COMPUTATION OF GAUSS-KRONROD QUADRATURE RULES WITH NON-POSITIVE WEIGHTS

G. S. AMMAR [*], D. CALVETTI [†], AND L. REICHEL [‡]

Abstract. Recently Laurie presented a fast algorithm for the computation of $(2n+1)$-point Gauss-Kronrod quadrature rules with real nodes and positive weights. We describe modifications of this algorithm that allow the computation of Gauss-Kronrod quadrature rules with complex conjugate nodes and weights or with real nodes and positive and negative weights.

Key words. orthogonal polynomials, indefinite measure, fast algorithm, inverse eigenvalue problem

1. Introduction.

Let dw be a nonnegative measure with support on the real axis and an infinite number of points of increase. Assume that the moments $\mu_k := \int_{-\infty}^{\infty} x^k dw(x)$, $k = 0, 1, 2, \ldots$, exist and are bounded. For notational convenience, we assume that $\mu_0 = 1$. An n-point Gauss quadrature rule for the integral

$$\mathcal{I}f := \int_{-\infty}^{\infty} f(x)dw(x) \tag{1.1}$$

is a formula of the form

$$\mathcal{G}_n f := \sum_{k=1}^{n} f(x_k)w_k \tag{1.2}$$

with nodes $x_1 < x_2 < \ldots < x_n$ in the convex hull of the support of the measure dw and positive weights w_k, such that

$$\mathcal{G}_n f = \mathcal{I}f \qquad \forall f \in \mathbb{P}_{2n-1}. \tag{1.3}$$

Here and throughout this paper $\mathbb{P}_j$ denotes the set of polynomials of degree at most j. The $(2n+1)$-point Gauss-Kronrod quadrature rule associated with the Gauss rule (1.2) is an integration rule of the form

$$\mathcal{K}_{2n+1} f := \sum_{k=1}^{2n+1} f(\tilde{x}_k)\tilde{w}_k, \tag{1.4}$$

such that

$$\mathcal{K}_{2n+1} f = \mathcal{I}f \qquad \forall f \in \mathbb{P}_{3n+1} \tag{1.5}$$

and

$$\{x_k\}_{k=1}^{n} \subset \{\tilde{x}_k\}_{k=1}^{2n+1}. \tag{1.6}$$

[*] Department of Mathematical Sciences, Northern Illinois University, DeKalb, IL 60115. E-mail: ammar@math.niu.edu.

[†] Department of Mathematics, Case Western Reserve University, Cleveland, OH 44106. E-mail: dxc57@po.cwru.edu. Research supported in part by NSF grant DMS-9806702.

[‡] Department of Mathematics and Computer Science, Kent State University, Kent, OH 44242. E-mail: reichel@mcs.kent.edu. Research supported in part by NSF grant DMS-9806413.

For notational simplicity we assume throughout this paper that

$$\tilde{x}_k = x_k, \qquad 1 \leq k \leq n. \tag{1.7}$$

We refer to the nodes (1.7) as Gauss nodes and the remaining nodes $\{\tilde{x}_k\}_{k=n+1}^{2n+1}$ as Kronrod nodes.

Pairs of Gauss and Gauss-Kronrod rules (1.2) and (1.4) are often evaluated together in order to determine accurate approximations with error estimates of integrals (1.1). Properties of Gauss-Kronrod rules (1.4) can be investigated by studying the Stieltjes polynomial $s_{n+1}(x) := \prod_{k=n+1}^{2n+1}(x - \tilde{x}_k)$, whose zeros are the Kronrod nodes; see Monegato [16] and the recent paper by Ehrich and Mastroianni [5]. Nice surveys of Gauss-Kronrod rules and their properties are provided by Gautschi [7] and Laurie [13]; see also Gautschi [8] for a recent discussion and further references.

It is known that Gauss-Kronrod quadrature rules, i.e., rules with the properties (1.5) and (1.6), do not always exist. If the $(2n + 1)$-point Gauss-Kronrod rule (1.4) does exist, then the Kronrod nodes may be real or appear in complex conjugate pairs. Weights $\tilde{w}_k$ associated with complex conjugate Kronrod nodes are complex conjugate. Note that the nonnegativity of the measure dw implies that the Gauss nodes are real. Gauss-Kronrod rules with real nodes may have positive or negative weights, and the nodes may or may not be contained in the smallest interval containing the support of dw. Gautschi [7, p. 52] notes that "Little has been *proved* with regard to these properties; any new piece of information, from whatever source - computational or otherwise - should therefore be greeted with appreciation." It is our hope that the algorithms of the present paper will be helpful in shedding light on these questions, as well as be useful for the computation of Gauss-Kronrod rules required in applications.

Laurie [13], and more recently Calvetti et al. [3], presented efficient algorithms that require only $\mathcal{O}(n^2)$ arithmetic operations for the computation of the nodes and weights of $(2n+1)$-point Gauss-Kronrod quadrature rules with real nodes and positive weights. This paper describes modifications of Laurie's algorithm that allow the computation of the nodes and weights of $(2n + 1)$-point Gauss-Kronrod rules with complex conjugate nodes, or with real nodes and positive and negative weights, in $\mathcal{O}(n^2)$ or $\mathcal{O}(n^3)$ arithmetic operations. The faster algorithm yields nodes and weights with sufficient accuracy for most applications. The slower algorithm gives higher accuracy for certain difficult problems.

The present paper is organized as follows. Section 2 reviews results by Laurie [13] and discusses modifications required for the computation of Gauss-Kronrod rules with complex conjugate nodes, or with real nodes and positive and negative weights. The new algorithms are described in Section 3, and Section 4 presents computed examples. Concluding remarks and a discussion of an extension can be found in Section 5. Throughout this paper we let $i := \sqrt{-1}$.

