coefficient matrix $\widehat{H_{n-1}}$ which is a strongly nonsingular Hankel matrix with respect to an orthogonal polynomial basis. This means that $\widehat{H_{n-1}} = L_{n-1}H_{n-1}L_{n-1}^T$, where $H_{n-1} = (h_{i+j-2})$ is a $n \times n$ real strongly nonsingular Hankel matrix and $L_{n-1} = (l_{i,j})$ is a real $n \times n$ lower triangular matrix such that the polynomials $p_{i-1}(x) = (l_{i,1}, \ldots, l_{i,i})(1, x, \ldots, x^{i-1})^T$ satisfy a three-term recurrence

$$p_0(x) = \alpha_0, \quad p_1(x) = \alpha_1 x - \beta_1$$
$$p_i(x) = (\alpha_i x - \beta_i)p_{i-1}(x) - \gamma_{i-1}p_{i-2}(x),$$

*Supported by 40% funds of the Progetto Analisi numerica e matematica computazionale of MURST and by G.N.I.M. of C.N.R..

655 Avenue of the Americas, New York, NY 10010 0024- 3795/93/$6.00

with $\alpha_{i-1} > 0$ and $\gamma_i > 0$ for $i \geq 1$.

The derivation of the new algorithm relies on a general approach to the solution of linear systems with structured coefficient matrices (see [11], [12], [17] and [18]). In fact, we first introduce a suitable displacement operator $\mathcal{K}: R^{n\times n} \to R^{n\times n}$ having the form

$$\mathcal{K}(X) = T_n X - X T_n^T$$

where T_n is a real tridiagonal matrix associated with the sequence $\{p_i(x)\}$. This matrix has the property that the characteristic polynomial of its left $i \times i$ principal submatrix T_i coincides with the monic polynomial associated with $p_i(x)$, that is,

$$\det(xI - T_i) = \frac{p_i(x)}{\alpha_0 \alpha_1 \ldots \alpha_i}.$$

Since $\mathcal{K}(\widehat{H_{n-1}})$ is a rank-two matrix, we are then able to construct recursively the vectors ρ_i which satisfy $\widehat{H_{i-1}}^{-1} e_i = \rho_i$ where e_i denotes the last column of the $i \times i$ identity matrix. The resulting algorithm, which will be presented in Section 2, takes $5.5n^2 + O(n)$ multiplications (divisions) and $O(n)$ storage. Moreover, by using a standard technique [11], the solution of the equation $\widehat{H_{n-1}} x = b$ can be obtained as soon as the vectors ρ_i, $i = 1, \ldots, n$, are found at the overall computational cost of $6.5n^2 + O(n)$ multiplications and $O(n)$ storage.

Hankel matrices with respect to an orthogonal polynomial basis appear naturally in the solution of some finite weighted Housdorff moment problems [15]. Consider the solution of the following first-kind integral equation with discrete data,

$$Mf = \mu = (\mu_1, \ldots, \mu_n)^T, \quad \mu_k = \int_0^1 p_{k-1}(x) f(x) w(x) dx.$$

If H_{n-1} denotes the positive definite Hankel matrix with the moments $\int_0^1 x^{i+j-2} w(x) dx$ along its antidiagonals, then solving the linear system

$$\widehat{H_{k-1}} x_k = L_k H_{k-1} L_k^T x_k = (\mu_1, \ldots, \mu_k)^T, \quad k = 1, \ldots, n,$$

amounts to compute the coefficients of a polynomial $f_k(x)$ of degree $k-1$, expressed in the orthogonal basis $\{p_i(x)\}$, which approximates the unknown function $f(x)$. The matrix $\widehat{H_{n-1}}$ can be much better conditioned than the Hankel matrix H_{n-1} [7] and, therefore, the transition from the modified moment problem to the classical moment problem, although possible from a theoretical point of view, is to be avoided. In section 3 we will report numerical examples which illustrate the accuracy of our algorithm when it is used for solving a finite weighted Housdorff moment problem.

