JACOBI MATRICES FOR MEASURES MODIFIED BY A RATIONAL FACTOR

SYLVAN ELHAY* AND JAROSLAV KAUTSKY†

Abstract.
This paper decribes how, given the Jacobi matrix J for the measure $d\sigma(t)$, it is possible to produce the Jacobi matrix $\hat{J}$ for the measure $r(t)d\sigma(t)$ where $r(t)$ is a quotient of polynomials. The method uses a new factoring algorithm to generate the Jacobi matrices associated with the partial fraction decomposition of $r(t)$ and then applies a previouly developed summing technique to merge these Jacobi matrices. The factoring method performs best just where Gautschi's minimal solution method for this problem is weakest and vice versa. This suggests a hybrid strategy which is believed to be the most powerful yet for solving this problem. The method is demonstrated on a simple example and some numerical tests illustrate its performance characteristics.

Key words. orthogonal polynomials, Jacobi matrices, rational factors modifying measures, Gauss quadratures

1. Introduction. Let $d\sigma(t) \geq 0$, a measure with finite, infinite or semi-infinite support $[a,b]$, be such that the power moments

$$\mu_j = \int_a^b t^j d\sigma(t) \tag{1}$$

exist and are finite for all $j = 0, 1, 2, \cdots$. Let $\boldsymbol{J}$ be the (symmetric, tri-diagonal) Jacobi matrix, the elements of which are the coefficients in the three-term recurrence relation for the polynomials $p_j(t)$ $j = 0, 1, 2, \cdots$ which are orthonormal with respect to $d\sigma(t)$ on $[a,b]$. Jacobi matrices and their associated orthogonal polynomials arise in connection with approximate numerical integration, least squares approximations for continuous and discrete measures, series expansions and continued fractions.

The Jacobi matrix for some $d\sigma(t)$ can, in principle, be computed from the power moments (1), but the build-up of roundoff errors in this process is so severe that in practice this method is almost useless. However, for certain of the classical measures the Jacobi matrices are explicitly known (see [2],[10] for some examples) and much effort has gone towards the investigation and development of methods which can be viewed as transformations from one or more known Jacobi matrices, $\boldsymbol{J}_1, \boldsymbol{J}_2, \cdots$, corresponding to measures $d\sigma_1(t), d\sigma_2(t), \cdots$, into the required Jacobi matrix $\hat{\boldsymbol{J}}$ for the measure $d\hat{\sigma}(t)$ which is related to the $d\sigma_i(t)$. This is, of course, essentially the same as transforming one set or sets of orthogonal polynomials into a set of others, but is done in a numerical linear algebra context.

Thus, transformation methods based on the modified moments

$$\hat{\mu}_j = \int_a^b p_j(t) d\hat{\sigma}(t), \quad j \geq 0, \tag{2}$$

(in which the p_j are orthogonal with respect to the measure $d\sigma(t)$), were proposed by Sack and Donovan in [9]. The Lower Triangular Lanczos (LTL) method of [8] (which is equivalent to the Modified Chebyshev method of [5]), is based on modified moments and is of particular interest in this paper.

* Computer Science Department, University of Adelaide, South Australia, 5005.
† School of Information Science and Technology, Flinders University, Bedford Park, South Australia, 5042.

In [8], Kautsky and Golub show, for $r(t) \geq 0 \quad \forall t \in [a,b]$ a polynomial, how to find $\hat{\boldsymbol{J}}$ for $d\hat{\sigma}(t) = r(t)d\sigma(t)$ from $\boldsymbol{J}$ for $d\sigma(t)$ (Polynomial Shift Implicit QR or PSIQR method). The present authors used this approach [7] to extend this work to the case where the polynomial r can change sign in $[a,b]$ and use the methods so developed [4] to compute sequences of Kronrod-Patterson type imbedded quadrature sequences for two classical measures, e^{-t} on $[0,\infty]$ and e^{-t^2} on $[-\infty,\infty]$, for which Jacobi matrices are explicitly known.

More recently Elhay, Golub and Kautsky show that it is possible (see [3]), given the Jacobi matrices $\boldsymbol{J}_1$ and $\boldsymbol{J}_2$ for the measures $d\sigma_1(t)$ and $d\sigma_2(t)$, to produce the Jacobi matrix $\boldsymbol{J}_3$ corresponding to the measure

$$d\sigma_3(t) = \rho_1 d\sigma_1(t) + \rho_2 d\sigma_2(t), \tag{3}$$

$\rho_{1,2}$ constants, if such a matrix $\boldsymbol{J}_3$ exists (SUM method).

Gautschi, in [5] and [6], shows how to compute the modified moments of the measures

$$d\sigma(t)/(t-v), \quad v \notin [a,b], \tag{4}$$

and

$$d\sigma(t)/((t-x)^2+y^2), \quad x \in \boldsymbol{R}, \quad y > 0, \tag{5}$$

by means of a so-called minimal solution to the three-term recurrence for the polynomials orthogonal with respect to $d\sigma(t)$ (MS method). Once these modified moments are available it is possible to compute the corresponding Jacobi matrix by means of the LTL method.

Note that the denominator in (5) is just $(t-v)(t-\overline{v})$ with complex $v = x+iy$. Thus modifications such as the ones in (4) and (5) allow for computation of $\boldsymbol{J}$ corresponding to $d\sigma(t)$ divided by either a real linear factor or a pair of complex conjugate factors.

Earlier work by Uvarov [11] uses the second solution and Christoffel-Darboux relations to establish determinantal expressions which relate the polynomials orthogonal with respect to some $d\sigma(t)$ to those orthogonal with respect to $r(t)d\sigma(t)$, $r(t)$ a rational fraction with known factors. However, there seems to be no easy way to use them to determine the required Jacobi matrix.

The PSIQR and SUM methods are based on orthogonal rotations and are numerically very stable. While the MS can produce very accurate Jacobi matrices for measures of the type (4) and (5) it suffers from the problem that, for many important cases, infeasibly large initial matrices may be required in order to produce a quite moderate dimension result. In these cases the problem is compounded if we need to apply the MS method to (already divided) measures such as (4) and (5) repeatedly, for example, to produce a Jacobi matrix that corresponds to $d\sigma(t)$ divided by a product of factors.

Thus, there exist very satisfactory methods for Jacobi matrices that correspond to sums and differences of measures, or products of measures with polynomials, but there remains a need for satisfactory methods to compute the Jacobi matrix for a measure modified by a quotient of polynomials.

More precisely, we are interested in the following problem:

PROBLEM 1.1. *Let the rational function $r(t) = g_n(t)/h_m(t)$, g_n and h_m polynomials of degree n and m respectively, be such that h_m has no zeros in $[a,b]$ and $r(t) \geq 0$, $\forall t \in [a,b]$. Given a finite $\boldsymbol{J}$ for $d\sigma(t)$ find $\hat{\boldsymbol{J}}$ for $d\hat{\sigma}(t) = r(t)d\sigma(t)$.*

Clearly, since $d\hat{\sigma}(t)$ is non-negative on the interval $[a,b]$, there exist polynomials $\hat{p}_j(t)$, $j = 0, 1, 2, \cdots$, each of exact degree j, which are orthogonal on $[a,b]$ with respect to the inner product

$$(f,g) = \int_a^b f(t)g(t)r(t)d\sigma(t).$$

The importance of a satisfactory solution to Problem 1.1 is illustrated by the following application. Suppose we know $\boldsymbol{J}$ for $d\sigma(t)$ and we require the Jacobi matrix $\hat{\boldsymbol{J}}$ for $d\hat{\sigma}(t) = q(t)d\sigma(t)$, q some smooth function. If the interval of orthogonality is finite we can approximate q by a polynomial g_n and use the PSIQR method to transform $\boldsymbol{J}$ into $\hat{\boldsymbol{J}}$ for $d\hat{\sigma}(t) = g_n d\sigma(t)$. But on an infinite interval no polynomial can usefully approximate a smooth function which does not grow beyond all bounds at infinity. Similarly, a function with a singularity close to the support of $d\sigma(t)$ may not be well approximated by any polynomial. This is just where rational approximations can be used with good effect. Thus Jacobi matrices have important application in the computation of Gaussian quadrature formulae for measures of the type $d\tilde{\sigma}(t) = r(t)d\sigma(t)$ where r is a rational approximation to q.