2. Some tridiagonal matrices.

Let $\{p_j\}_{j=0}^{\infty}$ be a sequence of monic orthogonal polynomials with respect to the inner product

$$(f, g) := \int_{-\infty}^{\infty} f(x)g(x)dw(x), \tag{2.1}$$

i.e.,

$$\deg(p_j) = j; \qquad (p_j, p_k) = 0, \qquad j \neq k. \tag{2.2}$$

The p_j satisfy the recursion relations

$$\begin{aligned} p_{k+1}(x) &= (x-a_k)p_k(x) - b_k^2 p_{k-1}(x), \qquad k=1,2,\dots, \\ p_1(x) &:= x-a_0, \qquad p_0(x) := 1, \end{aligned} \tag{2.3}$$

with coefficients

$$a_k := \frac{(p_k, xp_k)}{(p_k,p_k)}, \qquad k=0,1,\dots\ , \tag{2.4}$$

$$b_k^2 := \frac{(p_k,p_k)}{(p_{k-1},p_{k-1})}, \qquad k=1,2,\dots\ . \tag{2.5}$$

We will assume that the coefficients a_k and b_k are explicitly known. When only the measure dw is available, these coefficients can be computed by (2.4) and (2.5). It may be attractive to evaluate necessary inner products by a Clenshaw-Curtis quadrature rule; see Gautschi [6] for a discussion.

Laurie [13] and Calvetti et al. [3] presented efficient algorithms for the computation of nodes and weights of Gauss-Kronrod rules (1.4) with distinct real nodes and positive weights. These algorithms are based on Laurie's observation that to each $(2n+1)$-point Gauss-Kronrod quadrature rule with real nodes and positive weights, there is associated a real symmetric tridiagonal $(2n+1)\times(2n+1)$ matrix with positive subdiagonal entries

$$\tilde{T}_{2n+1} := \begin{bmatrix} \tilde{a}_0 & \tilde{b}_1 & & & \\ \tilde{b}_1 & \tilde{a}_1 & \tilde{b}_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \tilde{b}_{2n-1} & \tilde{a}_{2n-1} & \tilde{b}_{2n} \\ & & & \tilde{b}_{2n} & \tilde{a}_{2n} \end{bmatrix}. \tag{2.6}$$

We refer to this matrix as the Gauss-Kronrod matrix. Let $\tilde{T}_{2n+1}$ have spectral factorization

$$\tilde{T}_{2n+1} = \tilde{W}_{2n+1}\tilde{\Lambda}_{2n+1}\tilde{W}_{2n+1}^{-1}, \qquad \tilde{\Lambda}_{2n+1} = \mathrm{diag}[\tilde{\lambda}_1, \tilde{\lambda}_2, \dots, \tilde{\lambda}_{2n+1}]. \tag{2.7}$$

Since $\tilde{T}_{2n+1}$ is symmetric, the columns of the eigenvector matrix can be scaled so that

$$\tilde{W}_{2n+1}^{-1} = \tilde{W}_{2n+1}^{T}. \tag{2.8}$$

The nodes and weights of the Gauss-Kronrod quadrature rule (1.4) are then given by

$$\left\{ \begin{array}{l} \tilde{x}_j = \tilde{\lambda}_j, \\ \tilde{w}_j = (\boldsymbol{e}_1^T \tilde{W}_{2n+1}\boldsymbol{e}_j)^2, \end{array} \right. \qquad 1 \le j \le 2n+1; \tag{2.9}$$

see Golub and Welsch [11] for a discussion. Here we only note that formula (2.9) requires that the normalization (2.8) holds. We refer to the set

$$\{\tilde{\lambda}_j, \boldsymbol{e}_1^T \tilde{W}_{2n+1}\boldsymbol{e}_j\}_{j=1}^{2n+1} \tag{2.10}$$

as the partial spectral resolution of the matrix $\tilde{T}_{2n+1}$. The positivity of the subdiagonal entries of $\tilde{T}_{2n+1}$ implies that the eigenvalues $\tilde{\lambda}_j$, and therefore the nodes $\tilde{x}_j$, are distinct.

Laurie's algorithm [13] for determining the nodes and weights of a $(2n+1)$-point Gauss-Kronrod quadrature rule with real nodes and positive weights consists of two steps: i) compute the entries of the Gauss-Kronrod matrix (2.6) from the recursion coefficients (2.4) and (2.5), and ii) if each $\tilde{b}_k^2 > 0$ $(1 \leq k \leq 2n+1)$, compute the partial spectral resolution of the Gauss-Kronrod matrix (2.6) by the Golub-Welsch algorithm [11]. Each step requires $\mathcal{O}(n^2)$ arithmetic operations. We will discuss these steps further below. At this point we remark that for certain functions f, such as rational functions with known poles or functions that satisfy a recursion relation with few terms, the representation

$$\mathcal{K}_{2n+1} f = \boldsymbol{e}_1^T f(\tilde{T}_{2n+1}) \boldsymbol{e}_1 \tag{2.11}$$

may provide a more convenient way of evaluating the Gauss-Kronrod rule than (1.4), because (2.11) does not require the computation of the partial spectral resolution of $\tilde{T}_{2n+1}$. Formula (2.11) follows by combining (1.4) with (2.7)-(2.9); see Golub and Meurant [9].

Laurie's algorithm is based on the following key result.

PROPOSITION 2.1. *(Laurie [13]) The leading and trailing $n \times n$ principal submatrices of $\tilde{T}_{2n+1}$ have the same spectrum. Moreover, for n odd,*

$$\tilde{a}_{j-1} = a_{j-1}, \qquad \tilde{b}_j = b_j, \qquad 1 \leq j \leq \frac{3n+1}{2}, \tag{2.12}$$

and, for n even,

$$\begin{cases} \tilde{a}_j = a_j, & 0 \leq j \leq \frac{3n}{2}, \\ \tilde{b}_j = b_j, & 1 \leq j \leq \frac{3n}{2}, \end{cases} \tag{2.13}$$

where the a_j and b_j are given by (2.4) and (2.5).