2. DERIVATION OF THE ALGORITHM

Let us consider a nonsingular Hankel matrix

$$H_{n-1} = \begin{pmatrix} h_0 & h_1 & \dots & h_{n-1} \\ h_1 & \iddots & \iddots & h_n \\ \vdots & \iddots & \iddots & \vdots \\ h_{n-1} & h_n & \dots & h_{2n-2} \end{pmatrix}. \tag{2.1}$$

where we assume that $\det H_k \neq 0$ for all k, $0 \leq k \leq n-1$, i.e., H_{n-1} is a strongly nonsingular Hankel matrix.

Let $\{p_k(x)\}$ be a polynomial sequence such that $p_k(x)$ is a real polynomial of degree k which satisfies a three-term recurrence

$$p_0(x) = \alpha_0, \quad p_1(x) = \alpha_1 x - \beta_1 \tag{2.2}$$

$$p_k(x) = (\alpha_k x - \beta_k)p_{k-1}(x) - \gamma_{k-1}p_{k-2}(x) \tag{2.3}$$

with $\alpha_{k-1} > 0$ and $\gamma_k > 0$ for $k \geq 1$. It is well known that $p_k(x)$ are orthogonal polynomials with respect to a suitable scalar product $\langle \cdot, \cdot \rangle$; moreover, they coincide with the appropriately normalized characteristic polynomials of the leading principal submatrices of a real tridiagonal matrix T_n, that is,

$$\det(xI - T_k) = \frac{p_k(x)}{\alpha_0 \alpha_1 \dots \alpha_k}, \quad k \geq 1,$$

where

$$T_k = \begin{pmatrix} \frac{\beta_1}{\alpha_1} & \frac{1}{\alpha_1} & & \\ \frac{\gamma_1}{\alpha_2} & \frac{\beta_2}{\alpha_2} & \ddots & \\ & \ddots & \ddots & \frac{1}{\alpha_{k-1}} \\ & & \frac{\gamma_{k-1}}{\alpha_k} & \frac{\beta_k}{\alpha_k} \end{pmatrix}.$$

The matrix T_n reduces to a symmetric one if the condition $\langle p_i(x), p_i(x) \rangle = \langle p_j(x), p_j(x) \rangle$ holds for any i and j. This condition is clearly satisfied when $p_i(x)$ are orthonormal polynomials.

Let us denote as L_k the $(k+1) \times (k+1)$ lower triangular matrix such that

$$L_k(1, x, \dots, x^k)^T = (p_0(x), \dots, p_k(x))^T.$$

It can be easily seen that

$$L_{n-1}^{-1} T_n L_{n-1} = F(p_n)$$

where

$$F(u) = \begin{pmatrix} 0 & 1 & & & \\ & \ddots & \ddots & & \\ & & 0 & 1 \\ -\dfrac{u_0}{u_n} & \dots & \dots & -\dfrac{u_{n-1}}{u_n} \end{pmatrix}$$

represents the Frobenius matrix associated with the polynomial $u(x) = \sum_{i=0}^{n} u_i x^i$, $u_n \neq 0$.

In different contexts many authors [7], [9], [10], [13], [21] have recently considered the problem of solving a linear system with a coefficient matrix which has the form of the the modified Hankel matrix †

$$\widehat{H_{n-1}} = L_{n-1} H_{n-1} L_{n-1}^T.$$

Since H_{n-1} is nonsingular, then there exists an unique solution to the following linear system:

$$H_{n-1}(u_0^{(n)}, \dots, u_{n-1}^{(n)})^T = -(h_n, \dots, h_{2n-1})^T$$

where h_{2n-1} is any number. If we write

$$u_n(x) = \sum_{i=0}^{n-1} u_i^{(n)} x^i + x^n,$$

then it has been proved [19] that

$$F(u_n) H_{n-1} = H_{n-1} F(u_n)^T. \tag{2.4}$$

By replacing

$$F(u_n) = F(p_n) - e_n \delta_n^T,$$

in (2.4) we find that

$$T_n \widehat{H_{n-1}} - \widehat{H_{n-1}} T_n^T = L_{n-1}(F(p_n) H_{n-1} - H_{n-1} F(p_n)^T) L_{n-1}^T$$