In the next section we derive new algorithms which transform the known Jacobi matrix $\boldsymbol{J}$ for the measure $d\sigma(t)$ into the Jacobi matrix $\hat{\boldsymbol{J}}$ corresponding to $d\sigma(t)$ divided by a linear or complex conjugate pair quadratic factor. We call these, collectively, the Inverse Cholesky (IC) method. In §3 we indicate the IC algorithms themselves and briefly review the MS method. The discussion in §4 leads to a hybrid strategy for handling rational modification which uses both the IC and MS methods. In §5 we show a simple example and in §6 we summarize the results of some numerical tests. In §7 we make some concluding remarks. Tables of numerical results and certain moments, which may be needed when using the IC method with the classical measures, are displayed in §8.

2. The new methods. In §1 we mentioned that the Jacobi matrix for a measure modified by multiplication with a polynomial is stably and efficiently done with the PSIQR method. So our concern in the solution to Problem 1.1 is in the determination of a Jacobi matrix that corresponds to a measure $d\sigma(t)$ modified by division with a polynomial $h_m(t)$. We assume that the m (real or complex conjugate pair) zeros of h_m are known and simple.

We now establish some notation and derive the relations on which the IC method is based.

Suppose that the polynomials $\boldsymbol{p}(t) = (p_0(t), p_1(t), \cdots, p_{n-1}(t))^T$ and $p_n(t)$ are orthonormal on the interval $[a,b]$ with respect to the non-negative measure $d\sigma(t)$. Then there exists a (symmetric, tridiagonal) matrix

$$\boldsymbol{J} = \begin{pmatrix} \alpha_1 & \beta_1 & 0 & \cdots & 0 \\ \beta_1 & \alpha_2 & \beta_2 & \cdots & 0 \\ 0 & \beta_2 & \alpha_3 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \alpha_n \end{pmatrix},$$

the elements of which are the coefficients in the three-term recurrence which the polynomials satisfy. In the following identity, as elsewhere in this paper,

$$t\boldsymbol{p}(t) = \boldsymbol{J}\boldsymbol{p}(t) + \beta_n p_n(t)\boldsymbol{e}_n \tag{6}$$

the vector $\boldsymbol{e}_n$ is the last column of an identity matrix of appropriate dimension. Suppose now that $\hat{\boldsymbol{p}}(t) = (\hat{p}_0(t), \hat{p}_1(t), \cdots, \hat{p}_{n-1}(t))^T$ is a second set of polynomials orthonormal with respect to some measure $d\hat{\sigma}(t)$. The orthogonality of the polynomials implies (multiply (6) on the right by $\boldsymbol{p}(t)^T$ and integrate)

$$\int_a^b \boldsymbol{p}(t)\boldsymbol{p}(t)^T d\sigma(t) = \boldsymbol{I}, \quad \int_a^b t\boldsymbol{p}(t)\boldsymbol{p}(t)^T d\sigma(t) = \boldsymbol{J} \tag{7}$$

and there are corresponding relations for the second set of orthonormal polynomials.

Furthermore, there exist a nonsingular lower triangular matrix $\boldsymbol{L}$, a vector $\boldsymbol{c}$ and a non-zero constant γ_n, all such that

$$\boldsymbol{p}(t) = \boldsymbol{L}\hat{\boldsymbol{p}}(t), \quad p_n(t) = \boldsymbol{c}^T\hat{\boldsymbol{p}}(t) + \gamma_n\hat{p}_n(t). \tag{8}$$

Thus, if $\hat{\boldsymbol{J}}$ is the Jacobi matrix in the identity corresponding to (6) for the polynomials $\hat{\boldsymbol{p}}(t)$, $\hat{p}_n(t)$ it is easy to show [4] that

$$\hat{\boldsymbol{J}} = \boldsymbol{L}^{-1}\boldsymbol{J}\boldsymbol{L} + \boldsymbol{e}_n\boldsymbol{b}^T, \tag{9}$$

where $\boldsymbol{L}^{-1}\boldsymbol{e}_n = \theta_n\boldsymbol{e}_n$ for some non-zero constant θ_n and $\boldsymbol{b} = \beta_n\theta_n\boldsymbol{c}$. If we need the matrix $\hat{\boldsymbol{J}}$ of dimension n we can get it by discarding the last row and column of the $n+1$ dimension product $\boldsymbol{L}^{-1}\boldsymbol{J}\boldsymbol{L}$, thus obviating the need to compute $\boldsymbol{b}$.

The methods we present in this paper determine the matrix $\boldsymbol{L}$ directly from $\boldsymbol{J}$ and some additional information. Our starting point is the theorem:

THEOREM 2.1. *Let $r(t)$ be any analytic function of t. Then*

$$(\boldsymbol{J} - t\boldsymbol{I})\boldsymbol{p}(t)r(t) = (\boldsymbol{J} - t\boldsymbol{I})r(\boldsymbol{J})\boldsymbol{p}(t) + \beta_n p_n(t)(r(\boldsymbol{J}) - r(t)\boldsymbol{I})\boldsymbol{e}_n.$$

If t is not an eigenvalue of $\boldsymbol{J}$ then

$$\boldsymbol{p}(t)r(t) = r(\boldsymbol{J})\boldsymbol{p}(t) + \beta_n p_n(t)(\boldsymbol{J} - t\boldsymbol{I})^{-1}(r(\boldsymbol{J}) - r(t)\boldsymbol{I})\boldsymbol{e}_n. \tag{10}$$

If v is such that $p_n(v) = 0$ then $r(v)$ is an eigenvalue of $r(\boldsymbol{J})$ and $\boldsymbol{p}(v)$ is the corresponding eigenvector.

Proof. From (6) we can write

$$(\boldsymbol{J} - t\boldsymbol{I})\boldsymbol{p}(t) = -\beta_n p_n(t)\boldsymbol{e}_n.$$

If $r(t)$ is an analytic function of t then $\boldsymbol{J}$ and $r(\boldsymbol{J})$ commute so we have

$$(\boldsymbol{J} - t\boldsymbol{I})(r(\boldsymbol{J}) - r(t)\boldsymbol{I})\boldsymbol{p}(t) = -\beta_n p_n(t)(r(\boldsymbol{J}) - r(t)\boldsymbol{I})\boldsymbol{e}_n$$

from which (10) immediately follows. □

Multiplying (10) on the right by $\frac{1}{r(t)}\boldsymbol{p}(t)$ and integrating gives

$$\begin{aligned}\int_a^b \boldsymbol{p}(t)\boldsymbol{p}(t)^T d\sigma(t) \quad &= \quad r(\boldsymbol{J})\int_a^b \boldsymbol{p}(t)\boldsymbol{p}(t)^T \frac{d\sigma(t)}{r(t)} + \\ &\qquad \beta_n \int_a^b p_n(t)(\boldsymbol{J} - t\boldsymbol{I})^{-1}(r(\boldsymbol{J}) - r(t)\boldsymbol{I})\boldsymbol{e}_n\boldsymbol{p}(t)^T \frac{d\sigma(t)}{r(t)}.\end{aligned}$$

Now if the polynomials $\boldsymbol{p}(t)$ are orthonormal with respect to the measure $d\sigma(t)$ and the polynomials $\hat{\boldsymbol{p}}(t)$ are orthonormal with respect to the divided measure $d\hat{\sigma}(t) = \frac{d\sigma(t)}{r(t)}$

then we have, after substituting with (8),

$$\begin{aligned} \boldsymbol{I} &= r(\boldsymbol{J})\boldsymbol{L}\boldsymbol{L}^T + \\ &\beta_n \int_a^b (\boldsymbol{c}^T\hat{\boldsymbol{p}}(t) + \gamma_n\hat{p}_n(t))(\boldsymbol{J} - t\boldsymbol{I})^{-1}(r(\boldsymbol{J}) - r(t)\boldsymbol{I})\boldsymbol{e}_n\hat{\boldsymbol{p}}(t)^T d\hat{\sigma}(t)\boldsymbol{L}^T. \end{aligned} \tag{11}$$

The two cases of present interest are $r(t) = t - v$ and $r(t) = (t-x)^2 + y^2$. For the case $r(t) = t - v$ relation (11) reduces to

$$\begin{aligned} \boldsymbol{I} &= (\boldsymbol{J} - v\boldsymbol{I})\boldsymbol{L}\boldsymbol{L}^T + \beta_n \int_a^b (\boldsymbol{c}^T\hat{\boldsymbol{p}}(t) + \gamma_n\hat{p}_n(t))\boldsymbol{e}_n\hat{\boldsymbol{p}}(t)^T d\hat{\sigma}(t)\boldsymbol{L}^T, \\ &= (\boldsymbol{J} - v\boldsymbol{I})\boldsymbol{L}\boldsymbol{L}^T + \beta_n\boldsymbol{e}_n\boldsymbol{c}^T\boldsymbol{L}^T \end{aligned} \tag{12}$$

by orthogonality. Note that the matrix $\boldsymbol{L}^{-T}$ is a Cholesky-like factor of $(\boldsymbol{J} - v\boldsymbol{I})$ plus a rank-one correction to its last row.