Example 2.1. Let $n = 2$. The entries $\{\tilde{a}_j\}_{j=0}^3$ and $\{\tilde{b}_j\}_{j=1}^3$ of the Gauss-Kronrod matrix $\tilde{T}_5$ are recursion coefficients for orthogonal polynomials associated with the measure dw, but the entries marked by $*$ are not explicitly known,

$$\tilde{T}_5 := \begin{bmatrix} \tilde{a}_0 & \tilde{b}_1 & & & \\ \tilde{b}_1 & \tilde{a}_1 & \tilde{b}_2 & & \\ & \tilde{b}_2 & \tilde{a}_2 & \tilde{b}_3 & \\ & & \tilde{b}_3 & \tilde{a}_3 & * \\ & & & * & * \end{bmatrix}. \tag{2.14}$$

It follows from Proposition 2.1 that the leading and trailing principal 2×2 submatrices of $\tilde{T}_5$ have the same trace. This yields the equation

$$\tilde{a}_0 + \tilde{a}_1 = \tilde{a}_3 + \tilde{a}_4 \tag{2.15}$$

for $\tilde{a}_4$. The determinants of the leading and trailing principal 2×2 submatrices are also the same, and this gives the equation

$$\tilde{a}_0 \tilde{a}_1 - \tilde{b}_1^2 = \tilde{a}_3 \tilde{a}_4 - \tilde{b}_4^2 \tag{2.16}$$

for $\tilde{b}_4$. When (2.16) is satisfied by a real positive value of $\tilde{b}_4$, a Gauss-Kronrod rule with real nodes and positive weights exists. A purely imaginary solution $\tilde{b}_4$ of (2.16) signals that the Gauss-Kronrod quadrature rule either has complex conjugate nodes or real nodes and positive and negative weights. □

Example 2.2. Let $n = 2$ and consider the measure associated with the Hermite polynomials, $dw(x) := \pi^{-1/2}\exp(-x^2)dx.$ Then the recurrence coefficients (2.4) and (2.5) are given by $a_j = 0$ and $b_j = \sqrt{j/2}$. Equation (2.15) yields that $\tilde{a}_4 = 0$ and by equation (2.16) $\tilde{b}_4^2 = \tilde{b}_1^2$. We can choose $\tilde{b}_4 = \tilde{b}_1$, which shows that the 5-point Gauss-Kronrod quadrature rule has distinct real nodes and positive weights.

Let $n = 3$ instead. Then the Gauss-Kronrod matrix $\tilde{T}_7$ is complex symmetric with all entries real, except for $\tilde{b}_7 = i$. The Gauss-Kronrod rule has one pair of complex conjugate nodes, see Example 4.1 below, in agreement with the discussion by Monegato [15]. □

Example 2.3. Let $n = 2$ and consider the measure associated with the Laguerre polynomials,

$$dw(x) := \left\{ \begin{array}{cl} e^{-x}dx, & x \geq 0, \\ 0, & x < 0. \end{array} \right. \tag{2.17}$$

Then the recursion coefficients (2.4) and (2.5) are given by $a_j = 2j + 1$ and $b_j = j$. Equations (2.15) and (2.16) yield $\tilde{a}_4 = -3$ and $\tilde{b}_4^2 = -23$, respectively; hence $\tilde{b}_4 = i\sqrt{23}$. The 5-point Gauss-Kronrod quadrature rule has one pair of complex conjugate nodes; see Example 4.2 and the discussion in [15]. □

By Proposition 2.1 about $\frac{3}{4}$th of the entries of the Gauss-Kronrod matrix (2.6) are known. Laurie [13] observed that the entries of the trailing $n \times n$ principal submatrix $\breve{T}_n$ of (2.6) are recursion coefficients for a family of orthogonal polynomials $\{\breve{p}_j\}_{j=0}^{n-1}$ with respect to a bilinear form

$$< f, g >:= \int_{-\infty}^{\infty} f(x)g(x)d\breve{w}(x). \tag{2.18}$$

The measure $d\breve{w}$ is not explicitly known, and is not unique. Laurie [13] showed that the unknown entries of $\breve{T}_n$, and thereby of $\tilde{T}_{2n+1}$, can be computed in $\mathcal{O}(n^2)$ arithmetic operations by applying recursion formulas closely related to those used in the modified Chebyshev algorithm; see also Gautschi [8] for a discussion. The Gauss-Kronrod matrices (2.6) generated in this manner belong to $\mathbb{T}_{2n+1}$, the set of complex symmetric tridiagonal matrices of order $2n + 1$ with real diagonal entries and real or purely imaginary subdiagonal elements. Since, in general, (2.18) is not an inner product, it may happen that $< \breve{p}_j, \breve{p}_j >= 0$ for some index $j < n - 1$. In these (rare) cases the Gauss-Kronrod matrix (2.6) cannot be computed. We will assume that $< \breve{p}_j, \breve{p}_j >\neq 0$ for $0 \leq j \leq n$. Then the Gauss-Kronrod matrix (2.6) exists and has nonvanishing subdiagonal entries.

Laurie's scheme [13, Appendix A] for computing the unknown entries of the Gauss-Kronrod matrix (2.6) actually yields the elements of the real tridiagonal matrix

$$\hat{S}_{2n+1} := \begin{bmatrix} \tilde{a}_0 & 1 & & & \\ \tilde{b}_1^2 & \tilde{a}_1 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & \tilde{b}_{2n-1}^2 & \tilde{a}_{2n-1} & 1 \\ & & & \tilde{b}_{2n}^2 & \tilde{a}_{2n} \end{bmatrix} \in \mathbb{R}^{(2n+1)\times(2n+1)}, \tag{2.19}$$

which is similar to the Gauss-Kronrod matrix (2.6), i.e.,

$$\hat{S}_{2n+1} = \hat{D}_{2n+1}\tilde{T}_{2n+1}\hat{D}_{2n+1}^{-1}, \qquad \hat{D}_{2n+1} := \mathrm{diag}[1, \tilde{d}_1, \tilde{d}_2, \ldots, \tilde{d}_{2n}], \tag{2.20}$$

where

$$\tilde{d}_j := \tilde{b}_1\tilde{b}_2\cdots\tilde{b}_j, \qquad j = 1, \dots 2n. \tag{2.21}$$

The entries $\tilde{b}_j^2$ may be negative; see Example 2.3. When the matrix $\hat{S}_{2n+1}$ has one or several negative subdiagonal entries, evaluation of the Gauss-Kronrod rule by the formula

$$\mathcal{K}_{2n+1}f = \boldsymbol{e}_1^T f(\hat{S}_{2n+1})\boldsymbol{e}_1 \tag{2.22}$$

may be more convenient than by (2.11), because the latter representation requires complex arithmetic. Formula (2.22) follows from (2.11) and (2.20).