†An anonymous referee pointed out that fast algorithms for modified Hankel matrices have been also described in: G. Heinig, W. Hoppe, K. Rost, Structured matrices in interpolation and approximation problems, Wissenschaftl. Zeitschrift der TU Karl-Marx-Stad 31, 2 (1989), 196-202.

which implies

$$T_n \widehat{H_{n-1}} - \widehat{H_{n-1}} T_n^T = L_{n-1}(e_n \delta_n^T H_{n-1} - H_{n-1}\delta_n e_n^T) L_{n-1}^T.$$

In this manner we obtain that

$$T_n \widehat{H_{n-1}} - \widehat{H_{n-1}} T_n^T = e_n \tilde{\delta}_n^T \widehat{H_{n-1}} - \widehat{H_{n-1}} \tilde{\delta}_n e_n^T,$$

or, equivalently,

$$\mathcal{K}(\widehat{H_{n-1}}) = e_n \hat{\delta}_n^T - \hat{\delta}_n e_n^T. \tag{2.5}$$

By considering the $k \times k$ leading principal submatrix of $\mathcal{K}(\widehat{H_{n-1}})$, $1 \le k \le n-2$, then there follows that

$$T_k \widehat{H_{k-1}} - \widehat{H_{k-1}} T_k^T = \frac{1}{\alpha_k}\{(\hat{h}_k, \ldots, \hat{h}_{2k-1})^T e_k^T - e_k(\hat{h}_k, \ldots, \hat{h}_{2k-1})\},$$

where e_k denotes the last column of the $k \times k$ identity matrix. Hence, we can write

$$\widehat{H_{k-1}}^{-1} T_k - T_k^T \widehat{H_{k-1}}^{-1} = \frac{1}{\alpha_k} \widehat{H_{k-1}}^{-1} \{r_k e_k^T - e_k r_k^T\} \widehat{H_{k-1}}^{-1}, \tag{2.6}$$

where we set $r_k = (\hat{h}_k, \ldots, \hat{h}_{2k-1})^T$.

Now, let us introduce the vectors $\rho_k = (\rho_1^{(k)}, \ldots, \rho_k^{(k)})^T$, $1 \le k \le n$, defined by

$$\widehat{H_{k-1}}^{-1} e_k = \rho_k.$$

From

$$\widehat{H_k}^{-1} \frac{e_{k+1}}{\rho_{k+1}^{(k+1)}} = (\frac{\rho_1^{(k+1)}}{\rho_{k+1}^{(k+1)}}, \ldots, \frac{\rho_k^{(k+1)}}{\rho_{k+1}^{(k+1)}}, 1)^T,$$

we obtain that

$$\widehat{H_{k-1}}^{-1} r_k = -(\frac{\rho_1^{(k+1)}}{\rho_{k+1}^{(k+1)}}, \ldots, \frac{\rho_k^{(k+1)}}{\rho_{k+1}^{(k+1)}})^T. \tag{2.7}$$

In the view of relation (2.6), this implies that

$$\widehat{H_{k-1}}^{-1} T_k e_k - T_k^T \rho^{(k)} = \frac{1}{\alpha_k}\{\frac{\rho_k^{(k+1)}}{\rho_{k+1}^{(k+1)}} \rho_k - \frac{\rho_k^{(k)}}{\rho_{k+1}^{(k+1)}} (\rho_1^{(k+1)}, \ldots, \rho_k^{(k+1)})^T\}.$$