For the second case of interest, $r(t) = (t-x)^2 + y^2$, we have

$$\begin{aligned} \boldsymbol{I} &= ((\boldsymbol{J} - x\boldsymbol{I})^2 + y^2\boldsymbol{I})\boldsymbol{L}\boldsymbol{L}^T + \\ &\beta_n \int_a^b (\boldsymbol{c}^T\hat{\boldsymbol{p}}(t) + \gamma_n\hat{p}_n(t))((\boldsymbol{J} - x\boldsymbol{I}) + (t-x)\boldsymbol{I})\boldsymbol{e}_n\hat{\boldsymbol{p}}(t)d\hat{\sigma}(t)\boldsymbol{L}^T. \end{aligned} \tag{13}$$

Now the second term on the right of (13) reduces, by orthogonality, to

$$\begin{aligned} &\beta_n \left\{ (\boldsymbol{J} - 2x\boldsymbol{I})\boldsymbol{e}_n\boldsymbol{c}^T + \boldsymbol{e}_n\boldsymbol{c}^T \int_a^b t\hat{\boldsymbol{p}}(t)\hat{\boldsymbol{p}}(t)^T d\hat{\sigma}(t) \right\} \boldsymbol{L}^T + \hat{\gamma}_n\boldsymbol{e}_n\boldsymbol{e}_n{}^T \\ &= \beta_n \left\{ (\boldsymbol{J} - 2x\boldsymbol{I})\boldsymbol{e}_n\boldsymbol{c}^T + \boldsymbol{e}_n\boldsymbol{c}^T\hat{\boldsymbol{J}} \right\} \boldsymbol{L}^T + \hat{\gamma}_n\boldsymbol{e}_n\boldsymbol{e}_n{}^T \end{aligned} \tag{14}$$

for some constant $\hat{\gamma}_n$. Bearing in mind that the matrix $\hat{\boldsymbol{J}}$ in (14) is tridiagonal, it is easy to see that (14) is a matrix, the first $n-2$ rows of which vanish, so the $\boldsymbol{L}^{-T}$ matrix for this case is a Cholesky-like factor of $(\boldsymbol{J} - x\boldsymbol{I})^2 + y^2\boldsymbol{I}$ plus a rank-two correction to its last two rows.

The methods we describe in this paper directly generate the matrix $\boldsymbol{L}$, in each of the above cases, from the top down. If the process is terminated before reaching the row or rows which are affected by the low rank corrections, then the part of $\boldsymbol{L}$ produced to that point will be correct and the low rank corrections need not be computed.

3. The algorithms.

In this section we derive the algorithms for the Cholesky-like factors of the inverse of symetric tridiagonal and pentadiagonal matrices. In this section we write $\boldsymbol{L}_{i:j,k:m}$ to denote rows i through j of columns k through m of the matrix $\boldsymbol{L}$.

3.1. The linear-factor Inverse Cholesky algorithm.

Given a symmetric tridiagonal matrix $\boldsymbol{J}$, we require the factor $\boldsymbol{L}$ which satisfies

$$\boldsymbol{I} = \boldsymbol{J}\boldsymbol{L}\boldsymbol{L}^T + \boldsymbol{e}_n\boldsymbol{d}^T, \tag{15}$$

where the lower triangle $\boldsymbol{L}$ has elements

$$\boldsymbol{L} = \begin{pmatrix} \ell_{11} & 0 & 0 & \dots & 0 \\ \ell_{21} & \ell_{22} & 0 & \dots & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \ell_{n1} & \ell_{n2} & \ell_{n3} & \dots & \ell_{nn} \end{pmatrix}.$$

Noting that

$$\boldsymbol{e}_i^T\boldsymbol{J} = (\,0 \quad \ldots \quad 0 \quad \beta_{i-1} \quad \alpha_i \quad \beta_i \quad 0 \quad \ldots \quad 0\,) \qquad 1 < i < n$$

we have, considering only $1 \le j \le i \le n$,

$$\begin{aligned}(\delta_{ij} - \delta_{ni}d_j) &= \beta_{i-1}\sum_{k=1}^{n}\ell_{i-1,k}\ell_{jk} + \alpha_i\sum_{k=1}^{n}\ell_{ik}\ell_{jk} + \beta_i\sum_{k=1}^{n}\ell_{i+1,k}\ell_{jk} \\ &= \beta_{i-1}\sum_{k=1}^{\min i-1,j}\ell_{i-1,k}\ell_{jk} + \alpha_i\sum_{k=1}^{j}\ell_{ik}\ell_{jk} \\ &\quad +\beta_i\sum_{k=1}^{j-1}\ell_{i+1,k}\ell_{jk} + \beta_i\ell_{i+1,j}\ell_{jj}\,.\end{aligned}$$

Algorithm 3.1, which is based on this relation, determines the $\boldsymbol{L}$ and $\boldsymbol{d}$ of (15). Its input is a symmetric tridiagonal matrix $\boldsymbol{J}$ and $\ell_{11} \neq 0$.

ALGORITHM 3.1.
$\beta_0 = 0$
$\ell_{21} = (1/\ell_{11} - \alpha_1\ell_{11})/\beta_1$
for $i = 2 : n-1$
  $\ell_{i+1,1} = -(\beta_{i-1}\ell_{i-1,1} + \alpha_i\ell_{i1})/\beta_i$
end
$d_1 = -(\beta_{n-1}\ell_{n-1,1} + \alpha_n\ell_{n1})$
for $j = 2 : n$
  $s = \beta_{j-2}\boldsymbol{L}_{j-2,1:j-2}\boldsymbol{L}^T_{j,1:j-2} + \alpha_{j-1}\boldsymbol{L}_{j-1,1:j-1}\boldsymbol{L}^T_{j,1:j-1} + \beta_{j-1}\boldsymbol{L}_{j,1:j-1}\boldsymbol{L}^T_{j,1:j-1}$
  $\ell_{jj} = \sqrt{-s/\beta_{j-1}}$
  for $i = j : n-1$
    $s = \beta_{i-1}\boldsymbol{L}_{i-1,\min\{i-1,j\}}\boldsymbol{L}^T_{j,\min\{i-1,j\}} + \alpha_i\boldsymbol{L}_{i,1:j}\boldsymbol{L}^T_{j,1:j} + \beta_i\boldsymbol{L}_{i+1,1:j-1}\boldsymbol{L}^T_{j,1:j-1}$
    $\ell_{i+1,j} = (\delta_{ij} - s)/\beta_i/\ell_{jj}$
  end
  $d_j = -(\beta_{n-1}\boldsymbol{L}_{n-1,1:\min\{n-1,j\}}\boldsymbol{L}^T_{j,1:\min\{n-1,j\}} + \alpha_n\boldsymbol{L}_{n,1:j}\boldsymbol{L}^T_{j,1:j})$
end

3.2. The quadratic-factor Inverse Cholesky algorithm.
Here we require the lower triangle $\boldsymbol{L}$ which satisfies the relation

$$\boldsymbol{I} = \boldsymbol{J}^2\boldsymbol{L}\boldsymbol{L}^T + \boldsymbol{e}_n\boldsymbol{d}^T + \boldsymbol{e}_{n-1}\boldsymbol{f}^T. \tag{16}$$

where, $\boldsymbol{J}$ being tridiagonal, $\boldsymbol{J}^2$ is pentadiagonal and denoted (for different α's and β's) by

$$\boldsymbol{J}^2 = \begin{pmatrix} \alpha_1 & \beta_1 & \gamma_1 & 0 & 0 & \cdots & 0 \\ \beta_1 & \alpha_2 & \beta_2 & \gamma_2 & 0 & \cdots & 0 \\ \gamma_1 & \beta_2 & \alpha_3 & \beta_3 & \gamma_3 & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & \alpha_n \end{pmatrix}.$$