We remark that sometimes using the matrix

$$\tilde{S}_{2n+1} := \begin{bmatrix} \tilde{a}_0 & \mathrm{sign}(\tilde{b}_1^2)|\tilde{b}_1| & & & \\ |\tilde{b}_1| & \tilde{a}_1 & \mathrm{sign}(\tilde{b}_2^2)|\tilde{b}_2| & & \\ & \ddots & \ddots & \ddots & \\ & & |\tilde{b}_{2n-1}| & \tilde{a}_{2n-1} & \mathrm{sign}(\tilde{b}_{2n}^2)|\tilde{b}_{2n}| \\ & & & |\tilde{b}_{2n}| & \tilde{a}_{2n} \end{bmatrix} \tag{2.23}$$

instead of (2.19) may be preferable because it is better balanced; we will comment further on this at the end of Subsection 3.2. Here we only note that the matrix (2.23) also is diagonally similar to the Gauss-Kronrod matrix (2.6), i.e.,

$$\begin{aligned} \tilde{S}_{2n+1} &= \tilde{D}_{2n+1}\tilde{T}_{2n+1}\tilde{D}_{2n+1}^{-1}, \\ \tilde{D}_{2n+1} &:= \mathrm{diag}[1, \sqrt{\mathrm{sign}(\tilde{d}_1^2)}, \sqrt{\mathrm{sign}(\tilde{d}_2^2)}, \dots, \sqrt{\mathrm{sign}(\tilde{d}_{2n}^2)}], \end{aligned} \tag{2.24}$$

where the $\tilde{d}_j$ are given by (2.21).

For many integrands, formula (1.4) provides the most convenient way of evaluating the Gauss-Kronrod quadrature rule. This formula requires that the nodes and weights be computed. We therefore seek to develop algorithms for their efficient and accurate computation.

Golub and Welsch [11] used the connections between Gauss quadrature, orthogonal polynomials, and real symmetric tridiagonal matrices to show that the Gauss weights are the squares of the first component of the normalized eigenvectors. In fact, any real symmetric tridiagonal matrix with nonzero subdiagonal elements corresponds to a finite sequence of orthogonal polynomials for some (nonunique) nonnegative measure dw. The eigendecomposition of the matrix of order n determines the n-point Gauss rule for this family of measures.

In contrast, not every complex symmetric tridiagonal matrix with nonzero subdiagonal elements can be associated with a quadrature rule of the form (1.4), because these matrices are not guaranteed to have distinct eigenvalues. In fact, they need not be diagonalizable.

Example 2.4. Consider the matrix

$$M := \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & i \\ 0 & i & 0 \end{bmatrix} \in \mathbb{T}_3.$$

It has the eigenvalue zero of algebraic multiplicity three and geometric multiplicity one. □.

PROPOSITION 2.2. *Let M be a complex symmetric tridiagonal matrix with nonvanishing subdiagonal elements. Then M is diagonalizable if and only if it has no multiple eigenvalue.*

Proof. This follows from the well-known result that every eigenvalue of an upper Hessenberg matrix with nonzero subdiagonal elements has geometric multiplicity equal to one [10, Theorem 7.4.4]: since the subdiagonal elements of the matrix are nonzero, the nullspace of $M - \lambda I$ has dimension equal to one for every eigenvalue λ of M. □

Assume now that the eigenvalues of $\tilde{T}_{2n+1}$, and hence of $\tilde{S}_{2n+1}$, are distinct, and let

$$\tilde{S}_{2n+1} = \tilde{V}_{2n+1}\tilde{\Lambda}_{2n+1}\tilde{V}_{2n+1}^{-1}, \qquad \tilde{\Lambda}_{2n+1} = \mathrm{diag}[\tilde{\lambda}_1, \tilde{\lambda}_2, \ldots, \tilde{\lambda}_{2n+1}], \tag{2.25}$$

be a spectral factorization of $\tilde{S}_{2n+1}$.

THEOREM 2.3. *Let $\tilde{S}_{2n+1}$ be a matrix of the form (2.23) with distinct eigenvalues and spectral factorization (2.25). Then the nodes and weights of the associated Gauss-Kronrod quadrature rule (1.4) can be computed from*

$$\tilde{x}_j = \tilde{\lambda}_j, \tag{2.26}$$

$$\tilde{w}_j = (\boldsymbol{e}_j^T \tilde{V}_{2n+1}^{-1} \boldsymbol{e}_1)(\boldsymbol{e}_1^T \tilde{V}_{2n+1} \boldsymbol{e}_j), \tag{2.27}$$

for $1 \leq j \leq 2n+1$.

Proof. As shown in [1], this result follows directly from results of Gragg [12]. □

We refer to the set

$$\{\tilde{\lambda}_j, \boldsymbol{e}_j^T \tilde{V}_{2n+1}^{-1} \boldsymbol{e}_1, \boldsymbol{e}_1^T \tilde{V}_{2n+1} \boldsymbol{e}_j\}_{j=1}^{2n+1} \tag{2.28}$$

as the partial spectral resolution of the nonsymmetric tridiagonal matrix $\tilde{S}_{2n+1}$.

The following theorem discusses a structure-preserving spectral decomposition of complex symmetric tridiagonal matrices of the form generated in step i) of Laurie's algorithm.

