

ESTIMATING THE ERROR OF GAUSS-TURÁN QUADRATURE FORMULAS USING THEIR EXTENSIONS*

ALEKSANDAR S. CVETKOVIĆ[†] AND MIODRAG M. SPALEVIĆ[†]

Abstract. We consider extensions of Kronrod-type and extensions obtained by generalized averaged Gaussian quadrature formulas for Gauss-Turán quadrature formulas. Existence and uniqueness of these extensions are considered. Their numerical construction is proposed. It is the first general method and is based on a combination of well-known numerical methods for Gauss-Turán, Gauss, Gauss-Kronrod, Anti-Gauss, and generalized averaged Gaussian quadratures. We employ these extensions for estimating the remainder terms in the Gauss-Turán quadratures. Numerical results are presented.

Key words. quadrature rule, error estimate

AMS subject classifications. 65D32, 65D30

1. Introduction. Let $d\lambda$ be a given nonnegative measure on the real line \mathbb{R} with compact or unbounded support for which all moments $\mu_k = \int_{\mathbb{R}} t^k d\lambda(t)$ ($k = 0, 1, \dots$) exist and are finite with $\mu_0 > 0$. If λ is an absolutely continuous function, then $d\lambda(t) = \omega(t) dt$, where ω is a given nonnegative and integrable weight function over the smallest interval $[a, b]$ which contains the support of λ . Even though our results in this paper hold for any λ defined as above, we will present them for the most common case when $d\lambda(t) = \omega(t) dt$. As usual, for $k \in \mathbb{N}_0$, let \mathbb{P}_k denote the set of polynomials of degree at most k .

In 1950, P. Turán [40] proposed an interpolatory quadrature formula of the type

$$(1.1) \quad \int_{-1}^1 f(t) dt \approx \sum_{\nu=1}^n \sum_{i=0}^{2s} A_{i,\nu} f^{(i)}(\tau_\nu) \quad (s \in \mathbb{N}_0),$$

which has the highest possible algebraic degree of precision (ADP). In this paper we consider a generalization of formula (1.1) by including a weight function, i.e.,

$$(1.2) \quad \int_a^b f(t)\omega(t) dt \approx \sum_{\nu=1}^n \sum_{i=0}^{2s} A_{i,\nu} f^{(i)}(\tau_\nu) \quad (s \in \mathbb{N}_0).$$

In the Gauss-Turán quadrature formula (1.2), τ_ν are the zeros of a polynomial π_n of degree n known as an s -orthogonal polynomial, which satisfies the orthogonality relation

$$\int_a^b \pi_n^{2s+1}(t)p(t)\omega(t) dt = 0, \quad \text{for all } p \in \mathbb{P}_{n-1},$$

and the $A_{i,\nu}$ are determined through interpolation. If τ_ν and $A_{i,\nu}$ are chosen in this way, the ADP for (1.2) is $2(s+1)n - 1$. The coefficients $A_{i,\nu}$ in the Gauss-Turán quadrature formula (1.2), however, are not all positive in general. In the case $s = 0$, formula (1.2) reduces to well-known Gaussian quadrature.

Numerically stable methods for constructing nodes τ_ν and coefficients $A_{i,\nu}$ in Gauss-Turán quadrature formulas with multiple nodes and their generalizations can be found in

*Received July 19, 2013. Accepted August 28, 2013. Published online on February 17, 2014. Recommended by L. Reichel. This work was supported in part by the Serbian Ministry of Education, Science and Technological Development (Research Projects: "Approximation of integral and differential operators and applications" (#174015) and "Methods of numerical and nonlinear analysis with applications" (#174002)).

[†]University of Belgrade, Faculty of Mechanical Engineering, Kraljice Marije 16, 11120 Belgrade 35, Serbia ([acvetkovic, mspalevic}@mas.bg.ac.rs](mailto:{acvetkovic, mspalevic}@mas.bg.ac.rs)).

[15, 18, 28, 37] (see also [13]). For the asymptotic representation of the coefficients $A_{i,\nu}$ see [31]. Some interesting results concerning this quadrature formula and its applications can be found in [25, 36] and the references therein, as well as in [16, 21, 31].

The estimation of the error in a quadrature formula is an important problem. The purpose of this paper is to consider a Kronrod extension or an extension obtained using generalized averaged Gaussian quadrature formulas for Gauss-Turán quadrature formulas. These extensions can be applied to estimate the error in the original Gauss-Turán quadrature. A numerical construction of these extensions for Gauss-Turán quadrature formulas is proposed in this article. It is the first general method, and it is based on the combination of well-known numerical procedures for Gauss-Turán, Gauss, Gauss-Kronrod, Anti-Gauss, and generalized averaged Gaussian quadratures.

