FACTA UNIVERSITATIS (NIŠ)
SER. MATH. INFORM. 15 (2000, 69–83

NOTE ON A CONSTRUCTION OF WEIGHTS IN GAUSS-TYPE QUADRATURE RULE

Gradimir V. Milovanović and Aleksandar S. Cvetković

Dedicated to Professor R. Ž. Djordjević for his 65th birthday

Abstract. It is well-known that a standard numerical construction of Gaussian quadrature rule is based on determining the eigenvalues and the first components of the eigenvectors of a symmetric tridiagonal matrix using QR-algorithm. In this note we analyze the accuracy of this procedure and give examples when such a construction gives inaccurate results for constructed weights. Also, a modified algorithm is proposed for construction of weights in Gaussian quadrature rule.

1. Introduction

Let $d\lambda(x)$ be a given nonnegative measure on the real line $\mathbb{R}$, with compact or unbounded support, for which all moments $\mu_k = \int_{\mathbb{R}} x^k d\lambda(x)$ $k = 0, 1, \ldots$, exist and are finite, and $\mu_0 > 0$. Then, there exists a unique set of orthonormal polynomials $\{p_k(x)\}_{k\in\mathbb{N}_0}$ with respect to the inner product

$$(f, g) = \int_{\mathbb{R}} f(x)g(x)\, d\lambda(x), \tag{1.1}$$

such that

$$\begin{aligned} p_k(x) &= a_k x^k + \text{lower degree terms} \qquad (a_k > 0), \\ (p_k, p_m) &= \delta_{km}, \end{aligned}$$

where δ_{km} is Kronecker's delta. These polynomials satisfy the recurrence relation

$$xp_k(x) = \sqrt{\beta_{k+1}}\, p_{k+1}(x) + \alpha_k p_k(x) + \sqrt{\beta_k}\, p_{k-1}(x), \qquad k \geq 0, \tag{1.2}$$

Received November 16, 1999.
1991 *Mathematics Subject Classification.* 65D30, 65D32.

with $p_{-1}(x) = 0$ and $p_0(x) = 1/\sqrt{\mu_0}$. With $\mathcal{P}_n$ we denote the set of all algebraic polynomials of degree at most n ($\in \mathbb{N}_0$).

It is well-known that the n-point Gaussian quadrature rule

$$\int_{\mathbb{R}} f(x)\,d\lambda(x) = \sum_{k=1}^{n} A_k^{(n)} f(x_k^{(n)}) + R_n(f) \tag{1.3}$$

is exact for each $f \in \mathcal{P}_{2n-1}$. The nodes $x_k^{(n)}$, $k = 1, \ldots, n$, are zeros of $p_n(x)$ and the Christoffel numbers

$$A_k^{(n)} = \frac{1}{p_n'(x_k^{(n)})} \int_{\mathbb{R}} \frac{p_n(x)}{x - x_k^{(n)}}\,d\lambda(x), \qquad k = 1, \ldots, n,$$

using the Christoffel-Darboux identity (cf. [1]) can be expressed in the form

$$A_k^{(n)} = \frac{1}{\sqrt{\beta_n}} \cdot \frac{1}{p_n'(x_k^{(n)}) p_{n-1}(x_k^{(n)})}, \qquad k = 1, \ldots, n, \tag{1.4}$$

or

$$A_k^{(n)} = \frac{1}{\sum\limits_{\nu=0}^{n-1} \left(p_\nu(x_k^{(n)})\right)^2}, \qquad k = 1, \ldots, n. \tag{1.5}$$

The last formula due to Shohat [13].

An important progress in the construction of Gaussian formulas was made by Golub and Welsch [8] in 1969, giving a numerical algorithm which is based on determining the eigenvalues and the first components of the eigenvectors of a symmetric tridiagonal matrix. Namely, taking the first n equations from (1.2), they obtained

$$J_n \mathbf{p}_k^{(n)} = x \mathbf{p}_k^{(n)} + \sqrt{\beta_n}\, p_n(x) \mathbf{e}_n, \tag{1.6}$$