Since we have

$$T_k e_k = \frac{\beta_k}{\alpha_k} e_k + \frac{1}{\alpha_{k-1}} (e_{k-1}^T, 0)^T, \quad k > 1,$$

we can write

$$\begin{aligned} &\frac{1}{\alpha_{k-1}} \widehat{H_{k-1}}^{-1} (e_{k-1}^T, 0)^T + \frac{\beta_k}{\alpha_k} \rho_k - T_k^T \rho_k \\ = &\frac{1}{\alpha_k} \{ \frac{\rho_k^{(k+1)}}{\rho_{k+1}^{(k+1)}} \rho_k - \frac{\rho_k^{(k)}}{\rho_{k+1}^{(k+1)}} (\rho_1^{(k+1)}, \ldots, \rho_k^{(k+1)})^T \}. \end{aligned}$$

Replacing now $\widehat{H_{k-1}}^{-1} (e_{k-1}^T, 0)^T$ by

$$\chi^{(k)} = ({\rho_{k-1}}^T, 0)^T - a_k \rho_k, \quad k > 1,$$

for a certain a_k, produces

$$\frac{1}{\alpha_k} \frac{\rho_k^{(k)}}{\rho_{k+1}^{(k+1)}} (\rho_1^{(k+1)}, \ldots, \rho_k^{(k+1)})^T - \frac{1}{\alpha_{k-1}} ({\rho_{k-1}}^T, 0)^T \tag{2.8}$$

$$= (\frac{1}{\alpha_{k-1}} a_k - \frac{\beta_k}{\alpha_k} + \frac{1}{\alpha_k} \frac{\rho_k^{(k+1)}}{\rho_{k+1}^{(k+1)}} + T_k^T) \rho_k, \quad k > 1. \tag{2.9}$$

Comparison of ${\rho_{k+1}}^T \widehat{H_k} \chi^{(k+1)}$ and ${\chi^{(k+1)}}^T \widehat{H_k} \rho_{k+1}$ reveals that a_{k+1} is given by

$$a_{k+1} = - \frac{\rho_k^{(k+1)}}{\rho_{k+1}^{(k+1)}}.$$

Finally, we may determine a_{k+1} by means of

$$e_{k+1}^T \widehat{H_k} (\rho_k^T, 0)^T = a_{k+1}. \tag{2.10}$$

In order to perform efficiently these computations, we may observe that equality (2.8-9) can be rewritten in the following way:

$$\begin{aligned} &(\frac{1}{\alpha_{k-1}} a_k - \frac{\beta_k}{\alpha_k} + \frac{1}{\alpha_k} \frac{\rho_k^{(k+1)}}{\rho_{k+1}^{(k+1)}} + T_n^T) \begin{pmatrix} \rho_k \\ 0 \end{pmatrix} \\ = &\frac{1}{\alpha_k} \frac{\rho_k^{(k)}}{\rho_{k+1}^{(k+1)}} \begin{pmatrix} \rho_{k+1} \\ 0 \end{pmatrix} - \frac{1}{\alpha_{k-1}} \begin{pmatrix} \rho_{k-1} \\ 0 \end{pmatrix}, \quad k > 1, \end{aligned}$$

where we extend the vector ρ_k to dimension n by adding zeros. For $k = 1$, we have

$$(-\frac{\beta_1}{\alpha_1} + \frac{1}{\alpha_1} a_2 + T_n^T) \begin{pmatrix} \rho_1 \\ 0 \end{pmatrix} = \frac{1}{\alpha_1} \frac{\rho_1^{(1)}}{\rho_2^{(2)}} \begin{pmatrix} \rho_2 \\ 0 \end{pmatrix}. \qquad (2.11)$$