2. A Kronrod extension of Gauss-Turán quadrature formula. Following Kronrod's idea, Li [24] considered an extension of (1.2) of the form

$$(2.1) \quad \int_a^b f(t)\omega(t) dt \approx \sum_{\nu=1}^n \sum_{i=0}^{2s} B_{i,\nu} f^{(i)}(\tau_\nu) + \sum_{j=1}^{n+1} C_j f(\hat{\tau}_j) \quad (s \in \mathbb{N}_0),$$

where the τ_ν are the same nodes as in (1.2), and the new nodes $\hat{\tau}_j$ and new weights $B_{i,\nu}, C_j$ are chosen to maximize the ADP of (2.1). It is shown by Li [24] that when ω is any weight function on $[a, b]$, we can always obtain the maximum degree $2n(s+1) + n + 1$ by letting $\hat{\tau}_j$ be the zeros of the polynomial $\hat{\pi}_{n+1}$ that satisfies the orthogonality property

$$\int_a^b \hat{\pi}_{n+1}(t) \pi_n^{2s+1}(t) p(t) \omega(t) dt = 0, \quad \text{for all } p \in \mathbb{P}_n.$$

At the same time it is shown that $\hat{\pi}_{n+1}$ always exists and is unique up to a multiplicative constant. In the special case when $\omega(t) = (1-t^2)^{-1/2}$, Li [24] determined $\hat{\pi}_{n+1}$ explicitly and obtained the weights in (2.1) for $s = 1$ and $s = 2$. The weights in the remaining cases $s \geq 3$ were obtained later by Shi [35].

We are going to propose a different theoretical approach here in order to lay the foundation for numerical calculations of the quadrature formula (2.1) with high-precision arithmetic. It is well known (see [16]) that the nodes in Gauss-Turán quadrature formulas are all real, distinct, and internal in the interval $[a, b]$. We need a couple of facts concerning the theory of quadratures with multiple nodes, which, in particular, contains Gauss-Turán quadrature formulas. We recall the following theorem established by Ghizzetti and Ossicini [17].

THEOREM 2.1. *For any given set of odd multiplicities ν_1, \dots, ν_n , i.e., $\nu_j = 2s_j + 1$, and $s_j \in \mathbb{N}_0$, $j = 1, \dots, n$, there exists a unique quadrature formula of the form*

$$(2.2) \quad \int_a^b \omega(t) f(t) dt \approx \sum_{j=1}^n \sum_{i=0}^{\nu_j-1} a_{ji} f^{(i)}(x_j), \quad a \leq x_1 < \dots < x_n \leq b,$$

of ADP $= \nu_1 + \dots + \nu_n + n - 1$, which is the well known Chakalov-Popoviciu quadrature formula; see [5, 34]. The nodes x_1, \dots, x_n of this quadrature are determined uniquely by the orthogonality property

$$\int_a^b \omega(t) \prod_{k=1}^n (t - x_k)^{\nu_k} p(t) dt = 0, \quad \text{for all } p \in \mathbb{P}_{n-1}.$$

The corresponding (monic) orthogonal polynomial $\prod_{k=1}^n (t - x_k)$ is known in the classical literature as σ -orthogonal polynomial, with $\sigma = \sigma_n = (s_1, \dots, s_n)$, where n indicates the size of the array and $\nu_k = 2s_k + 1$, $s_k \in \mathbb{N}_0$, $k = 1, \dots, n$, in the preceding formula.

Bojanov and Petrova [2] (see also [27]) stated and proved the following important theorem that reveals the relation between the standard interpolatory quadratures of type (2.2) and the quadratures for Fourier coefficients, which is of particular interest for the problem that we consider here. Namely, we will show that a Kronrod extension (2.1) of (1.2) has the same nodes as the corresponding Gauss-Kronrod quadrature formula with the weight function $(\pi_n)^{2s}\omega$. A similar result holds if—instead of extensions of Kronrod-type—we use extensions obtained by generalized averaged Gaussian quadrature formulas.