where

$$J_n = \begin{bmatrix} \alpha_0 & \sqrt{\beta_1} & & & \mathrm{O} \\ \sqrt{\beta_1} & \alpha_1 & \sqrt{\beta_2} & & \\ & \sqrt{\beta_2} & \alpha_2 & \ddots & \\ & & \ddots & \ddots & \sqrt{\beta_{n-1}} \\ \mathrm{O} & & & \sqrt{\beta_{n-1}} & \alpha_{n-1} \end{bmatrix}, \quad \mathbf{p}_k^{(n)} = \begin{bmatrix} p_0(x) \\ p_1(x) \\ p_2(x) \\ \vdots \\ p_{n-2}(x) \\ p_{n-1}(x) \end{bmatrix},$$

and $\mathbf{e}_n$ is the last coordinate vector, $\mathbf{e}_n = [0\ 0\ \ldots\ 1]^T$. The matrix J_n is known as the Jacobi matrix. Putting the zero $x_k^{(n)}$ $(k = 1, \ldots, n)$ of the polynomial $p_n(x)$ in (1.6), instead of x, it is clear that $x_k^{(n)}$ is the eigenvalue and $\mathbf{p}_k^{(n)}$ is the corresponding eigenvector of the Jacobi matrix J_n, so that (1.5) can be expressed in the form $A_k^{(n)} \|\mathbf{p}_k^{(n)}\|_E^2 = 1$ $(k = 1, \ldots, n)$, where $\|\mathbf{a}\|_E^2 = \mathbf{a}^T\mathbf{a}$ $(\mathbf{a} \in \mathbb{R}^n)$. After a normalization of eigenvectors of the Jacobi matrix,

$$\frac{\mathbf{p}_k^{(n)}}{\|\mathbf{p}_k^{(n)}\|_E} =: \mathbf{q}_k^{(n)} = [q_{k1}^{(n)}\ q_{k2}^{(n)}\ \ldots\ q_{kn}^{(n)}]^T,$$

and a fact that $\|\mathbf{p}_k^{(n)}\|_E = \|\mathbf{p}_k^{(n)}\|_E / \|\mathbf{q}_k^{(n)}\|_E = p_0(x_k^{(n)})/q_{k1}^{(n)} = (1/\sqrt{\mu_0})/q_{k1}^{(n)}$, the Christoffel numbers become

$$A_k^{(n)} = \mu_0 (q_{k1}^{(n)})^2 \qquad (k = 1, \ldots, n). \tag{1.7}$$

Finally, simplifying QR algorithm so that only the first components of the eigenvectors are computed, Golub and Welsch [8] gave an efficient procedure for construction of Gaussian quadrature rules. This procedure is implemented in several programming package including the most known `ORTHPOL` given by Gautschi [6].

Since, the Jacobi matrix in (1.6) is symmetric and tridiagonal, QR-algorithm can be used for construction of weights. It is known that in the case of the positive definite orthogonal polynomials, i.e., when the coefficients β_k are positive, QR-algorithm can be used successfully for a construction of nodes (see [9]). However, in the case when some of coefficients β_k are negative, QR-algorithm may become ill-conditioned.

In this note we give our observations on numerical behavior of this algorithm in some cases (Section 2), propose a modification of the algorithm (Section 3), and finally we give a few interesting numerical examples.

2. Some Observations on Numerical Behavior of the Algorithm Based on (1.7)

The ill-conditioning of QR-algorithm for finding eigenvalues of the Jacobi matrix was studied in [12] and presented for the class of the generalized Bessel polynomials, which coefficients in the three-term recurrence relation are given by

$$\alpha_k \ = \ -\frac{2\alpha}{(\alpha + 2k)(a + 2 + 2k)}, \qquad k \in \mathbb{N}_0,$$

$$\beta_k = -\frac{4k(\alpha+k)}{(\alpha+2k)^2((\alpha+2k)^2-1)}, \quad k \in \mathbb{N},$$

and

$$\beta_0 = -\frac{4\pi \mathrm{i}}{1+\alpha}.$$

As it can be seen all sub-diagonal entries in the Jacobi matrix for the generalized Bessel polynomials are purely imaginary numbers. It is also believed that for all definite classes of polynomials, a case when some of the coefficients β_k are negative, QR-algorithm exhibits ill-conditioning in the process of the eigenvalues construction. In our numerical experiments this was not proved. For example, for a class of polynomials orthogonal on the semicircle (see [7]) QR-algorithm can be used for constructing the eigenvalues of the Jacobi matrix and such construction is not ill-conditioned. For some other definite classes of orthogonal polynomials, QR-algorithm also does not exhibit ill-conditioning.