By means of relation (2.5), we find that

$$(\frac{1}{\alpha_{k-1}} a_k - \frac{\beta_k}{\alpha_k} + \frac{1}{\alpha_k} \frac{\rho_k^{(k+1)}}{\rho_{k+1}^{(k+1)}} - e_n \hat{\delta}_n^T + T_n) \widehat{H_{n-1}} \begin{pmatrix} \rho_k \\ 0 \end{pmatrix}$$
$$= \frac{1}{\alpha_k} \frac{\rho_k^{(k)}}{\rho_{k+1}^{(k+1)}} \widehat{H_{n-1}} \begin{pmatrix} \rho_{k+1} \\ 0 \end{pmatrix} - \frac{1}{\alpha_{k-1}} \widehat{H_{n-1}} \begin{pmatrix} \rho_{k-1} \\ 0 \end{pmatrix}, \quad k > 1,$$

and, therefore, we are able to compute the vectors $\hat{H}_{n-1}(\rho_k, 0)^T$ in a recursive way using only $O(n)$ storage once we know the first and the last two columns of the matrix $\hat{H}_{n-1}$.

The resulting algorithm for determining the vectors ρ_k, $k = 1, \ldots, n$, proceeds as follows:

Algorithm 1

Input: the entries of the tridiagonal matrix T_n associated with the sequence of the real polynomials $p_i(x)$; the entries of the first and the last two columns of a real strongly nonsingular Hankel matrix $\widehat{H_{n-1}}$ with respect to the polynomial basis $\{p_i(x)\}$.

Output: the entries of the vectors ρ_k, $k = 1, \ldots, n$, which satisfy $\widehat{H_{k-1}}^{-1} e_k = \rho_k$.

- Determine $\hat{\delta}$ by means of (2.5) (observe that its last entry can be choosen arbitrarily). Compute $\rho_1^{(1)} = 1/\hat{h}_{1,1}$ and ρ_2 by means of (2.11).
- Compute recursively for $k = 2, \ldots, n-1$:

1. evaluate a_{k+1} by means of (2.10);
2. find the vector $\rho_{k+1}/\rho_{k+1}^{(k+1)}$ by means of (2.8-9);
3. determine $\rho_{k+1}^{(k+1)}$ by means of $\hat{H}_k(\rho_{k+1}/\rho_{k+1}^{(k+1)}) = e_{k+1}/\rho_{k+1}^{(k+1)}$.

The number of multiplications and divisions performed at step k of the Algorithm 1 is $11k + O(1)$. The initialization phase has the cost of

$O(n)$ arithmetical operations. Therefore, the number of multiplications and divisions in the whole algorithm is $5.5n^2 + O(n)$ and, moreover, it can be implemented using only $O(n)$ storage.

Algorithm 1 generalizes some well known fast procedures for solving Hankel linear systems (see [8] and the references given there about Levinson-type algorithm for Hankel matrices). In fact, in the case where $\alpha_{i-1} = 1$, $\beta_i = 0$ and $\gamma_i = 0$ $i = 1, \ldots, n$, then the corresponding polynomials (2.2-3) become just $p_i(x) = x^i$ — which do not constitute an orthogonal set — and, moreover, T_n reduces to the shift operator $P_n = (p_{i,j})$ with $p_{i,j} = \delta_{i,j-1}$.

Now the solution of the equation $\widehat{H_{n-1}}x = b$, $b = (b_1, \ldots, b_n)^T$, can be obtained from the following classical result [11].

Proposition 2.1. *The matrix $\widehat{H_{n-1}}$ is strongly nonsingular if and only if the equations*

$$\widehat{H_{k-1}}\rho_k = e_k, \quad k = 1, \ldots, n,$$

are solvable. In this case $e_k^T \rho_k \neq 0$, $k = 1, \ldots, n$. Moreover, if the vectors z_k are defined recursively by means of

$$\begin{aligned} z_1 &= b_1/\hat{h}_{11} \\ \eta_k &= b_k - e_k^T \widehat{H_{k-1}}(z_{k-1}^T, 0)^T \\ z_k &= (z_{k-1}^T, 0)^T + \eta_k \rho_k, \end{aligned}$$

then we find that $\widehat{H_{n-1}}z_n = b$.