We assume that the principal 2×2 block of $\boldsymbol{L}$ is prescribed. We have

$$\boldsymbol{e}_i^T\boldsymbol{J} = (\,0 \quad \ldots \quad 0 \quad \gamma_{i-2} \quad \beta_{i-1} \quad \alpha_i \quad \beta_i \quad \gamma_i \quad 0 \quad \ldots \quad 0\,), \quad 2 < i < n-2,$$

and denoting $\boldsymbol{L}$ as in the previous section, we have, for $1 \le j \le i \le n$,

$$\begin{aligned}(\delta_{ij} - \delta_{n,i-1}f_j - \delta_{ni}d_j) \;=\; & \gamma_{i-2}\sum_{k=1}^{\min\{i-2,j\}} \ell_{i-2,k}\ell_{jk} + \beta_{i-1}\sum_{k=1}^{\min\{i-1,j\}} \ell_{i-1,k}\ell_{jk} \\ & +\alpha_i\sum_{k=1}^{j}\ell_{ik}\ell_{jk} + \beta_i\sum_{k=1}^{j}\ell_{i+1,k}\ell_{jk} \\ & +\gamma_i\sum_{k=1}^{j-1}\ell_{i+2,k}\ell_{jk} + \gamma_i\ell_{i+2,j}\ell_{jj}\,.\end{aligned}$$

Algorithm 3.2, which is based on this relation, determines the $\boldsymbol{L}$ of (16). Its input is the symmetric pentadiagonal matrix $\boldsymbol{J}^2$ and the 2×2 principal submatrix of $\boldsymbol{L}$. The algorithm then finds the rest of $\boldsymbol{L}$. The steps which compute $\boldsymbol{d}$ and $\boldsymbol{f}$ are omitted for brevity.

ALGORITHM 3.2.

$\ell_{3,1} = (1/\ell_{1,1} - \alpha_1\ell_{1,1} - \beta_1\ell_{2,1})/\gamma_1$

$\ell_{4,1} = -(\beta_1\ell_{1,1} + \alpha_2\ell_{2,1} + \beta_2\ell_{3,1})/\gamma_2$

for $i = 4 : n-1$

$\quad \ell_{i+1,1} = -(\gamma_{i-3}\ell_{i-3,1} + \beta_{i-2}\ell_{i-2,1} + \alpha_{i-1}\ell_{i-1,1} + \beta_{i-1}\ell_{i,1})/\gamma_{i-1}$

end

$\ell_{3,2} = -(\alpha_1\ell_{2,1}\ell_{1,1} + \beta_1\ell_{2,1}^2 + \gamma_1\ell_{2,1}\ell_{3,1} + \beta_1\ell_{2,2}^2)/(\gamma_1\ell_{2,2})$

$s = \beta_1\ell_{2,1}\ell_{1,1} + \alpha_2\ell_{2,1}^2 + \beta_2\ell_{2,1}\ell_{3,1} + \gamma_2\ell_{2,1}\ell_{4,1} + \alpha_2\ell_{2,2}^2 + \beta_2\ell_{2,2}\ell_{3,2}$

$\ell_{4,2} = (1-s)/(\gamma_2\ell_{2,2})$

for $i = 3 : n-2$

$\quad s = \gamma_{i-2}\boldsymbol{L}_{i-2,1:\min\{i-2,2\}}\boldsymbol{L}_{2,1:\min\{i-2,2\}}^T + \beta_{i-1}\boldsymbol{L}_{i-1,1:\min\{i-1,2\}}\boldsymbol{L}_{2,1:\min\{i-1,2\}}^T$

$\qquad +\alpha_i\boldsymbol{L}_{i,1:2}\boldsymbol{L}_{2,1:2}^T + \beta_i\boldsymbol{L}_{i+1,1:2}\boldsymbol{L}_{2,1:2}^T + \gamma_i\ell_{i+2,1}\ell_{2,1}^T$

$\quad \ell_{i+2,2} = -s/(\gamma_i\ell_{2,2})$

end

for $j = 3 : n$

$\quad s = 0$

$\quad$ if $j > 3, s = s + \beta_{j-3}\boldsymbol{L}_{j-3,1:j-3}\boldsymbol{L}_{j,1:j-3}^T$, end

$\quad$ if $j > 4, s = s + \gamma_{j-4}\boldsymbol{L}_{j-4,1:j-4}\boldsymbol{L}_{j,1:j-4}^T$, end

$\quad s = s + \alpha_{j-2}\boldsymbol{L}_{j-2,1:j}\boldsymbol{L}_{j,1:j}^T + \beta_{j-2}\boldsymbol{L}_{j-1,1:j}\boldsymbol{L}_{j,1:j}^T + \gamma_{j-2}\boldsymbol{L}_{j,1:j-1}\boldsymbol{L}_{j,1:j-1}^T$

$\quad \ell_{j,j} = \sqrt{-s/\gamma_{j-2}}$

$\quad$ for $i = j-1 : n-2$

$\qquad s = \gamma_{i-2}\boldsymbol{L}_{i-2,1:\min\{i-2,j\}}\boldsymbol{L}_{j,1:\min\{i-2,j\}}^T + \beta_{i-1}\boldsymbol{L}_{i-1,1:\min\{i-1,j\}}\boldsymbol{L}_{j,1:\min\{i-1,j\}}^T +$

$\qquad\quad \alpha_i\boldsymbol{L}_{i,1:j}\boldsymbol{L}_{j,1:j}^T + \beta_i\boldsymbol{L}_{i+1,1:j}\boldsymbol{L}_{j,1:j}^T + \gamma_i\boldsymbol{L}_{i+2,1:j-1}\boldsymbol{L}_{j,1:j-1}^T$

$\qquad \ell_{i+2,j} = (\delta_{ij} - s)/(\gamma_i\ell_{j,j})$

$\quad$ end

end

3.2.1. Starting values. The linear factor IC method will be used for real v. To start it we need the value of

$$\ell_{11} = (\tilde{\mu}_0/\mu_0\,,)^{\frac{1}{2}}$$

(see (8)) where

$$\tilde{\mu}_0 = \int_a^b \frac{d\sigma(t)}{t-v},$$

is available, from the tables provided, for the classical measures.

The quadratic factor IC method will be used for complex conjugate pairs v, $\overline{v}$ where $v = x + iy$, x, y real, $y > 0$. To start it we need the principal 2×2 submatrix of $\boldsymbol{L}$,

$$\begin{pmatrix} \ell_{11} & 0 \\ \ell_{21} & \ell_{22} \end{pmatrix}.$$

The measure $d\sigma(t)$ has here been modified by the factor $(t-v)(t-\overline{v}) = (t-x)^2 + y^2$. Now denote

$$\tilde{\mu}_j = \int_a^b \frac{t^j}{(t-x)^2 + y^2} d\sigma(t), \quad j = 0, 1, 2.$$

Using (8) and orthognality it quickly follows that

$$\begin{aligned} \ell_{11}^2 &= \tilde{\mu}_0/\mu_0 \\ \ell_{21} &= \ell_{11}(\tilde{\mu}_1/\tilde{\mu}_0 - \alpha_1)/\beta_1 \\ \ell_{22}^2 &= (\tilde{\mu}_2 - \tilde{\mu}_1^2/\tilde{\mu}_0)/(\mu_0 \beta_1^2). \end{aligned}$$

For the classical measures, the $\tilde{\mu}_j$ can be found, again in terms of the tabled quantities, by

$$\begin{aligned} \tilde{\mu}_0 &= \frac{1}{2iy}\left(\int_a^b \frac{1}{t-v} d\sigma(t) - \int_a^b \frac{1}{t-\overline{v}} d\sigma(t)\right) \\ \tilde{\mu}_1 &= \frac{1}{2}\left(\int_a^b \frac{1}{t-v} d\sigma(t) + \int_a^b \frac{1}{t-\overline{v}} d\sigma(t)\right) - x\tilde{\mu}_0 \\ \tilde{\mu}_2 &= \mu_0 + 2x\tilde{\mu}_1 - (x^2 + y^2)\tilde{\mu}_0. \end{aligned}$$

3.3. Modified moments from the minimal solution. For completeness, we state here Gautschi's minimal solution (MS) method re-written to operate for normalized rather than monic polynomials.