The problem of ill-conditioning of QR-algorithm in a construction of eigenvalues, can be resolved only by using higher precision arithmetics. Especially, for the class of the generalized Bessel polynomials, the construction can be performed by an algorithm proposed in [12]. This algorithm uses Newton-Kantorvič method with certain good starting values applied on the second order differential equation which is satisfied by the generalized Bessel polynomial. It is clear that in the case of the generalized Bessel polynomials, QR-algorithm cannot be used for a construction of related Gaussian quadrature rules, because the construction of eigenvalues is ill-conditioned.

It is also easy to check that the algorithm for weights based on (1.7) cannot be used for construction of Gaussian quadrature rule even in cases when a class of orthogonal polynomials is positive definite. Running QR-algorithm for a construction of Gaussian quadrature rule for the generalized Hermite weight function $w(x) = |x|^{2\mu}e^{-x^2}$ on $\mathbb{R}$, with the parameter $\mu = 10$ and $n = 100$, introduces errors in the weight coefficients. If the weights are constructed using eigenvectors of the Jacobi matrix, in the fashion proposed in [8], the maximal relative error in constructed weights is of the order 10^{-3}. If we decrease parameter μ, this error in the weights is encountered for bigger degree of polynomials.

Similar behavior is exhibited for the generalized Gegenbauer polynomials, orthogonal with respect to the weight function $w(x) = |x|^{2\alpha+1}(1-x^2)^{\beta}$ on $(-1, 1)$. Increasing the parameter α, when β is fixed, increases relative errors in the weights calculated using eigenvectors of the Jacobi matrix.

It is important to indicate that algorithm for the construction of Gaussian quadrature rule using eigenvalues and eigenvectors, proposed in [8], does not control errors in the constructed weights. Because of this fact, the eigenvalues can be constructed with the machine precision, but weights can be constructed with a different error.

3. A Modified Algorithm

In this section we describe a modification of the previous algorithm for construction of weights which can be applied in all cases when the calculation of orthogonal polynomials using three term recurrence relation is stable. In this modification, the nodes $x_k^{(n)}$, $k = 1, \ldots, n$, are calculated in a usual way using QR-algorithm, but for the calculation of the weight coefficients $A_k^{(n)}$ we apply the Shohat's formula (1.5). Notice that such a way was also used in the period before an application of QR-procedure (cf. Gautschi [3]). Thus, we use here complete eigenvectors, but not only their first components.

Since, the numerical stability of calculation of orthogonal polynomials using the three term recurrence relation is crucial, we gave some preliminary results on general theory of difference equations. In [2, 4] the following definition can be found.

Definition 2.1. Let f_k and g_k be two linearly independent solutions of the recurrence relation (difference equation)

$$y_{k+1} + a_k y_k + b_k y_{k-1} = 0, \quad k \in \mathbb{N}_0. \tag{3.1}$$

For the solution f_k of (3.1) is said to be *minimal*, and for g_k to be *dominant* if

$$\lim_{k \to +\infty} \frac{f_k}{g_k} = 0.$$

It is known that dominant solutions can be calculated numerically stable directly from (3.1) (see for example [2, 4, 9]). However, the minimal solution cannot be calculated numerically stable using equation (3.1) straightforward. But it can be calculated using the same equation backwards.

If we start from some index m, and if we put $z_k = y_{k+1}/y_k$, then we have following equalities

$$z_{k-1} = -\frac{b_k}{a_k + z_k}, \qquad k = m-1, \ldots, 1. \tag{3.2}$$

If we know the value z_{m-1} then we can calculate numerically stable other values z_k, $k = m-2, \ldots, 1, 0$. In that case, the minimal solution can be expressed by the following equalities

$$f_{k-1} = \frac{f_k}{z_{k-1}}, \quad k = m, \ldots, 1.$$

This procedure can be found in [2, 4, 9].

Thus, this consideration shows that we can calculate both solutions (dominant and minimal) of a second order difference equation in a numerically stable way.

In order to calculate the weights by (1.5) we need the values of the sequence $p_\nu(x_k^{(n)})$, $\nu = 0, 1, \ldots, n-1$, where $x_k^{(n)}$, $k = 1, \ldots, n$, are zeros of the polynomial p_n. If orthogonal polynomials are dominant solution of the difference equation (1.2), then the calculations can be made straightforward.