THEOREM 2.4. *Let $\tilde{T}_{2n+1} \in \mathbb{T}_{2n+1}$ have distinct eigenvalues, and further assume that every eigenvector $\boldsymbol{x}$ of $\tilde{T}_{2n+1}$ satisfies $\boldsymbol{x}^T\boldsymbol{x} \neq 0$. Then $\tilde{T}_{2n+1}$ has a spectral factorization of the form (2.7) with a complex orthogonal eigenvector matrix $\tilde{W}_{2n+1} \in \mathbb{C}^{(2n+1)\times(2n+1)}$. The eigenvalues λ_j are real or appear in complex conjugate pairs. The nodes and weights of the Gauss-Kronrod rule (1.4) can be determined by (2.9).*

Proof. The matrix $\tilde{T}_{2n+1}$ is of the form (2.6) and is diagonally similar to the real matrix (2.23). Therefore its eigenvalues are real or appear in complex conjugate pairs. The (formal) orthogonality of the eigenvectors of $\tilde{T}_{2n+1}$ associated with distinct eigenvalues can be shown in the same way as the analogous result for real symmetric matrices. Moreover, since each eigenvector $\boldsymbol{x}$ satisfies $\boldsymbol{x}^T\boldsymbol{x} \neq 0$, the columns of the eigenvector matrix $\tilde{W}_{2n+1}$ can be scaled so that the eigenvector matrix $\tilde{W}_{2n+1}$ satisfies (2.8). Note that $\tilde{W}_{2n+1}$ might not be unitary. Formula (2.9) now follows from (2.24), (2.25) and (2.27). □

Theorem 2.4 provides the basis for a structure-exploiting algorithm that determines the partial spectral resolution (2.10) of a matrix $T_{2n+1} \in \mathbb{T}_{2n+1}$ in $\mathcal{O}(n^2)$ arithmetic operations. The algorithm is of QR type, and generates a sequence of matrices in $\mathbb{T}_{2n+1}$ similar to T_{2n+1} by applying a succession of real orthogonal and complex orthogonal similarity transformations, and is a generalization of the Golub-Welsch algorithm [11]; see Subsection 3.1 for details.

Complex orthogonal matrices can be ill-conditioned, and when very ill-conditioned similarity transformations are used in the algorithm, reduced accuracy of the computed partial spectral resolution may result. This loss of accuracy may be avoided by instead applying the standard QR algorithm for nonsymmetric Hessenberg matrices to $\tilde{S}_{2n+1}$. The latter algorithm uses only real orthogonal similarity transformations, and requires $\mathcal{O}(n^3)$ arithmetic operations because it does not preserve the tridiagonal structure of $\tilde{S}_{2n+1}$. (In [1], this technique is applied to $\hat{S}_{2n+1}$.) We will show in Subsection 3.2 how the standard QR algorithm can be used to compute the partial spectral resolution (2.10) without storing the eigenvector matrix.

3. Algorithms for computing the partial spectral resolution.

This section describes two algorithms for the computation of the nodes and weights of a $(2n+1)$-point Gauss-Kronrod rule (1.4) from its associated Gauss-Kronrod matrix (2.6) or the similar real nonsymmetric matrix (2.23). We rely on ideas related to the well-known QR algorithm. A nice presentation of the QR algorithm is provided by Watkins [20, Chapter 4]. Many issues of importance for an efficient implementation are discussed by Golub and Van Loan [10, Chapter 7].

3.1. A generalized Golub-Welsch algorithm.

We describe a generalization of the Golub-Welsch algorithm [11] that allows for the computation of the partial spectral resolution (2.10) of matrices in the set $\mathbb{T}_{2n+1}$ in $\mathcal{O}(n^2)$ arithmetic operations. The algorithm of Golub and Welsch for computing Gauss rules corresponding to a real symmetric tridiagonal matrix is based on the QR algorithm, which preserves the tridiagonal structure of the initial matrix. Our generalization relies on a QR-type algorithm, based on similarity transformations that are orthogonal, but possibly complex (and therefore non-unitary). These similarity transformations preserve the complex symmetric structure of the initial matrix, and can therefore be used to compute the partial spectral resolution in $\mathcal{O}(n^2)$ arithmetic operations.

A structure-preserving QR-type iteration for matrices in the class $\mathbb{T}_{2n+1}$, and other classes of related structures, is the HR algorithm of Bunse-Gerstner [2]. In fact, each $T \in \mathbb{T}_{2n+1}$ is a J-Hermitian matrix for some matrix $J = \mathrm{diag}[\pm1, \pm1, \ldots, \pm1]$, so the HR algorithm can be applied efficiently to such a matrix T. More recently, Cullum and Willoughby [4] considered aspects of a structure-preserving QR-type algorithm for the slightly larger class of all complex symmetric tridiagonal matrices. A related algorithm is also outlined by Luk and Qiao [14].

We describe here a generalization of the Golub-Welsch algorithm based on a modification of the complex symmetric tridiagonal QR algorithm of [4]. A generalization based on the HR algorithm of [2] can be obtained similarly.

Let $T = T^{(0)}$ be a given matrix in $\mathbb{T}_{2n+1}$. Our algorithm generates a sequence of similar matrices

$$(3.1)\quad T^{(j+1)} := Q^{(j)} T^{(j)} [Q^{(j)}]^T \in \mathbb{T}_{2n+1}, \quad [Q^{(j)}]^T = [Q^{(j)}]^{-1} \in \mathbb{C}^{(2n+1)\times(2n+1)},$$

for $j = 0, 1, \ldots$, which converge to a diagonal matrix or a block diagonal matrix with blocks of order 1 or 2. The matrices $Q^{(j)}$ are products of plane transformations, i.e., matrices that are equal to the identity matrix of appropriate size, except for a 2×2 block on the diagonal. We represent these blocks as

$$(3.2)\qquad G := \begin{bmatrix} c & -s \\ s & c \end{bmatrix}, \qquad c^2 + s^2 = 1,$$

so that $G^T = G^{-1}$. Of course, when c and s are real, G is a (unitary) Givens rotation.

Each of the matrices G is generated so that it maps a vector $\boldsymbol{v} = [a, b]^T \in \mathbb{C}^2$ to a multiple of the axis vector $\boldsymbol{e}_1$:

$$G^T v = d\boldsymbol{e}_1, \qquad \text{where } d := \sqrt{a^2 + b^2} \in \mathbb{C}.$$

If $d \neq 0$, then we can take c $= a/d$ and $s = b/d$. If both a and b are real or purely imaginary, then we can choose c and s real, and G is a real Givens matrix. If $a = b = 0$, then we take $G = I_2$; otherwise G remains undefined when $d^2 = a^2 + b^2 = 0$. Note that a complex orthogonal plane transformation G can be arbitrarily ill-conditioned. If a is real and b is purely imaginary, then we may choose c real and $s = i\sigma$, $\sigma \in \mathbb{R}$, and the condition number of G is given by

$$\kappa(G) := \left| \frac{|c| + |\sigma|}{|c| - |\sigma|} \right|. \tag{3.3}$$

The algorithm of [4] uses an implicit single-shift strategy based on Wilkinson shifts. In order to avoid possible difficulties associated with complex conjugate eigenvalues (which is a contingency that is not relevant for real matrices in $\mathbb{T}_{2n+1}$), the iteration begins with a single randomly chosen complex shift to move the matrices into the larger class of complex symmetric tridiagonal matrices. Of course this then releases the constraint that the non-real eigenvalues occur in complex conjugate pairs. We therefore implement the algorithm using the Francis double shift strategy so that the complex conjugate symmetry of the eigenvalues is preserved.