THEOREM 2.2. *For any given sets of multiplicities $\bar{\mu} := (\mu_1, \dots, \mu_k)$, $\bar{\nu} := (\nu_1, \dots, \nu_n)$, and nodes $y_1 < \dots < y_k$, $x_1 < \dots < x_n$, there exists a quadrature formula of the form*

$$\int_a^b \omega(t) \Lambda^{\bar{\mu}}(t; \mathbf{y}) f(t) dt \approx \sum_{j=1}^n \sum_{i=0}^{\nu_j-1} c_{ji} f^{(i)}(x_j),$$

where $\Lambda^{\bar{\mu}}(t; \mathbf{y}) := \prod_{m=1}^k (t - y_m)^{\mu_m}$, with $\text{ADP} = N$ if and only if there exists a quadrature formula of the form

$$\int_a^b \omega(t) f(t) dt \approx \sum_{m=1}^k \sum_{\lambda=0}^{\mu_m-1} b_{m\lambda} f^{(\lambda)}(y_m) + \sum_{j=1}^n \sum_{i=0}^{\nu_j-1} a_{ji} f^{(i)}(x_j),$$

which has $\text{ADP} = N + \mu_1 + \dots + \mu_k$. In the case $y_m = x_j$ for some m and j , the corresponding terms in both sums can be combined into one sum of the form

$$\sum_{\lambda=0}^{\mu_m + \nu_j - 1} d_{m\lambda} f^{(\lambda)}(y_m).$$

We are now in the position to state and prove the following theorem, which represents the central point in the paper.

THEOREM 2.3. *Let π_n be an s -orthogonal polynomial with respect to the weight function ω whose zeros $\tau_1 < \tau_2 < \dots < \tau_n$ are the nodes of a quadrature formula (1.2). Then there exists an interpolatory quadrature formula of the form*

$$(2.3) \quad \int_a^b f(t) [\pi_n(t)]^{2s} \omega(t) dt \approx \sum_{\nu=1}^n D_\nu f(\tau_\nu) + \sum_{j=1}^{n+1} H_j f(\tilde{\tau}_j) \quad (s \in \mathbb{N}_0)$$

with nodes $\tilde{\tau}_j$ that are ordered, i.e., $\tilde{\tau}_1 < \tilde{\tau}_2 < \dots < \tilde{\tau}_n$, which has $\text{ADP} = N$ if and only if there exists a quadrature formula of the form

$$(2.4) \quad \int_a^b f(t) \omega(t) dt \approx \sum_{\nu=1}^n \sum_{i=0}^{2s} \hat{D}_{i,\nu} f^{(i)}(\tau_\nu) + \sum_{j=1}^{n+1} \hat{H}_j f(\tilde{\tau}_j) \quad (s \in \mathbb{N}_0),$$

which has $\text{ADP} = N + 2ns$.

Proof. Since $\omega(t) dt$ is a nonnegative measure, $(\pi_n(t))^{2s}\omega(t) dt$ is a nonnegative measure as well, i.e., $(\pi_n(t))^{2s}\omega(t)$ is a new weight function (cf. [10]). According to Theorem 2.2, there exists a quadrature formula of the form (2.3) which has $\text{ADP} = N$ if and only if there exists a quadrature formula

$$\int_a^b f(t) \omega(t) dt \approx \sum_{\nu=1}^n \sum_{i=0}^{2s-1} L_{i,\nu} f^{(i)}(\tau_\nu) + \sum_{\nu=1}^n M_\nu f(\tau_\nu) + \sum_{j=1}^{n+1} P_j f(\tilde{\tau}_j) \quad (s \in \mathbb{N}_0),$$

i.e., of the form (2.4), which has $\text{ADP} = N + 2ns$. \square

Let the quadrature formula (2.3) be the standard Gauss-Kronrod quadrature with respect to the weight function $(\pi_n)^{2s}\omega$, which has $\text{ADP} = 3n + 1$. According to Theorem 2.3, the quadrature formula of the form (2.4), namely formula (2.1), has $\text{ADP} = 2ns + 3n + 1$. In this case we have $\hat{H}_j = C_j$, $\tilde{\tau}_j = \hat{\tau}_j$, $j = 1, 2, \dots, n + 1$.

In the theory of standard Gauss-Kronrod quadrature formulas, in particular for (2.3), the Stieltjes polynomials $\hat{\pi}_{n+1} := \prod_{j=1}^{n+1} (t - \hat{\tau}_j)$, whose zeros are the nodes $\tilde{\tau}_j = \hat{\tau}_j$, play an important role. Also, of foremost interest are weight functions for which the Gauss-Kronrod quadrature formula has the property that

- (i) all $n + 1$ nodes $\hat{\tau}_j$ (i.e., zeros of the Stieltjes polynomial $\hat{\pi}_{n+1}$) are in (a, b) and are simple.