When sequence $p_\nu(x_k^{(n)})$, $\nu = 0, 1, \ldots, n-1$, has been calculated we can evaluate $p_n(x_k^{(n)})$ also. Of course, this value is zero, but this gives us an opportunity to measure the error propagation in calculations. Let *eps* be the corresponding machine precision in computer arithmetics used in calculations. ¿From (1.2), which can be rewritten in the form

$$p_n(x_k^{(n)}) = \frac{(x_k^{(n)} - \alpha_{n-1})p_{n-1}(x_k^{(n)}) - \sqrt{\beta_{n-1}}p_{n-2}(x_k^{(n)})}{\sqrt{\beta_n}}, \tag{3.3}$$

it can be seen that the quantity

$$(x_k^{(n)} - \alpha_{n-1})p_{n-1}(x_k^{(n)}) - \sqrt{\beta_{n-1}}p_{n-2}(x_k^{(n)}), \tag{3.4}$$

has to be *eps* times smaller then any included summand. If this is not true, then the calculation suffers from ill-conditioning and the expression

$$\log_{10}\left|\frac{(x_k^{(n)} - \alpha_{n-1})p_{n-1}(x_k^{(n)}) - \sqrt{\beta_{n-1}}\,p_{n-2}(x_k^{(n)})}{\sqrt{\beta_{n-1}}\,p_{n-2}(x_k^{(n)})}\right| - \log_{10} eps \tag{3.5}$$

gives the number of decimal digits lost due to ill-conditioning of the three term recurrence relation.

In the case when sequence of orthogonal polynomials is the minimal solution of (1.2), then calculations are performed backwards. Since $p_n(x_k^{(n)})$ is zero, we have $z_{n-1} = 0$, which enables a construction over the system of

equations (3.2). To control the error propagation in calculations we can use the fact that

$$z_0 = \frac{x_k^{(n)} - \alpha_0}{\sqrt{\beta_1}}.$$

Then the number of decimal digits lost due to ill-conditioning can be measured by the following expression

$$\log_{10} \left| \frac{z_0\sqrt{\beta_1} - x_k^{(n)} + \alpha_0}{x_k^{(n)} - \alpha_0} \right| - \log_{10} eps. \tag{3.6}$$

Using the criterions (3.5) and (3.6) we can decide about the way how to calculate the weights. It is important to indicate here that the three term recurrence relation is not always stable to calculate the values of related orthogonal polynomials. If we deal with a positive measure which supporting set has isolated points or it consists only of isolated points, then the calculation of related orthogonal polynomials near the isolated point is not numerically stable if the three term recurrence relation is used straightforward. In other words, near isolated points of the support, the sequence of orthogonal polynomials is not dominant solution for the three term recurrence relation, it is rather minimal. This is understandable if we recall that every isolated point of the supporting set has near itself at least one zero of the orthogonal polynomials when degree of polynomial is big enough (see [5]).

For example, using a straightforward calculation, given with relation (1.2) for the generalized Charlier polynomials, which are supported on the set of nonnegative integers (cf. [1]), will give an inaccurate result if calculations are performed near any nonnegative integer and when degree of the polynomial is large enough. This is a consequence of the fact that the zeros of Charlier polynomials are distributed near nonnegative integers if degree of polynomial is big enough.

If there is an isolated point of the supporting set, the calculations of values of orthogonal polynomials near that point should be performed backwards using equations (3.2). For such points, the values of orthogonal polynomials are minimal solution.

But if the supporting set for the measure has no isolated points all calculations can be performed using straightforward relation (1.2). For any reasonable number of points, the concentration of zeros of orthogonal polynomials cannot be that dense that solution of the three term recurrence relation becomes minimal. For example, calculations for Jacobi, generalized

Gegenbauer, generalized Hermite, generalized Laguerre, Lindelöf, Abel and logistic polynomials can be performed safely using relation (1.2).

It is interesting to mention that in the case of the generalized Bessel polynomials, the calculation cannot be done straightforward. This means that at zeros of the generalized Bessel polynomial of degree n, the generalized Bessel polynomials are minimal solution of the corresponding three term recurrence relation. This fact can be easily understand if we know that the zeros of the generalized Bessel polynomials are grouped near number zero in the complex plane. Almost all zeros of the generalized Bessel polynomial of degree n, for n large enough, are arbitrarily close to zero. Since QR-algorithm cannot be applied even for a construction of zeros for the generalized Bessel polynomials, the construction of Gaussian quadrature rules involving the generalized Bessel polynomials cannot be done using QR-algorithm. However, it can be constructed using algorithm proposed in [12] with construction of weights using the three term recurrence relation backwards (3.2).