The above recursions take $n^2 + O(n)$ multiplications and, therefore, we are able to solve a linear system with coefficient matrix $\widehat{H_{n-1}}$ at the overall computational cost of $6.5n^2 + O(n)$ multiplications (divisions) and $O(n)$ storage.

3. AN APPLICATION TO A MODIFIED MOMENT PROBLEM

Finite moment problems are well known and a wide literature exists containing numerical methods for solving these problems (see [15], [20] and [22] for general references). Specifically, we are interested in a modified weighted finite moment problem which consists of solving the following first-kind integral equation

$$Mf = \mu = (\mu_1, \ldots, \mu_n)^T, \quad (Mf)_k = \int_0^1 p_{k-1}(x) f(x) w(x) dx, k = 1, \ldots, n,$$

where the polynomials $p_{k-1}(x)$ are the orthonormal Legendre polynomials in $[0,1]$ [1]. These polynomials satisfy the following three-term recurrence relation

$$p_0(x) = 1, \quad p_1(x) = 2\sqrt{3}x - \sqrt{3} \tag{3.1}$$

$$p_{i+1}(x) = \frac{(2i+1)\sqrt{2i+3}}{(i+1)\sqrt{2i+1}}(2x-1)p_i(x) - \frac{i\sqrt{2i+3}}{(i+1)\sqrt{2i-1}}p_{i-1}(x). \tag{3.2}$$

The vector μ is the known vector of the modified moments, $w(x)$ is a positive weight on $[0.1]$ and $f(x)$ is an unknown function. Now, if we are looking for a polynomial approximation

$$f_k(x) = \sum_{i=0}^{k} f_i^{(k)} p_k(x), \quad k = 0, \ldots, n-1$$

of $f(x)$, then we find that the vectors $f_k = (f_0^{(k)}, \ldots, f_k^{(k)})^T$ are the solution of the following linear systems

$$\widehat{H_k} f_k = (\mu_1, \ldots, \mu_{k+1})^T, \quad k = 0, \ldots, n-1. \tag{3.3}$$

The matrix $\widehat{H_k}$ has the form of a modified Hankel matrix, that is,

$$\widehat{H_k} = L_k H_k L_k^T$$

where H_k is the $(k+1) \times (k+1)$ leading principal submatrix of the $n-$th order positive definite Hankel matrix H_{n-1} with the moments

$$\int_0^1 x^{i+j-2} w(x) dx$$

of $w(x)$ along its antidiagonals and, moreover, L_k is the $(k+1)\times(k+1)$ lower triangular matrix defined by the coefficients of the orthonormal Legendre polynomials $p_i(x)$ on $[0,1]$. It is well known that L_k is the lower triangular factor of the Cholesky decomposition of the inverse of the Hilbert matrix of order $k+1$ [22].

The matrix H_k is usually ill-conditioned already for small values of k [23] so that the solution of a linear system with coefficient matrix H_k should be avoided. However, in [7] D. Fasino proved that L_k is a good preconditioner for H_k in the sense that

$$cond(\widehat{H_k}) = cond(L_k H_k L_k^T) < \frac{2k+1}{l}, \quad \inf_{x\in[0,1]} w(x) = l > 0$$

where $cond(A)$ represents the spectral condition number of the matrix A, that is, the ratio between its largest and smallest singular value.