Given $\boldsymbol{J}$ of dimension m, the zero-th moment, μ_0 of $d\sigma(t)$, a knot v outside the support interval of $d\sigma(t)$ and an integer $n = n(m, v) < m$, the algorithm produces the vector $\hat{\boldsymbol{\phi}} = (\phi_0, \phi_1, \phi_2, \ldots, \phi_{n-1})^T$. For real v, the modified moments of the measure (4) are the $\hat{\mu}_i = \phi_i$ and for complex v the modified moments of the measure (5) are given by

$$\hat{\mu}_i = \frac{Imag(\phi_i)}{Imag(v)}.$$

ALGORITHM 3.3.
$p_m = 0$
for $i = m : -1 : 2$
  $p_{i-1} = \beta_{i-1}/(v - \alpha_i - \beta_i p_i)$
end
$\phi_0 = \sqrt{\mu_0}/(\beta_1 p_1 - (v - \alpha_1))$
for $i = 1 : n - 1$
  $\phi_i = \phi_{i-1} p_i$
end

For the measures in Table 5 Gautschi proposes a lower bound on $n(m,v)$ which is large enough to ensure that the $\hat{\mu}_i$ are computed to sufficiently high accuracy. As the point v approaches support of $d\sigma(t)$ the value of $n(m,v)$ increases and larger dimension $\boldsymbol{J}$ is needed for the same accuracy.

4. Use of the IC and MS methods. Our discussion of just how to use the IC and MS methods divides naturally into two parts. The first part addresses the question of how to use the methods repeatedly to find the $\boldsymbol{J}$ for a measure divided by a general polynomial of degree $m > 2$. The second part addresses the question of how to decide on which of the IC and MS methods should be used for each of the linear and quadratic factors in the rational function denominator. This issue arises because, as we show later in the numerical tests, the IC and MS methods behave very differently from one another according to whether the division factors correspond to poles close to, or far from, the support of the measure $d\sigma(t)$.

4.1. Repeated divisions. In this section we compare

(a) the strategy of starting with a known $\boldsymbol{J}$, modifying it by division with a factor, modifying the result by division with the next factor, and so on, (the *product strategy*), with

(b) the strategy of modifying the known $\boldsymbol{J}$ for division by each of the factors separately and then using the summing method to produce the final result (the *summing strategy*).

To start, we compute the partial fraction decomposition of

$$h_m(t) = \prod_{k=1}^{m}(t - v_k).$$

In the case where the zeros of h_m are conjugate pairs, the partial fraction decomposition contains terms of the form

$$\frac{Ax + B}{(t-x)^2 + y^2}, \quad y > 0. \tag{17}$$

To compute the Jacobi matrix for such a term we use one of the algorithms below for the denominator first and then apply the PSIQR method to the result for the numerator.

4.1.1. The product strategy.

Denote by $\boldsymbol{J}_s$ the Jacobi matrix for the measure

$$\frac{d\sigma(t)}{\prod_{k=1}^{s}(t - v_k)}. \tag{18}$$

In the product strategy we start from the given $\boldsymbol{J}_0 = \boldsymbol{J}$ for $d\sigma(t)$ and sequentially calculate $\boldsymbol{J}_1, \boldsymbol{J}_2, \ldots$, finishing up with $\boldsymbol{J}_m = \hat{\boldsymbol{J}}$ for $d\sigma(t)/h_m(t)$. Each step in this sequence, in which we produce $\boldsymbol{J}_s$ *from* $\boldsymbol{J}_{s-1}$, can be performed, in principle, by either of the following algorithms (for complex pairs of zeros the IC method can be used to transform from $\boldsymbol{J}_{s-1}$ to $\boldsymbol{J}_{s+1}$ directly using only real arithmetic, as indicated in §3, but the MS method requires complex floating point arithmetic).

One stage using the MS method requires an algorithm such as:

ALGORITHM 4.1. *Input:* $\boldsymbol{J}_{s-1}$ *of sufficiently large dimension* j *and starting values as in §3.2.1.*

(a) Use the MS method to produce $2(i+2)$ *modified moments.*

(b) Use the LTL method to directly get $\boldsymbol{J}_s$ *of dimension* i.

By comparison, one stage using the IC method would require

ALGORITHM 4.2. *Input:* $\boldsymbol{J}_{s-1}$ *of dimension* $i+1$ *and starting values as in §3.2.1.*

(a) Use the IC method to produce $\boldsymbol{L}_s$ *of dimension* $i+1$.

(b) Compute $\boldsymbol{L}_s^{-1}\boldsymbol{J}_{s-1}\boldsymbol{L}_s = \boldsymbol{J}_s - \boldsymbol{e}_n\boldsymbol{b}_s^T$ *of dimension* $i+1$ *(see (9)).*

(c) Discard the last row and column of the product to get $\boldsymbol{J}_s$ *of dimension* i.

However, there are difficulties with Algorithm 4.1. The estimates of the size of j for given v_k when $d\sigma(t)$ is one of the classical (Jacobi, Laguerre or Hermite) measures can be quite large, if the point v_k is close to the support of the interval: for example to modify a dimension $i = 10$ Laguerre Jacobi matrix by the factor $(t-0.01)$ requires an initial Jacobi matrix of dimension nearly $j = 2000$ when using standard double precision IEEE arithmetic. Thus, even for quite modest final dimension $\hat{\boldsymbol{J}}$, a very large initial matrix may be needed.

More problematic is the fact that the only estimates for the size of the initial matrix at each stage are for the classical measures and there are not estimates for the divided classical measures. It is possible that one could check the accuracy of the process at each stage by recomputing with varied estimates of the starting size matrix but clearly this process is unsatisfactory. Indeed, it may even be necessary to abandon the calculation and start again from scratch at some stage if the remaining Jacobi matrix is too small to go to the next step.

Using the Algorithm 4.2 requires an initial matrix of only about $n+m$ but there are difficulties as before: the starting values, which are computed from the zero-th moments of measures such as (18) are not generally available.

4.1.2. The summing strategy. An alternative strategy uses the results of [3] mentioned in §1. If we denote the method for computing the Jacobi matrix which corresponds to a sum or difference of measures by SUM then the algorithm we propose is:

ALGORITHM 4.3. *Given* $\boldsymbol{J}$ *of dimension* $n+1$ *for* $d\sigma(t)$

(a) For each $k = 1:m$ *either*

(i) Use the MS method to produce $2(n+2)$ *modified moments for* $\tilde{\boldsymbol{J}}_{v_k}$ *corresponding to* $d\tilde{\sigma}(t) = d\sigma(t)/(t-v_k)$ *(or* $d\tilde{\sigma}(t) = d\sigma(t)/(t-v_k)(t-\overline{v}_k)$*).*

(ii) use the LTL method with these modified moments to get $\tilde{\boldsymbol{J}}_{v_k}$ *of dimension* n *for* $d\tilde{\sigma}(t)$

or

(i) Use the IC to produce $\boldsymbol{L}_k$ *for* $d\tilde{\sigma}(t) = d\sigma(t)/(t-v_k)$ *(or* $d\tilde{\sigma}(t) = d\sigma(t)/(t-v_k)(t-\overline{v}_k)$*).*

(ii) Compute $\boldsymbol{L}_k^{-1}\boldsymbol{J}\boldsymbol{L}_k = \tilde{\boldsymbol{J}}_{v_k} - \boldsymbol{e}_n\boldsymbol{b}_{v_k}^T$ *of dimension* $n+1$ *(or* $n+2$*) (see (9)).*

(iii) Discard the last row (two rows) and column (two columns) of the product to get $\tilde{\boldsymbol{J}}_{v_k}$ *of dimension* n *for* $d\tilde{\sigma}(t)$.

(b) Use method SUM to find $\hat{\boldsymbol{J}}$ *from the separate Jacobi matrices.*

This scheme overcomes the disadvantages, mentioned above, of the product strategy since the starting values are known (for both the IC and MS methods) for the classical measures and, in the case of the MS method, all the divisions use the same initial matrix and its dimension can be determined once at the outset from the estimates given in [5].

If the numerator of any term such as (17) changes sign in the support of $d\sigma(t)$, there may not exist a Jacobi matrix for it. However, by adding and subtracting a

constant to the numerator we can rewrite (17) as two separate terms each of which does *not* change sign on the support. We can then sum all the terms which are positive on the support and then subtract those which are negative. Since the final Jacobi matrix corresponds to a measure which is positive on the support (by assumption), the result must exist and be computable this way.

4.2. Distribution of the poles. Consider the case of modifying $\boldsymbol{J}$ for $d\sigma(t)$ to $\hat{\boldsymbol{J}}$ for $d\hat{\sigma}(t) = d\sigma(t)/(t-v)$, v real or complex, outside the support, $[a,b]$, of $d\sigma(t)$. The closer is v to $[a,b]$, the larger j, the dimension of $\boldsymbol{J}$, needs to be for the MS method to maintain accuracy. The estimates given in [5] for the size j when $d\sigma(t)$ is one of the Jacobi, Laguerre or Hermite measures, are reported there to be realistic in the sense that, using $\boldsymbol{J}$ of dimension 5 fewer than given by the estimate rarely gave sufficient accuracy and the given estimate was only occaisionally required to be increased by about 5 or 10. Thus close to the support, the MS method can be quite unsatisfactory.