In the case of positive-definite orthogonal polynomials, all quantities $p_\nu^2(x_k^{(n)})$, $\nu = 0, 1, \ldots, n-1$, are positive and, therefore, the summation in (1.5) should be performed starting from the smallest terms, (see for example [9]). Otherwise, if the sequence of orthogonal polynomials is not positive definite, summation can be made arbitrarily.

In numerical experiments have been conducted it was enough to perform summation as it is given in (1.5) for all classes of orthogonal polynomials of interest.

In order to evaluate weights using (1.4) we need the values $p_{n-1}(x_k^{(n)})$, $k = 1, \ldots, n$. These values are not critical and their evaluations can be performed either using recurrence straightforward or backwards. However, an evaluation of the derivative p_n' at the points $x_k^{(n)}$, $k = 1, \ldots, n$, can be numerically unstable. Since zeros of the polynomial p_n are known (i.e. the points $x_k^{(n)}$, $k = 1, \ldots, n$), a calculation of derivative at the points $x_k^{(n)}$, $k = 1, \ldots, n$, is easy. But if the zeros are dense in an interval, which is a case when the sequence of orthogonal polynomials is positive definite (i.e., when polynomials are orthogonal with respect to positive measure on the finite interval), the calculation of the derivative is numerically unstable. The derivative can be calculated using the following expression

$$p_n'(x_k^{(n)}) = \frac{1}{\sqrt{\beta_0 \cdots \beta_n}} \prod_{\nu=0,\, \nu \neq k}^{n} (x_k^{(n)} - x_\nu^n). \tag{3.7}$$

It is clear that if zeros are closely distributed, a calculation using this equa-

tion is numerically unstable. An asymptotic distribution of zeros of Legendre polynomials is given in [14]. It can be concluded that for polynomial of degree 22, at least two zeros have the same first significant digit. Subtracting these zeros makes decreasing of one significant digit in the result. Of course, if zeros are not so densely distributed then equation (1.4) can be used for the construction, where the derivative is calculated using (3.7).

Remark 1. There is still another way for the calculation of derivatives. Differentiating (1.2) we get

$$p_n(x) + xp'_n(x) = \sqrt{\beta_{n+1}}p'_{n+1}(x) + \alpha_n p'_n(x) + \sqrt{\beta_n}p'_{n-1}(x).$$

If we replace $p'_n(x)$ by $q_n(x)$, this equation can be qualified as a three term recurrence equation of a non-homogeneous type

$$p_n(x) + xq_n(x) = \sqrt{\beta_{n+1}}q_{n+1}(x) + \alpha_n q_n(x) + \sqrt{\beta_n}q_{n-1}(x), \tag{3.8}$$

where the starting values are given by $q_1 = 1/\sqrt{\beta_1\beta_0}$ and $q_0 = 0$. Numerical stability of this equation is governed by the non-homogeneous term. It would be interesting to make some experiments with this equation and to find some numerical stability properties of this equation.

The solution of the equation (3.8) can be given in the following form

$$q_n(x) = -\sum_{\nu=0}^{n-1} N^{\nu+1}(p_n(x))p_\nu(x) + C_1 p_n(x) + C_2 N(p_n(x)), \tag{3.9}$$

where C_1 and C_2 are arbitrary constants (in the case of derivatives $C_1 = C_2 = 0$) and N is the numerator operator, which acts on the sequence of orthogonal polynomials producing another sequence of orthogonal polynomials which satisfy the following recurrence relation

$$xN^\nu(p_n(x)) = \sqrt{\beta_{n+1}}N^\nu(p_{n+1}(x)) + \alpha_n N^\nu(p_n(x))\sqrt{\beta_n}N^\nu(p_{n-1}(x)),$$

with initial conditions given by

$$N^\nu(p_\nu(x)) = -\frac{1}{\sqrt{\beta_\nu}}, \quad N^\nu(p_k) = 0, \quad k < \nu.$$

4. Numerical Examples

The generalized Hermite polynomials satisfy the following recurrence relation

$$xp_n(x) = \sqrt{\beta_{n+1}}p_{n+1}(x) + \sqrt{\beta_n}p_{n-1}(x), \tag{4.1}$$

where

$$\beta_n = \begin{cases} k, & \text{if } n = 2k, \\ (2k+1+2\mu)/2, & \text{if } n = 2k+1. \end{cases} \tag{4.2}$$

They are orthogonal with respect to the measure $d\lambda(x) = |x|^{2\mu} e^{-x^2}\, dx$ on the interval $(-\infty, +\infty)$.