There is a wide recent literature reporting attempts to stabilize the fast algorithms for solving linear structured systems. Most of these attempts are based on the key observation that, from the stability point of view, the available fast methods for Hankel and Toeplitz matrices perform roughly like Gaussian elimination without pivoting. Cabay and Meleshko [5] reported a full discussion about the stability problems of the Levinson-type algorithms for Hankel matrices. Further, they developed the first weakly stable algorithm for inverting Hankel matrices by considering a block version of the Gaussian elimination. Bini and Gemignani [2], [3], showed that a sort of block Gaussian elimination can be applied in order to factorize the inverses of Hankel matrices without destroying the structure properties. Freund and Zha [8] observed that the Levinson-type algorithms for Hankel matrices perform poorly when a small pivot is encountered in Gaussian elimination. In order to overcome these numerical difficulties, they proposed to compute a "well-conditioned" sequence of the so-called fundamental vectors by jumping from a well-conditioned leading principal submatrix to another one. Similar techniques have been also promoted by Chan and Hansen [6], Gutknecht [16], and Bojanczyk and Heinig [4]. Although rigorous stability analyses of the proposed algorithms are missing, the reported numerical results indicate that they have a stable behaviour when applied to the solution of well-conditioned problems. In our context, since all of the leading principal submatrices of $\widehat{H_{n-1}}$ are well-conditioned, then the Freund and Zha approach reduces to Algorithm 1. In fact, concerning the computation of the triangular decomposition $\widehat{H_{n-1}} = U^T DU$, we observe that relation (2.7) easily yields that the size of the off-diagonal elements of U is bounded. Indeed, this is the key for proving the numerical stability of Gaussian elimination.

According to these observations, we conjecture that our method is numerically stable when applied to the solution of (3.3). To substantiate our belief numerically we have performed some numerical experiments by using Mathematica$^{\mathrm{TM}}$ on Macintosh with machine-precision arithmetic, i.e., about 18 decimal digits. Below, we report the results concerning the following 4 weight functions on $[0, 1]$:

1. $w_1(x) = (x + 2)^2(2 - x)^3$;
2. $w_2(x) = \sqrt{1 + x^2}$;
3. $w_3(x) = 1$ if $0 \leq x \leq 1/2$ and $w_3(x) = 1/2$ otherwise;
4. $w_4(x) = (2 - x)(sin^2(2\pi x) + 1)$.

For each weight $w_l(x)$ we have then applied Algorithm 1 to the 12×12

modified Hankel matrix $\widehat{H_{11,l}} = (h_{i,j}^{(l)})$ defined by

$$h_{i,j}^{(l)} = \int_0^1 p_{i-1}(x)p_{j-1}(x)w_l(x)dx.$$

We denote the output vectors as $\rho_{k,l}$, $1 \leq k \leq 12$, $1 \leq l \leq 4$. We have then solved the linear systems $\widehat{H_{11,l}}x_l = e_{12}$ in a stable way by means of the Mathematica$^{\mathrm{TM}}$ function LinearSolve which uses Gaussian elimination with partial pivoting and back substitution. The relative errors ϵ_l are defined as $\epsilon_l = ||x_l - \rho_{12,l}||_\infty / ||x_l||_\infty$. Table 1 below reports the results of our computation.

Table 1

l	1	2	3	4
ϵ_l	$1.8\ 10^{-15}$	$3.9\ 10^{-19}$	$2.0\ 10^{-16}$	$5.9\ 10^{-19}$

These results strongly depend on the procedure which we use for constructing the modified matrix $\widehat{H_{11,l}}$. In our experiments the computation of the entries of $\widehat{H_{11,l}}$ has been carried out by means of numerical quadrature rules evaluated with multiprecision arithmetic. It can be easily seen that the conditioning of the matrix L_k grows exponentially with k, so that the transformation of the moments to the corresponding modified moments is to be avoided (a discussion of these issues can be found in [1]). In the practice of computation, one should have the expansion of $w_l(x)$ in terms of the orthonormal Legendre polynomials so that the entries $h_{i,j}^{(l)}$ can be computed by using the well known expansion of $p_i(x)p_j(x)$ in terms of the same $p_i(x)$ [14]. If we assume that the coefficients of such expansions are already available, then we we may determine the needed $O(n)$ entries of the matrix $\widehat{H_{n-1}}$ at the overall cost of $O(n^2)$ arithmetic operations.