If it were known beforehand that $T^{(0)}$ had only real eigenvalues, then a single-shift strategy could be employed with real shifts. In this case the algorithm could be viewed as being the reverse of the inverse eigenvalue algorithm presented in [17] for the construction of a complex symmetric tridiagonal matrix from the partial spectral resolution (2.10).

Although Cullum and Willoughby [4] only discuss the computation of the spectrum of T, the eigenvectors can also be computed in $\mathcal{O}(n^3)$ operations by accumulating the individual plane transformations in the same way as is done in the QR algorithm when the latter is applied to a real matrix $T \in \mathbb{T}_{2n+1}$. Accordingly, the first components of the complex symmetric eigenvector matrix W can be computed by accumulating the transformations against the first axis vector, as in the Golub-Welsch algorithm.

3.2. Application of the standard QR algorithm. We outline the computations required by the QR algorithm for real symmetric upper Hessenberg matrices with distinct eigenvalues, and discuss its application to the computation of the nodes and weights of the Gauss-Kronrod quadrature rule (1.4) from the matrix $\tilde{S}_{2n+1}$ defined by (2.23). We apply the QR algorithm to $\tilde{S}_{2n+1}$ rather than to the similar Gauss-Kronrod matrix (2.6), because this reduces the complex arithmetic necessary.

The QR algorithm applies a sequence of unitary Givens similarity transformations to $\tilde{S}_{2n+1}$ to obtain a Schur factorization

$$\tilde{S}_{2n+1} = \tilde{U}_{2n+1}\tilde{R}_{2n+1}\tilde{U}^*_{2n+1}, \qquad \tilde{U}^*_{2n+1}\tilde{U}_{2n+1} = I_{2n+1}, \tag{3.4}$$

where $\tilde{R}_{2n+1} \in \mathbb{C}^{(2n+1)\times(2n+1)}$ is a an upper triangular matrix and the superscript $*$ denotes transposition and complex conjugation.

The spectral factorization of $\tilde{S}_{2n+1}$ is then given by (2.25), where the diagonal matrix $\tilde{\Lambda}_{2n+1}$ is formed from the diagonal entries of $\tilde{R}_{2n+1}$, and where the eigenvector matrix is given by

$$\tilde{V}_{2n+1} = \tilde{U}_{2n+1}\tilde{Z}_{2n+1}. \tag{3.5}$$

Here, $\tilde{Z}_{2n+1}$ is an upper triangular eigenvector matrix of $\tilde{R}_{2n+1}$, which is computed by back substitution. In view of (2.24), the matrix

$$\tilde{W}_{2n+1} = \tilde{D}_{2n+1}^{-1}\tilde{V}_{2n+1}$$

is an eigenvector matrix of the Gauss-Kronrod matrix (2.6).

The straightforward computation of the weights by formula (2.27) requires $\mathcal{O}(n^3)$ arithmetic operations. We now describe how $\{\boldsymbol{e}_j^T\tilde{V}_{2n+1}^{-1}\boldsymbol{e}_1, \boldsymbol{e}_1^T\tilde{V}_{2n+1}\boldsymbol{e}_j\}_{j=1}^{2n+1}$ can be computed in $\mathcal{O}(n^2)$ arithmetic operations without storing the eigenvector matrix $\tilde{V}_{2n+1}$. Introduce the vectors

$$\begin{aligned}\tilde{\boldsymbol{w}} &:= \tilde{V}_{2n+1}^{-1}\boldsymbol{e}_1 = \tilde{Z}_{2n+1}^{-1}\tilde{U}_{2n+1}^{*}\boldsymbol{e}_1,\\ \tilde{\boldsymbol{w}}' &:= \tilde{V}_{2n+1}^{T}\boldsymbol{e}_1 = \tilde{Z}_{2n+1}^{T}\tilde{U}_{2n+1}^{T}\boldsymbol{e}_1,\end{aligned}$$

where the right-most expressions follow from (3.5). We first compute $\tilde{U}_{2n+1}^{*}\boldsymbol{e}_1$ and $\tilde{U}_{2n+1}^{T}\boldsymbol{e}_1$ by applying the unitary Givens matrices that make up $\tilde{U}_{2n+1}$ in the order they are generated to vectors $\boldsymbol{e}_1$. Thus, the matrix $\tilde{U}_{2n+1}$ does not have to be stored. The columns of the triangular eigenvector matrix $\tilde{Z}_{2n+1}$ are generated one at a time starting with last one and ending with the first one. One column at a time is used in back substitution to compute $\tilde{\boldsymbol{w}}$ from $\tilde{U}_{2n+1}^{*}\boldsymbol{e}_1$. The matrix vector product $\tilde{Z}_{2n+1}^{T}(\tilde{U}_{2n+1}^{T}\boldsymbol{e}_1)$ which yields $\tilde{\boldsymbol{w}}'$ also is evaluated by using the columns of $\tilde{Z}_{2n+1}$ one at a time. In particular, entries of the matrix $\tilde{Z}_{2n+1}$ do not need to be stored simultaneously. The computation of $\tilde{\boldsymbol{w}}$ and $\tilde{\boldsymbol{w}}'$ as described requires $\mathcal{O}(n^2)$ arithmetic operations.

We have described how the QR algorithm can be applied to compute the partial spectral resolution of the matrix $\tilde{S}_{2n+1}$. We note that the QR algorithm can be applied to the matrix $\hat{S}_{2n+1}$ defined by (2.19) in an analogous fashion. However, in our experience the latter approach often yields inferior accuracy due to poor balancing of the matrix $\hat{S}_{2n+1}$.