Table 4.1: Maximal relative errors (max err) in weights calculated by (1.7) for the generalized Hermite measure for some selected values of μ and n

μ	5	10	15	20	25
n	100	80	60	40	30
max err	1.5(−10)	2.5(−5)	9.0(−4)	5.3(−6)	3.8(−9)

The maximal relative errors (max err) in the weights $\widetilde{A}_k^{(n)}$, calculated by the standard formula (1.7), are given in Table 4.1. (Numbers in parentheses denote decimal exponents.)

As we can see, the maximal relative error increases if the parameter of the family (μ) and the degree of orthogonal polynomials are increased.

Table 4.2: Nodes $x_k^{(n)}$ and weights $\widetilde{A}_k^{(n)}$ and $A_k^{(n)}$, calculated by (1.7) and (1.5), respectively, for the generalized Hermite weight with $\mu = 25$ and $n = 50$

k	$x_k^{(n)}$	$\widetilde{A}_k^{(n)}$	$A_k^{(n)}$
⋮			
45	8.87048112408543	5.83527133818019(13)	5.83527133818503(13)
46	9.27131769434522	3.89993136521226(11)	3.89993135706594(11)
47	9.69733297167313	1.22170372571917(9)	1.22170373445053(9)
48	1.01587551407497(1)	1.43752516649416(6)	1.43752457423890(6)
49	1.06753984703247(1)	4.16975520190378(2)	4.16971067128949(2)
50	1.12991277513983(1)	1.05084478232937(−2)	1.03057175800162(−2)

We also give the weights constructed with both algorithms for the generalized Hermite measure, with a parameter of the family $\mu = 25$, when the polynomial degree $n = 50$. The weights obtained by (1.7) and using the modified algorithm are denoted by $\widetilde{A}_k^{(n)}$ and $A_k^{(n)}$, respectively. The nodes $x_k^{(n)}$ and corresponding weights are given in Table 4.2. A distribution of errors in the weights with respect to the position of nodes in the support of the measure is very interesting. The weights associated with the biggest

nodes (in modulus) are given with the smallest precision. There is also a small precision for nodes which absolute values are closest to zero.

The generalized Gegenbauer polynomials satisfy the following three term recurrence equation

$$xp_n(x) = \sqrt{\beta_{n+1}}p_{n+1}(x) + \sqrt{\beta_n}p_{n-1}(x), \tag{4.3}$$

where

$$\beta_n = \begin{cases} \dfrac{(k+1+\alpha)(k+1+\alpha+\beta)}{(2k+1+\alpha+\beta)(2k+2+\alpha+\beta)}, & \text{if } n = 2k+1, \\ \dfrac{k(k+\beta)}{(2k+\alpha+\beta)(2k+1+\alpha+\beta)}, & \text{if } n = 2k. \end{cases} \tag{4.4}$$

These polynomials are orthogonal with respect to the measure $d\lambda(x) = |x|^{1+2\alpha}(1-x^2)^{\beta}dx$ supported on $[-1,1]$.

The maximal relative errors in the Gaussian weights for the generalized Gegenbauer weight function, with parameters $\beta = 10$ and $\alpha = 0(5)30$, constructed by using (1.7) with $n = 100$ nodes, are given in Table 4.3.

Table 4.3: Maximal relative errors in the weights $\widetilde{A}_k^{(n)}$ $(n = 100)$ for the generalized Gegenbauer polynomials, with parameters $\beta = 10$ and $\alpha = 0(5)30$

α	0	5	10	15	20	25	30
max err	9.0(−7)	6.9(−9)	3.6(−9)	2.5(−7)	3.4(−5)	1.1(−3)	8.0(−1)

As we can see, the relative error in the weights $\widetilde{A}_k^{(n)}$, constructed by (1.7), increases when the parameter α increases. The same effect appears as n increases and parameters of the family are fixed.

The nodes $x_k^{(n)}$ $(k = 46, \ldots, 55)$ and corresponding weights are given in Table 4.4 for 100-point Gaussian quadrature rule with respect to the generalized Gegenbauer measure with $\alpha = 30$ and $\beta = 10$. The biggest relative error is achieved for nodes which are closest to the number zero. There are also relative errors in weights associated with other nodes, but these relative errors are smaller. Unlike example for the generalized Hermite measure, in this example, the errors in constructed weights appear symmetrically.