REFERENCES

1 U. Amato, M. F. Carfora, *The modified moment problem in the presence of noise*, Approximation Theory and its Appl., **9** (1993), pp. 50-70.

2 D. Bini, L.Gemignani, *On the Euclidean scheme for polynomials having interlaced real zeros*, Proc. 2nd Ann. ACM Symp. on Parallel Algorithms and Architectures, pp. 254-258, ACM Press, 1990.

3 D. Bini, L. Gemignani, *Fast parallel computation of polynomial remainder sequence via Bezout and Hankel matrices* , SIAM J. Comput., **24** (1995), pp. 63-77.

4 A. Bojanczyk, G. Heinig, *A multi-step algorithm for Hankel matrices*, J. of Complexity, **10** (1994), pp. 142-164.

5 S. Cabay, R. Meleshko, *A weakly stable algorithm for Padé approximants and the inversion of Hankel matrices*, SIAM J. Matrix Anal. Appl., **14** (1993), pp.735-765.

6 T. F. Chan, P. C. Hansen, *A look-ahead Levinson algorithm for indefinite Toeplitz systems*, SIAM J. Matrix Anal. Appl., **13** (1992), pp. 490-506.

7 D. Fasino, *Preconditioning finite moment problems*, Pubbl. I.A.G.A., CNR, Firenze 1994. To appear in J. Comput. Appl. Math..

8 R. W. Freund, H. Zha, *A look-ahead algorithm for the solution of general Hankel systems*, Numer. Math., **64** (1993), pp. 295-321.

9 L. Gemignani, *Rational interpolation at Chebyshev points*, tech. report. In preparation.

10 L. Gemignani, *Manipulating polynomials in generalized form*, tech. report. In preparation.

11 I. Gohberg, T. Kailath, I. Koltracht, *Efficient solution of Linear Systems of Equations with Recursive Structure*, Linear Algebra Appl., **80** (1986), pp. 81-113.

12 I. Gohberg, I. Koltracht, *Efficient Algorithm for Toeplitz plus Hankel Matrices*, Integral Equations and Operator Theory, **12** (1989), pp. 136-142.

13 I. Gohberg, V. Olshevsky, *Fast Inversion of Chebyshev-Vandermonde Matrices*, Numer. Math., **67** (1994), pp. 71-92.

14 I. S. Gradshteyn, I. M. Ryzhik, *Table of integrals, series and products*, Academic Press, New York, 1965.

15 S. A. Gustafson, *On Computational Applications of the Theory of Moment Problems*, Rocky Mountain J. of Math., **2** (1974), pp. 227-240.

16 M. H. Gutknecht, *Stable row recurrences in the Padé table and generically superfast look-ahead solvers for non-hermitian Toeplitz systems*, Linear Algebra Appl., **188/189** (1993), pp.351-421.

17 G. Heinig, K. Rost, *Algebraic Methods for Toeplitz-like Matrices and Operators*, Birkhäuser, Boston (MA), 1984.

18 T. Kailath, A. H. Sayed, *Displacement Structure: Theory and Applications*, SIAM Review, **37** (1995), pp. 297-386.

19 V. Pták , *Lyapunov, Bézout and Hankel*, Linear Algebra Appl., **58** (1984), pp. 363-390.

20 G. Rodriguez, S. Seatzu, *Numerical solution of the finite moment problemin a reproducing kernel Hilbert space*, J. Comput. Appl. Math., **33** (1990), pp. 233-244.

21 K. Rost, *Generalized Companion Matrices and Matrix Representations for Generalized Bezoutians*, Linear Algebra Appl., **193** (1993), pp.151-172.

22 G. Talenti, *Recovering a function from a finite number of moments*, Inv. Prob., **3** (1987), pp. 501-517.

23 E. E. Tyrtyshnikov, *How bad are Hankel matrices?*, Numer. Math., **67** (1994), pp. 261-269.