By contrast, the IC method, as the numerical tests reported in §6 show, returns poorer results as the point v moves away from the support but is extremely accurate close to the support. Our numerical evidence also suggests that this effect is more pronounced for the measures with infinite or semi-inifinte support.

The obvious solution then, is to use Algorithm 4.3, applying the MS method on points far from $[a,b]$ and the IC methods on those close to $[a,b]$. Both methods can be applied to points which are neither close nor far and the two results used for confirmation. Thus, Algorithm 4.3 provides a stable and efficient method for a solution to Problem 1.1.

5. A simple example. The function $\ln(1+4t)$ has power series

$$\ln(1+4t) = 4\,t - 8\,t^2 + \frac{64\,t^3}{3} - 64\,t^4 + \frac{1024\,t^5}{5} - \frac{2048\,t^6}{3} + \frac{16384\,t^7}{7} + O(t^8),$$

and a Pade approximation

$$r(t) = g_3(t)/h_3(t) = \frac{60\,t + 240\,t^2 + 176\,t^3}{15 + 90\,t + 144\,t^2 + 48\,t^3} \tag{19}$$

which, on the subinterval $[0,20]$, has error bounded, in absolute value, by about 0.228. The denominator $h_3(t)$ has zeros

$$\{-1/2, -\frac{1}{4}(5 \mp \sqrt{15})\} \approx \{-0.5, -0.28175, -2.21825\}$$

and the numerator $g_3(t)$ has zeros $\approx \{0, -.3, -1\}$ so both $g_3(t)$ and $h_3(t)$ do not change sign on the support $[0,\infty]$ of the Laguerre measure.

The partial fraction decomposition of $1/h_3(t)$ is

$$\begin{aligned} & \frac{A_1}{t-v_1} + \frac{A_2}{t-v_2} + \frac{A_3}{t-v_3} \\ = \; & -\frac{1/18}{t+1/2} + \frac{1}{6}\frac{1/(15-3\sqrt{15})}{(t+(5-\sqrt{15})/4)} + \frac{1}{6}\frac{1/(15+3\sqrt{15})}{(t+(5+\sqrt{15})/4)}. \end{aligned}$$

We demonstrate the technique by finding the dimension 7 Jacobi matrix which corresponds to the measure $d\sigma(t) = e^{-t}r(t)$ on the interval $[0,\infty]$ from the (known, see [2]) Jacobi matrix for the measure $d\sigma(t) = e^{-t}$ on the same interval. Since

the processes we employ require us to discard one or two rows and columns of the computed matrices, we assume that the starting matrix is sufficiently large to give a result of the required dimensions. All the calculations were done in IEEE standard double precision arithmetic (machine epsilon $\approx 2.2 \times 10^{-16}$).

Let us denote the following power moments by

$$\tilde{\mu}_{j v_i} = \int_0^\infty \frac{t^j e^{-t}}{t - v_i} dt, \quad i = 1, 2, 3, \ j = 0, 1, 2, \cdots. \tag{20}$$

To start with, we use Table 6 to compute $\tilde{\mu}_{0 v_i}$ for each of the zeros v_i. These numbers (computed to 50D with Maple V and correctly rounded) are given in Table 1. Of

v_1	$9.229106324837305 \times 10^{-1}$
v_2	$1.262616187530857 \times 10^{+0}$
v_3	$3.334956765605249 \times 10^{-1}$

TABLE 1
Values of $\tilde{\mu}_{0 v_i}$ for the three zeros of $h_3(t)$.

course, $\mu_0 = 1$. We next use Algorithm 4.3 (choosing the IC method option) to compute the three Jacobi matrices $\tilde{\boldsymbol{J}}_{v_i}$, $i = 1, 2, 3$, each of which corresponds to the measure $e^{-t}/(t - v_i)$. We now apply the method SUM, referred to in §1, to $\tilde{\boldsymbol{J}}_{v_2}$ and $\tilde{\boldsymbol{J}}_{v_3}$ to compute the Jacobi matrix, $\boldsymbol{J}_{temp_1}$, corresponding to the sum of the two positive terms $e^{-t} A_2/(t - v_2)$ and $e^{-t} A_3/(t - v_3)$. The constants ρ_i, $i = 1, 2$ of (3) are here chosen as $A_i \tilde{\mu}_{0 v_i}/(\tilde{\mu}_{0 v_2} + \tilde{\mu}_{0 v_3})$. Next, we use the differencing version of SUM to compute the Jacobi matrix, $\boldsymbol{J}_{temp_2}$ corresponding to the measure

$$e^{-t} \left(\frac{A_2}{t - v_2} + \frac{A_3}{t - v_3} \right) + \frac{e^{-t} A_1}{t - v_1}.$$

(recall that $A_1 < 0$) from $\boldsymbol{J}_{temp_1}$ and $\boldsymbol{J}_{v_1}$. Finally, we apply the PSIQR method (see §1) to $\boldsymbol{J}_{temp_2}$ with the polynomial $g_3(t)$ to get $\hat{\boldsymbol{J}}$ corresponding to $r(t) d\sigma(t)$. After the PSIQR process we discard the last 2 rows and columns of the matrix $\hat{\boldsymbol{J}}$.

The resulting matrix has diagonal α_j and subdiagonal β_j components shown in Table 2 and the 7-point Gauss quadrature formula

$$Q(f) = \sum_{j=1}^{n} w_j f(t_j) \tag{21}$$

found from it is in Table 3.

Of course, this Jacobi matrix determines *all* the Gauss quadratures (for the modified measure $r(t) e^{-t}$) with seven or fewer points.

It is worth noting that we could have run the processes in a different order: start with the Jacobi matrix for e^{-t}, apply the PSIQR method with polynomial $g_3(t)$ and apply the IC method to this matrix once for each of the poles of $r(t)$. The three resulting matrices could then be merged using the SUM method.

Although it is not a reliable test of the *accuracy* of a quadrature fromula, we applied $Q(f)$ to the powers of t to confirm that it is indeed the required formula. We expect $Q(f)$ to compute the integrals

$$I_{apprx}(t^j) = \int_0^\infty t^j e^{-t} r(t) dt \tag{22}$$

j	α_j	β_j
1	$1.4735833243222072 \times 10^{+0}$	$1.1522306687481318 \times 10^{+0}$
2	$3.3670760331235998 \times 10^{+0}$	$2.1334246051843322 \times 10^{+0}$
3	$5.3094870243833645 \times 10^{+0}$	$3.1192467212534445 \times 10^{+0}$
4	$7.2722958664718185 \times 10^{+0}$	$4.1087092689689841 \times 10^{+0}$
5	$9.2458162399428527 \times 10^{+0}$	$5.1005612164049339 \times 10^{+0}$
6	$1.1225756881713069 \times 10^{+1}$	$6.0940348010727305 \times 10^{+0}$
7	$1.3209894402244561 \times 10^{+1}$	–

TABLE 2
The Jacobi matrix for $r(t)e^{-t}$.

j	t_j	w_j
1	$3.5062711038856043 \times 10^{-1}$	$3.8341359149652016 \times 10^{-1}$
2	$1.2766545994519591 \times 10^{+0}$	$6.2342241377367935 \times 10^{-1}$
3	$2.8684045755244458 \times 10^{+0}$	$2.7006149975344507 \times 10^{-1}$
4	$5.2294260605084268 \times 10^{+0}$	$4.2479756575009894 \times 10^{-2}$
5	$8.5281548379832977 \times 10^{+0}$	$2.3626792343825810 \times 10^{-3}$
6	$1.3090876238966882 \times 10^{+1}$	$3.6376377487377584 \times 10^{-5}$
7	$1.9759766349377919 \times 10^{+1}$	$7.4770525561381938 \times 10^{-8}$

TABLE 3
The 7-point Gauss quadrature formula for $r(t)e^{-t}$.

to within a modest multiple of machine accuracy for the range $j = 0, 1, 2, \ldots, 13$ and we expect $I_{apprx}(t^j) - Q(t^j)$ to be quite different from (machine) zero for $j = 14$. Further, we expect that $Q(f)$ will approximate

$$I_{exact}(t^j) = \int_0^\infty t^j e^{-t} \ln(1+4t)dt \tag{23}$$

to roughly the accuracy that $I_{apprx}(t^j)$ approximates $I_{exact}(t^j)$. In fact, $I_{apprx}(t^j)$ approximates $I_{exact}(t^j)$ with a relative error that varies almost linearly between 0.0145 and 0.2376 as j ranges from 0 to 14.