Similar behavior in constructed weights can be demonstrated for the generalized Laguerre weight $w(x) = x^s e^{-x}$ $(s > -1)$ on $(0, +\infty)$, also. The sequence of the generalized Laguerre polynomials satisfies the following three

Table 4.4: Nodes $x_k^{(n)}$ and weights $\widetilde{A}_k^{(n)}$ and $A_k^{(n)}$, calculated by (1.7) and (1.5), respectively, for the generalized Gegenbauer weight ($\alpha = 30$, $\beta = 10$) and $n = 100$

k	$x_k^{(n)}$	$\widetilde{A}_k^{(n)}$	$A_k^{(n)}$
⋮			
46	−0.373269526254518	4.40712931244721(−29)	4.40712947393128(−29)
47	−0.347403738539906	7.18119599285471(−31)	7.18119847637293(−31)
48	−0.320010581648552	6.29195033882866(−33)	6.29196900506548(−33)
49	−0.290248701077263	2.21111291156777(−35)	2.21117188982564(−35)
50	−0.255811616198617	1.50414218779289(−38)	1.50459116124500(−38)
51	0.255811616198618	1.51643281414687(−38)	1.50459116124529(−38)
52	0.290248701077264	2.21070054842355(−35)	2.21117188982596(−35)
53	0.320010581648553	6.29201139272186(−33)	6.29196900506574(−33)
54	0.347403738539905	7.18120162277844(−31)	7.18119847637186(−31)
55	0.373269526254518	4.40712893209631(−29)	4.40712947393113(−29)
⋮			

term recurrence relation

$$xp_n(x) = \sqrt{(n+1)(n+1+s)}\, p_{n+1}(x) + (2n+1+s)p_n(x) + \sqrt{n(n+s)}\, p_{n-1}(x).$$

Table 4.5 shows the maximal relative errors in the weights $\widetilde{A}_k^{(n)}$ for the generalized Laguerre weight function. If we further increase the number of nodes in the corresponding Gaussian quadrature, the maximal relative error in weights becomes bigger for the same value of s. However, it seems that for any reasonable number of nodes (e.g. smaller then 300), the maximal relative error in the weights (for $s = 0$) is approximatively constant and its value is of order 10^{-13}.

Table 4.5: Maximal relative error in weights $\widetilde{A}_k^{(n)}$ for the generalized Laguerre measure in 150-point Gaussian quadrature rule

ν	0	10	20	30	40	50	60
max err	1.9(−12)	4.6(−11)	1.4(−7)	6.9(−5)	1.3(−4)	8.3(−3)	1.8(−1)

The problem in the precision of constructed weights can also be found when a QR-modification of some positive measure is performed. As we know, QR-modifications of some measure $d\lambda(x)$ can always be represented in the

Table 4.6: Nodes, weights with errors, and the exact weights for QR-modified Legendre measure given by (4.6)

k	$x_k^{(n)}$	$\widetilde{A}_k^{(n)}$	$A_k^{(n)}$
⋮			
60	0.107812380640346	2.09914865872597(−18)	2.09914865837238(−18)
61	0.138710282843229	2.13892724571921(−20)	2.13892724438224(−20)
62	0.219458604066318	2.21698613750904(−25)	2.21698643737794(−25)
63	0.329221328136957	1.24201083492062(−27)	1.24200433447655(−27)
64	0.432757097507551	7.27768189481662(−27)	7.27762869886060(−27)
65	0.496125821491815	3.78302684067583(−25)	3.78302470868232(−25)
66	0.554326542758999	5.07429534068515(−21)	5.07429532520943(−21)
⋮			

following way

$$d\widetilde{\lambda}(x) = \prod_{i=1}^{n}(x - x_i)^{2k_i} d\lambda(x), \tag{4.5}$$

where the points x_i are inside the support of the measure $d\lambda(x)$ and k_i are nonnegative integers. The total degree of a modification is the sum of all numbers k_i. For almost all measures, which are interesting in applications, the algorithm for weights based on (1.7) returns inaccurate values for constructed weights, if the total degree of modification is big enough. For example, if we perform QR-modifications on the Legendre measure in this way

$$((x - 1/2)(x - 2/5)(x - 3/10)(x - 1/5))^6\, dx, \tag{4.6}$$

then the maximal relative error in weights is going to be of the magnitude 10^{-6}. Table 4.6 shows nodes $x_k^{(n)}$, weights with errors $\widetilde{A}_k^{(n)}$, and the exact weights $A_k^{(n)}$ $(60 \le k \le 66)$ for the previous measure, when the number of nodes in the quadrature rule is $n = 100$.