4. Numerical examples. The computations were carried out on an HP 9000 workstation using Matlab, i.e., with about 15 significant digits. We refer to the fast algorithm of QR-type for complex tridiagonal matrices described in Subsection 3.1 as "CSTQR." This algorithm is compared to the implementation of the QR algorithm for real Hessenberg matrices $\tilde{S}_{2n+1}$ furnished by Matlab (function eig). We refer to the latter algorithm as "HQR." The nodes and weights are determined by (2.9) and (2.26)-(2.27). Several of the quadrature rules listed in the tables have been discussed by Monegato [15]. In all examples the recursion coefficients a_j and b_j^2 for the orthogonal polynomials associated with the given measures dw are explicitly known; see, e.g., [19].

Example 4.1. We consider $(2n+1)$-point Gauss-Kronrod rules (1.4) associated with the Hermite measure $dw(x) := \pi^{-1/2}\exp(-x^2)dx$. Table 4.1 shows the number of pairs of complex conjugate weights with nonvanishing imaginary parts, as well as the number of real negative weights, of a few Gauss-Kronrod rules. The Gauss-Kronrod rules for $n = 2$ and $n = 3$ already have been considered in Example 2.2. The latter rule has one pair of complex conjugate nodes, each of which is associated

TABLE 4.1
Properties of Gauss-Kronrod rules for the Hermite measure

n	number of pairs of complex conjugate weights	number of real negative weights
3	0	2
4	0	2
5	2	0
10	2	0
25	10	0

TABLE 4.2
Errors in computed Gauss-Kronrod rules for the Hermite measure

n	$\max \kappa(G)$	discrepancy in Gauss nodes by CSTQR	discrepancy in Gauss nodes by HQR	discrepancy in nodes by HQR and CSTQR	discrepancy in weights by HQR and CSTQR
3	2.4	1.2E-15	6.7E-16	1.8E-15	3.1E-16
4	3.6	3.7E-15	2.6E-15	1.6E-15	6.8E-16
5	2.4E2	4.4E-14	8.9E-16	4.4E-14	2.1E-15
10	6.0E2	7.0E-13	4.0E-15	7.0E-13	7.6E-16
25	1.7E4	2.3E-12	2.2E-14	2.7E-10	2.2E-15

with a real negative weight. These weights are of the same magnitude. The other quadrature rules of Table 4.1 have the same number of pairs of complex conjugate nodes as they have pairs of complex conjugate weights with nonvanishing imaginary parts. For $n = 25$ all weights associated with nonreal nodes are of magnitude less than $5 \cdot 10^{-20}$.

Table 4.2 illustrates the accuracy achieved by the algorithms CSTQR and HQR for the quadrature rules of Table 4.1. In exact arithmetic the Gauss nodes are a subset of the Gauss-Kronrod nodes; cf. (1.6). We mark computed approximations of nodes with a prime or a double prime. Thus, we compute Gauss nodes $x'_1 < x'_2 < \ldots < x'_n$ as the eigenvalues of the leading real symmetric $n \times n$ principal submatrix of the Gauss-Kronrod matrix (2.6) using the Matlab function eig. Let $\{\tilde{x}'_k, \tilde{w}'_k\}_{k=1}^{2n+1}$ denote the set of node-weight pairs of the Gauss-Kronrod rule computed by algorithm CSTQR, and let $\tilde{x}'_1 < \tilde{x}'_2 < \ldots < \tilde{x}'_n$ be the subset of (approximations of) Gauss nodes; cf. (1.7). Similarly, let $\{\tilde{x}''_k, \tilde{w}''_k\}_{k=1}^{2n+1}$ denote the set of node-weight pairs of the Gauss-Kronrod rule computed by algorithm HQR. We order the pairs in both sets so that computed nodes and weights with the same index are approximations of the same (exact) node-weight pair.

A large number of plane transformations (3.2) are applied during the computations with algorithm CSTQR. When the matrix G has real entries only, it is a (unitary) Givens matrix, and therefore its condition number is one. Transformations (3.2) with not all entries real can have an arbitrarily large condition number $\kappa(G)$. The second column of Table 4.2 displays

$$\max \kappa(G), \tag{4.1}$$

where the maximum is taken over all plane transformations (3.2) applied in algorithm CSTQR. We use the notation 2.4E2 for $2.4 \cdot 10^2$.

The third column of Table 4.2 displays the discrepancies $\max_{1 \le k \le n} |x'_k - \tilde{x}'_k|$ and the fourth column shows the discrepancies $\max_{1 \le k \le n} |x'_k - \tilde{x}''_k|$. Under the assumption

TABLE 4.3
Properties of Gauss-Kronrod rules for the Laguerre measure

n	number of pairs of complex conjugate weights	number of real negative weights
2	1	0
3	1	0
10	5	0

TABLE 4.4
Errors in computed Gauss-Kronrod rules for the Laguerre measure

n	$\max \kappa(G)$	discrepancy in Gauss nodes by CSTQR	discrepancy in Gauss nodes by HQR	discrepancy in nodes by HQR and CSTQR	discrepancy in weights by HQR and CSTQR
2	1.0E2	3.3E-15	8.9E-16	9.6E-14	6.1E-16
3	3.1E1	3.6E-15	5.3E-15	6.5E-15	6.7E-16
10	2.6E4	3.8E-12	3.7E-14	3.9E-11	1.3E-12

that the error in all computed nodes is of about the same magnitude, these columns yield estimates of the magnitude of the error in all the nodes. These estimates were computed by Laurie [13] for Gauss-Kronrod rules with real nodes and positive weights.

The fifth and sixth columns tabulate the discrepancies $\max_{1\le k\le 2n+1}|\tilde{x}'_k-\tilde{x}''_k|$ and $\max_{1\le k\le 2n+1}|\tilde{w}'_k-\tilde{w}''_k|$, respectively.