Once $\tilde{\mu}_{0v_i}$ is known, the numbers $\tilde{\mu}_{jv_i}$, $j > 0$ can easily be computed from the recursion

$$\tilde{\mu}_{jv} = v\tilde{\mu}_{j-1,v} + \mu_{j-1},\ j > 0. \tag{24}$$

From these and the partial fraction decomposition, we can find the $I_{apprx}(t^j)$ of (22). Furthermore, the numbers $\zeta_j = I_{exact}(t^j)$ of (23) can be found from the recursion

$$\zeta_j = j\zeta_{j-1} + \tilde{\mu}_{j,-\frac{1}{4}},\ \zeta_0 = \tilde{\mu}_{0,-\frac{1}{4}}$$

once (24) has been used with $v = -1/4$ to compute the $\tilde{\mu}_{j,-\frac{1}{4}}$. The absolute values of the relative errors for $Q(t^j)$ approximating the numbers in (22) and (23) are summarized in Table 4 and follow our expectations.

6. Numerical Tests. In this section we report the results of some tests which compare the performance of the IC method and the MS method and which identify two significant differences in the way these methods behave. Our tests were applied to all the classical measures shown in Table 5. In all cases the calculations were done

	Error of $Q(t^j)$ for	
j	$I_{exact}(t^j)$	$I_{apprx}(t^j)$
0	1.4×10^{-2}	6.0×10^{-15}
1	2.9×10^{-2}	5.0×10^{-14}
2	4.5×10^{-2}	5.2×10^{-14}
3	6.2×10^{-2}	3.3×10^{-14}
4	7.9×10^{-2}	5.7×10^{-14}
5	9.4×10^{-2}	1.5×10^{-13}
6	1.1×10^{-1}	3.0×10^{-13}
7	1.2×10^{-1}	3.9×10^{-13}
8	1.3×10^{-1}	4.3×10^{-13}
9	1.5×10^{-1}	6.4×10^{-13}
10	1.6×10^{-1}	6.0×10^{-13}
11	1.7×10^{-1}	5.0×10^{-13}
12	1.8×10^{-1}	3.8×10^{-13}
13	1.8×10^{-1}	1.0×10^{-13}
14	1.9×10^{-1}	2.2×10^{-4}

TABLE 4
Relative errors of $Q(t^j)$

in IEEE standard double precision arithmetic (machine epsilon $\approx 2.2 \times 10^{-16}$). The tables in this section show the accuracy of computed Jacobi matrices when compared with the explicitly known results. For each real knot v, or complex knot $x + iy$, the entry headed N in the table shows the number $N = n(m, v)$ of recurrence coefficients which the MS method requires. Where this number was considered too large to be feasible we used 2000 coefficients and display the required N in parentheses. The remaining columns show the accuracy of the diagonal α_j and subdiagonal β_j elements produced by the IC and MS methods. Thus for the columns headed β we show

$$\min_j \left\{ -\log_{10} \left| \frac{\beta_j^{exact} - \beta_j^{approx}}{\beta_j^{exact}} \right| \right\}.$$

and, except where they are zero by virtue of the symmetry of the measure $d\sigma(t)$, similar quantities for the diagonals α_j.

Recall that the IC method produces a matrix $\boldsymbol{L}$ from which we compute the required Jacobi matrix using (9). We computed the 2-norm condition numbers for the $\boldsymbol{L}$ matrices that are generated by the IC process for these tests and found that they remain modest for all the cases shown. For $n = 10$ all but 5 are smaller than 80, and the balance are smaller than 700. For $n = 50$ they are all smaller than 10^4.

The tests themselves exploit the fact that the PSIQR and either the IC or MS are inverses of each other if the polynomial $r(t)$ is a real linear factor ($r(t) = t - v$) or the product of a complex conjugate pair of factors ($r(t) = ((t - x)^2 + y^2)$, $y > 0$). Since the PSIQR method produces matrices which are accurate to about machine epsilon, the difference between the starting matrix and the result, after applying both the PSIQR and either IC or MS, is a good measure of the error. This error is just what we display in the tables of this section.

The tables illustrate that the MS method performance improves as the point v moves away from the support of $d\sigma(t)$ and also shows that the MS method is better overall on measures which have finite support. By contrast, the IC method

performance improves as the point v approaches the support interval and is better overall for the measures with infinite support.

These phenomena are clearly shown as the point v approaches the support interval from the left, in the case of linear factors (Tables 7 and 8) and as it approaches the origin along the line $y = x$ in the case of complex pair factors or for the Hermite measure (Tables 9, 10 and 11).

Table 12 shows the errors for the methods when the point $x + iy$ moves parallel but close to the x-axis from $x = 5$ to $x = -5$ and illustrates further the behaviour of the two methods close to and far from the support.

Most of the tables in this section show the results of the tests applied to produce matrices of dimension $n = 10$. However, these tables are representative of larger cases as the examples in Tables 13 and 14, with $n = 50$, show. The loss of accuracy seems to be quite gradual as n increases.

We used the non-symmetric Jacobi measure (with measure parameters $\alpha = -1/2$, $\beta = 1/7$) as an example of a finite support measure to illustrate the performance of the methods on both the α_j and the β_j components of the matrix. The Legendre, Gegenbauer and Chebyshev measures give similar results.

The Hermite and Laguerre measures used here have the measure parameter $\alpha = -1/3$. The calculation seems, however, to be relatively insensitive to this parameter and other values of α that we tested gave very similar results.

These differences in performance suggest a hybrid method for the solution of Problem 1.1. The decision about which of the IC or MS methods to choose in Algorithm 4.3 should be driven by the phenomena the results here indicate: the closeness of the point in question to the support interval and whether or not the suport interval is infinite or semi-infinite. Where either method is suitable they could both be used and the results compared for verification.

7. Conclusions. We have derived a new method, called the IC method, for the computation of the Jacobi matrix corresponding to a measure modified by a rational function. Our method is based on a new factoring algorithm which can be used to produce the Jacobi matrix corresponding to a known measure divided by a linear or quadratic factor. We use the factoring algorithm once for each pole or conjugate pair of poles and then apply the authors' SUM method to find the Jacobi matrix corresponding to the given measure divided by the denominator of the rational function. We then apply the PSIQR method to this for the numerator.

The IC method requires certain starting values which are expressed, for the classical measures, in terms of well known special functions. We computed these to very high accuracy without difficulty by use of the symbolic manipulators Maple and Mathematica. Subroutine approximations for these special functions are also readily available from software libraries such as Netlib.

We have shown that the product strategy of §4 has serious problems and that our new summing strategy overcomes these problems. We have also shown that the IC method performs best just where the previously known MS is weakest and vice versa. This has led to a hybrid strategy which we believe is the most powerful yet for this problem.

The new method was demonstrated on a simple example and some tables, which are representative of the numerical tests we ran, show the pattern of performance.

Problem 1.1 remains difficult and a stable method which does not require the poles of the rational function to be known is still desirable.

8. Tables.

8.1. Zero-th moments of classical measures and divided classical measures. The following are sometimes referred to as the classical measures:

Name	$d\sigma(t)$	Interval	Constraints
Legendre	1	$[-1,1]$	
Chebyshev	$(1-t^2)^{-1/2}$	$[-1,1]$	
Gegenbauer	$(1-t^2)^{\alpha}$	$[-1,1]$	$\alpha > -1$
Jacobi	$(1-t)^{\alpha}(1+t)^{\beta}$	$[-1,1]$	$\alpha, \beta > -1$
Laguerre	$t^{\alpha}e^{-t}$	$[0,\infty]$	$\alpha > -1$
Hermite	$\lvert t\rvert^{\alpha}e^{-t^2}$	$[-\infty,\infty]$	$\alpha > -1$

TABLE 5
Classical measures.

The table below shows the zero-th moments, μ_0, of these measures and the zero-th moments of these when divided by a linear factor,

$$\tilde{\mu}_0 = \int_a^b \frac{1}{t-v} d\sigma(t), \quad v = x + iy \notin [a,b].$$

In the case of the Hermite measure we require $y > 0$, but v may be real for the other measures.