If we increase the number of QR-modifications, the maximal relative error is going to be increased also. For example, if we take the measure to be

$$((x - 1/2)(x - 2/5)(x - 3/10)(x - 1/5))^{12}\, dx,$$

the magnitude of corresponding maximal relative error in the weights becomes 10^{-5}, when $n = 100$.

A very similar behavior can be encountered with other different types of measures (e.g. generalized Hermite measure, generalized Laguerre measure, Jacobi measure, generalized Gegenabuer measure).

Table 4.7: Zeros x_ν ($\nu = 1, 2, 3, 4$) of the s-orthogonal generalized Hermite polynomial of degree four, obtained by the standard and modified method

ν	standard method	modified method
1	$-7.28962206\ldots$	-7.289621792645020
2	$-3.66640712\ldots$	-3.666407011304882
3	$3.66640695\ldots$	3.666407011304883
4	$7.28962175\ldots$	7.289621792645021

The problem with constructed weights, using algorithm based on (1.7), originated in an implementation of an algorithm proposed in [10]. This algorithm uses Gaussian quadrature rule to approximate integrals which then simplifies the computation of the solution for certain system of nonlinear equations characterizing s-orthogonal polynomials. Using Gaussian quadrature rule constructed by algorithm based on (1.7), returned zeros of s-orthogonal polynomials have severely damaged precision. As an illustration of this fact, we give zeros of a s-orthogonal polynomial of degree 4 for $s = 10$, when the orthogonality is taken with respect to the generalized Hermite measure with $\mu = 15$. The corresponding zeros are given in Table 4.7. A similar behavior is exhibited for smaller values of μ, but then the degree of a s-orthogonal polynomial has to be bigger. It can be observed that zeros of the s-orthogonal polynomial are not even symmetric if a calculation is performed using weights constructed with the algorithm based on (1.7).

REFERENCES

1. T.S. Chihara: *An Introduction to Orthogonal Polynomials*. Gordon and Breach, New York, 1978.
2. W. Gautschi: *Computational aspects of three-term recurrence relations*. SIAM Review **9** (1967), 24–82.
3. W. Gautschi: *Construction of Gauss-Christoffel quadrature formulas*. Math. Comp. **22** (1968), 251–270.
4. W. Gautschi: *Minimal solutions of three-term recurrence relations and orthogonal polynomials*. Math. Comp. **36** (1981), 337–342.
5. W. Gautschi: *Is the recurrence relation for orthogonal polynomials always stable*. BIT **33** (1993), 277–284.

6. W. Gautschi: *Algorithm* 726*: ORTHPOL – A package of routines for generating orthogonal polynomials and Gauss-type quadrature rules*. ACM Trans. Math. Software **10** (1994), 21–62.
7. W. Gautschi and G.V. Milovanović: *Polynomials orthogonal on the semi-circle*. J. Approx. Theory **46** (1986), 230–250.
8. G.H. Golub and J.H. Welsch: *Calculation of Gauss quadrature rule*. Math. Comput. **23** (1986), 221–230.
9. G.V. Milovanović: *Numerical Analysis*. Naučna knjiga, Beograd, 1991 (Serbian).
10. G.V. Milovanović and M.M. Spalević: *Quadrature formulae connected to σ-orthogonal polynomials*. J. Comput. Appl. Math. (to appear).
11. G.V. Milovanović, M.M. Spalević, and A.S. Cvetković: *Calculation of Gaussian quadratures with multiple nodes* (in preparation).
12. L. Pasquini: *Accurate computation of the zeroes of the generalized Bessel polynomials*. Numer. Math., **86** (2000), 507–538.
13. J.A. Shohat: *On a certain formula of mechanical quadratures with non-equidistant ordinates*. Trans. Amer. Math. Soc. **31** (1929), 448–463.
14. G. Szegő: *Orthogonal Polynomials*. Amer. Math. Soc. Colloq. Publ., Vol. **23**, 4th ed., Amer. Math. Soc., Providence, R.I., 1975.

Faculty of Electronic Engineering
Department of Mathematics
P.O. Box 73
18000 Niš, Serbia – Yugoslavia