The error estimates displayed in the table suggest that both algorithms CSTQR and HQR yield accuracy much higher than required in many applications. Generally, algorithm HQR gives higher accuracy. The quantity (4.1) is seen to give an indication of the error in the computed nodes and weights by algorithm CSTQR. For instance, when (4.1) is about $1\cdot 10^2$, then the nodes and weights are computed with an error of magnitude of about 10^{15-2}. □

Example 4.2. We consider $(2n+1)$-point Gauss-Kronrod rules (1.4) associated with the Laguerre measure (2.17). The Gauss-Kronrod rule for $n=3$ already has been considered in Example 2.3. Table 4.3 is analogous to Table 4.1. None of the tabulated rules have real negative weights. The complex conjugate weights for $n=10$ are all of magnitude less than $3\cdot 10^{-15}$. Table 4.4 is analogous to Table 4.2. □

Example 4.3. We consider $(2n+1)$-point Gauss-Kronrod rules (1.4) associated with the Jacobi measure

$$dw(x) := c_0(1-x)^\alpha(1+x)^\beta dx, \qquad -1<x<1, \qquad \alpha,\beta>-1,$$

where the scaling factor c_0 is chosen to make $\mu_0=1$. The Tables 4.5 and 4.6 are analogous to the Tables 4.1 and 4.2, respectively. □

5. Conclusion and extension. This paper describes two algorithms for the computation of Gauss-Kronrod quadrature rules with complex conjugate nodes and weights or with real nodes and positive and negative weights. In our experience both algorithms yield sufficient accuracy for many applications. The slower scheme HQR generally yields nodes and weights with higher accuracy.

We have assumed throughout this paper that the measure dw in (1.1) is nonnegative. However, the algorithms discussed may be applied also when the measure is indefinite; see Struble [18] for a discussion on orthogonal polynomials and quadrature rules for indefinite measures with support on the real axis.

TABLE 4.5
Properties of Gauss-Kronrod rules for the Jacobi measure

n	α	β	number of pairs of complex conjugate weights	number of real negative weights
15	3.5	3.5	0	3
25	3.5	3.5	0	10
5	7.5	7.5	0	2
25	7.5	7.5	12	0
10	0	5	4	1

TABLE 4.6
Errors in computed Gauss-Kronrod rules for the Jacobi measure

	max $\kappa(G)$	discrepancy in Gauss nodes by CSTQR	discrepancy in Gauss nodes by HQR	discrepancy in nodes by CSTQR and HQR	discrepancy in weights by CSTQR and HQR
$\alpha = 3.5$					
$\beta = 3.5$					
n =15	1.3E3	1.0E-13	1.3E-15	1.0E-13	2.7E-14
n =25	2.9E3	1.6E-12	4.3E-15	1.6E-12	4.7E-11
α =7.5					
β =7.5					
n =5	2.1E1	2.8E-15	3.5E-15	5.4E-15	2.0E-12
n =25	2.8E2	1.1E-13	5.3E-15	1.4E-13	1.5E-15
α =0					
β =5					
n =10	1.8E2	3.7E-14	2.1E-15	3.4E-14	1.0E-14

Acknowledgement. L.R. would like to thank Jane Cullum for discussions on the QL algorithm for complex symmetric matrices.

REFERENCES

[1] G. Ammar, W. Dayawansa and C. Martin, Exponential interpolation: theory and numerical algorithms, Appl. Math. Comput., 41 (1991), pp. 189–232.

[2] A. Bunse-Gerstner, An analysis of the HR-algorithm for computing the eigenvalues of a matrix, Linear Algebra Appl., 35 (1981), pp. 155–173.

[3] D. Calvetti, G. H. Golub. W. B. Gragg and L. Reichel, Computation of Gauss-Kronrod quadrature rules, Math. Comp., to appear.

[4] J. K. Cullum and R. A. Willoughby, A QL procedure for computing the eigenvalues of complex symmetric tridiagonal matrices, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 83–109.

[5] S. Ehrich and G. Mastroianni, Stieltjes polynomials and Lagrange interpolation, Math. Comp., 66 (1997), pp. 311–331.

[6] W. Gautschi, On generating orthogonal polynomials, SIAM J. Sci. Stat. Comput., 3 (1982), pp. 289–317.

[7] W. Gautschi, Gauss-Kronrod quadrature - a survey, in Numerical Methods and Approximation Theory III, ed. G. V. Milovanović, University of Niš, 1987, pp. 39–66.

[8] W. Gautschi, Orthogonal polynomials and quadrature, Elec. Trans. Numer. Anal., to appear.

[9] G. H. Golub and G. Meurant, Matrices, moments and quadrature, in Numerical Analysis 1993, eds. D. F. Griffiths and G. A. Watson, Longman, Essex, England, 1994, pp. 105–156.

[10] G. H. Golub and C. F. Van Loan, Matrix Computations, 3rd ed., Johns Hopkins University Press, Baltimore, 1996.

[11] G. H. Golub and J. H. Welsch, Calculation of Gauss quadrature rules, Math. Comp., 23 (1969),

pp. 221–230.

[12] W. B. Gragg, Matrix interpretations and applications of the continued fraction algorithm, Rocky Mountain J. Math., 4 (1974), pp. 213–225.

[13] D. P. Laurie, Calculation of Gauss-Kronrod quadrature rules, Math. Comp., 66 (1997), pp. 1133–1145.

[14] F. T. Luk and S. Qiao, Using complex-orthogonal transformations to diagonalize a complex symmetric matrix, in Advanced Signal Processing: Algorithms, Architectures, and Implementations VII, ed. F. T. Luk, Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE), vol. 3162, The International Society for Optical Engineering, Bellingham, WA, 1997, pp. 418–425.

[15] G. Monegato, A note on extended Gaussian quadrature rules, Math. Comp., 30 (1976), pp. 812–817.

[16] G. Monegato, Stieltjes polynomials and related quadrature rules, SIAM Rev., 24 (1982), pp. 137–158.

[17] L. Reichel, Construction of polynomials that are orthogonal with respect to a discrete bilinear form, Adv. Comput. Math., 1 (1993), pp. 241–258.

[18] G. Struble, Orthogonal polynomials: variable-signed weight functions, Numer. Math., 5 (1963), pp. 88–94.

[19] G. Szegő, Orthogonal Polynomials, 4th ed., Amer. Math. Society, Providence, 1975.

[20] D. S. Watkins, Fundamentals of Matrix Computations, Wiley, New York, 1991.