Name	μ_0	$\tilde{\mu}_0$
Legendre	2	$\ln((v-1)/(v+1))$
Chebyshev	π	$\pi/\sqrt{v^2-1}$
Gegenbauer	$\frac{2^{2\alpha+1}\Gamma(\alpha+1)^2}{\Gamma(2\alpha+2)}$	$\frac{2^{2\alpha+1}}{(1-v)} B(1+\alpha,1+\alpha)_2F_1(1,\alpha+1;2\alpha+2;\frac{2}{1-v})$
Jacobi	$\frac{2^{\alpha+\beta+1}\Gamma(\alpha+1)\Gamma(\beta+1)}{\Gamma(\alpha+\beta+2)}$	$\frac{2^{\alpha+\beta+1}}{(1-v)} B(1+\alpha,1+\beta)_2F_1(1,\alpha+1;\alpha+\beta+2;\frac{2}{1-v})$
Laguerre ($\alpha = 0$)	1	$e^{-v}E_1(-v)$
Hermite ($\alpha = 0$)	$\sqrt{\pi}$	$i\pi e^{-v^2} erfc(-iz)$

TABLE 6
Moments of classical measures.

Here the Gamma, Beta, and Hypergeometric functions are defined by

$$\begin{aligned} \Gamma(z) &= \int_0^{\infty} t^{z-1}e^{-t}dt, \quad Re(z) > 0, \\ B(z,u) &= \Gamma(z)\Gamma(u)/\Gamma(z+u), \end{aligned}$$

$$ {}_2F_1(a,b;c;z) \quad = \quad \frac{\Gamma(c)}{\Gamma(a)\Gamma(b)} \sum_{j=0}^{\infty} \frac{\Gamma(a+j)\Gamma(b+j)}{\Gamma(c+j)} \frac{z^j}{j!}. $$

Further,

$$ E_1(v) = \int_1^{\infty} \frac{e^{-vt}}{t} dt, \quad Re(v) > 0 $$

is the Exponential Integral and $erfc(z)$ is the Complementary Error Function

$$ erfc(z) = \frac{2}{\sqrt{\pi}} \int_z^{\infty} e^{-t^2} dt. $$

For details, see [1].

Built-in functions are available in the symbolic manipulators Maple and Mathematica for all the entries in the tables above and we have used them to compute high precision zero-th moments, for real and complex v without difficulty.

8.2. Tables for the results of numerical tests.

Linear factor

		-log(Maximum errors)			
		IC		MS	
v	N	α	β	α	β
-2.000	39	1.9	6.1	11.8	15.0
-1.500	44	4.6	8.6	12.2	15.0
-1.100	66	9.3	13.0	11.4	15.0
-1.010	153	11.2	14.5	11.7	15.0
-1.001	428	11.0	14.3	11.3	14.5

TABLE 7

Errors for Jacobi measure $\alpha = -1/3, \beta = 1/7$, $n = 10$

Linear factor

		-log(Maximum errors)			
		IC		MS	
v	N	α	β	α	β
-2.000	127	10.8	11.1	11.0	11.3
-1.000	194	12.3	12.4	10.1	10.4
-0.500	312	14.0	14.3	9.0	9.6
-0.100	1000(1117)	14.9	14.6	7.2	7.7
-0.010	1000(9033)	14.3	13.8	7.2	7.5
-0.001	1000(84032)	14.4	14.0	5.9	6.3

TABLE 8

Errors for Laguerre measure $\alpha = -1/3$, $n = 10$

Quadratic factor $v = x + iy$

			-log(Maximum errors)			
			IC		MS	
x	y	N	α	β	α	β
2.000	2.000	37	0.0	2.4	11.9	15.4
1.000	1.000	44	3.8	7.9	12.1	15.2
0.500	0.500	61	8.0	11.7	12.0	15.4
0.200	0.200	116	10.3	14.5	11.7	15.0
0.100	0.100	207	10.5	14.4	12.4	15.1
0.010	0.010	1829	11.4	14.4	11.5	14.3
0.001	0.001	2000(18049)	11.1	14.3	10.3	12.9

TABLE 9

Errors for Jacobi measure $\alpha = -1/3, \beta = 1/7, n = 10$

Quadratic factor $v = x + iy$

			-log(Maximum errors)			
			IC		MS	
x	y	N	α	β	α	β
2.000	2.000	366	13.6	13.8	8.1	8.6
1.000	1.000	622	13.0	13.3	7.6	7.9
0.500	0.500	1098	13.0	13.1	6.7	7.1
0.200	0.200	2000(2442)	12.0	12.0	5.5	6.3
0.100	0.100	2000(4591)	12.1	12.0	6.3	7.0
0.010	0.010	2000(41270)	11.9	12.4	3.0	3.4

TABLE 10

Errors for Laguerre measure $\alpha = 0, n = 10$

Quadratic factor $v = x + iy$

			-log(Maximum errors)			
			IC		MS	
x	y	N	α	β	α	β
2.000	2.000	134	-	10.4	-	15.2
1.000	1.000	322	-	13.6	-	13.7
0.500	0.500	941	-	14.2	-	12.9
0.200	0.200	2000(4749)	-	14.4	-	12.1
0.100	0.100	2000(17591)	-	14.7	-	11.0
0.010	0.010	2000(1637201)	-	14.9	-	11.0

TABLE 11

Errors for Hermite measure $\alpha = -1/3, n = 10$

Quadratic factor $v = x + iy$

			-log(Maximum errors)			
			IC		MS	
x	y	N	α	β	α	β
5.000	0.100	2000	13.9	13.7	5.5	6.0
2.000	0.100	2000	13.7	13.4	4.7	5.2
0.500	0.100	2000	13.2	13.0	4.7	5.2
0.000	0.100	2000	12.8	12.4	5.0	5.3
-0.500	0.100	317	12.0	12.2	6.8	7.3
-1.000	0.100	200	11.2	11.4	7.2	7.7
-2.000	0.100	132	9.6	9.8	9.2	9.5
-5.000	0.100	84	6.0	6.4	11.1	11.5

TABLE 12

Errors for Laguerre measure, v moving parallel to x axis. $\alpha = 0, n = 10$

Quadratic factor $v = x + iy$

			-log(Maximum errors)			
			IC		MS	
x	y	N	α	β	α	β
0.200	0.200	196	-	6.5	-	14.7
0.100	0.100	287	-	10.7	-	14.8
0.010	0.010	1909	-	13.0	-	14.2

TABLE 13

Errors for Legendre measure $n = 50$

Quadratic factor $v = x + iy$

			-log(Maximum errors)			
			IC		MS	
x	y	N	α	β	α	β
0.200	0.200	2000(2980)	9.7	9.7	-1.0	0.1
0.100	0.100	2000(5319)	9.1	9.2	-0.8	-0.3
0.010	0.010	2000(43398)	8.5	8.5	-0.2	-0.1

TABLE 14

Errors for Laguerre measure $n = 50$

REFERENCES

[1] M. Abramowitz and I.A. Stegun. *Handbook of mathematical functions.* Dover Publications, New York, 1972.

[2] T.S. Chihara. *An Introduction to orthogonal polynomials.* Gordon and Breach, New York, 1978.

[3] S. Elhay, G.H. Golub, and J. Kautsky. Jacobi matrices for sums of weight functions. *BIT*, 32:143–166, 1992.

[4] S. Elhay and J. Kautsky. Generalized Kronrod Patterson type imbedded quadratures. *Aplikace Matematiky*, 37(2):81–103, 1992.

[5] W. Gautschi. Minimal solutions of three-term recurrence relations and orthogonal polynomials. *Math. Comp.*, 36(154):547–554, 1981.

[6] W. Gautschi. On generating orthogonal polynomials. *SIAM J. Sci. Statist. Comput.*, 3:289–317, 1982.

[7] J. Kautsky and S. Elhay. Gauss quadratures and Jacobi matrices for weight functions not of one sign. *Math. Comp.*, 43(168):543–550, 1984.

[8] J. Kautsky and G.H. Golub. On the calculation of Jacobi matrices. *Linear Algebra Appl.*, 52/53:439–455, 1983.

[9] R.A. Sack and A.F. Donovan. An algorithm for Gaussian quadrature given modified moments. *Numer. Math.*, 18:465–478, 1971/72.

[10] G. Szego. *Orthogonal polynomials.* AMS Colloquium Publications 23. Americam Mathematical Society, Providence, Rhode, RI, 4th edition, 1975.

[11] V.B. Uvarov. The connection between systems of polynomials orthogonal with respect to different distribution functions. *USSR Computational Math. and Math. Phys.*, 9(6):25–36, 1969.