Also desirable are weight functions which additionally to (i) satisfy

- (ii) the *interlacing property*, i.e., the nodes $\hat{\tau}_j$ and τ_ν separate each other (the $n + 1$ zeros of $\hat{\pi}_{n+1}$ separate the n zeros of the s -orthogonal polynomial $\pi_n(t) = \prod_{\nu=1}^n (t - \tau_\nu)$), and
- (iii) all quadrature weights are positive.

The most important case of Gauss-Kronrod quadrature formulas has been considered from the computational point of view by Laurie [23] and later by Calvetti et al. [3]. Laurie's algorithm works in the case when all quadrature weights in the Gauss-Kronrod quadrature formula are positive. Then all zeros of the Stieltjes polynomial are real, simple, belong to (a, b) (except possibly τ_1, τ_{n+1}), and the interlacing property holds. Otherwise, the algorithm is being stopped with the message "Gauss-Kronrod does not exist". The Matlab code for this method `kronrod.m` (or in the case of symbolic calculations `skronrod.m`) can be found in the toolbox by Gautschi [14] (assembled as a companion piece to the book [13]), as well as in the Mathematica package `OrthogonalPolynomials` presented in [8, 26]. In our case of the Gauss-Kronrod quadrature formula (2.3), the coefficients of the three-term recurrence relation subject to the weight function $(\pi_n)^{2s}\omega$ that we need for an implementation of Laurie's method can be determined in the same manner as in the work of Gautschi and Milovanović [15]; see also the Matlab codes `turan.m`, `sturan.m` in [14]. However, for the general case this method can be time-consuming and might not lead to a successful construction. An alternative approach is to use the methods from [28] or [37] in order to compute π_n and then to construct the three-term recurrence coefficients for the weight $(\pi_n)^{2s}\omega$ using a moment-based method such as the Chebyshev algorithm. An implementation of the Chebyshev algorithm can be found in the package `OrthogonalPolynomials` in the function `aChebyshevAlgorithm` or in the routine `chebyshev.m` in the corresponding Matlab toolbox. There is still another possibility for a computation based on Christoffel modifications of the measure, which we further discuss in Section 4. We note that the Christoffel modification algorithm can be used in machine precision arithmetic in contrast to the Chebyshev algorithm, which requires higher precision arithmetic. The Package `OrthogonalPolynomials` implements the Christoffel modification algorithm in the function `aChristoffelAlgorithm` and the Matlab toolbox has it implemented in the file `chri2.m`.

The positive interpolatory quadrature formula (2.3), if it exists, is uniquely determined. According to Theorem 2.3, its nodes τ_ν ($\nu = 1, \dots, n$), $\tilde{\tau}_j = \hat{\tau}_j$ ($j = 1, \dots, n + 1$) are in fact the nodes of the quadrature formula (2.1), τ_ν with multiplicity $2s + 1$ ($\nu = 1, \dots, n$) and $\hat{\tau}_j$ with multiplicity one ($j = 1, \dots, n + 1$). The coefficients in the quadrature formula (2.1) can be calculated using the method from [28] for interpolatory quadrature formulas with multiple nodes of variable multiplicity. Therefore, the quadrature formula constructed in this manner is uniquely determined. As it is known, in the general case of quadrature with multiple nodes,

not all quadrature weights have to be positive. Therefore, for Kronrod extensions of Gaussian quadrature formulas with multiple nodes of the type (2.1), we cannot expect property (iii) to hold in the general case.

3. An extension of Gauss-Turán quadrature formulas obtained using generalized averaged Gaussian quadrature formulas. The existence of a positive Gauss-Kronrod quadrature formula, i.e., a Gauss-Kronrod quadrature formula with all quadrature weights being positive (the property (iii) above), depends on ω , and there are several known instances where positive weights do not exist, e.g., for the Gauss-Laguerre and Gauss-Hermite cases [19]. For the Gegenbauer weight $\omega^{(\alpha,\alpha)}(t) = (1-t^2)^\alpha$, Peherstorfer and Petras [32] have shown that Gauss-Kronrod formulas for n sufficiently large and $\alpha > 5/2$ do not exist. Analogous results for the Jacobi weight function $\omega^{(\alpha,\beta)}(t) = (1-t)^\alpha(1+t)^\beta$ can be found in their paper [33], particularly the nonexistence for large n of Gauss-Kronrod formulas when $\min(\alpha, \beta) \geq 0$ and $\max(\alpha, \beta) > 5/2$. In such situations, it is of interest to find an adequate alternative to the corresponding Gauss-Kronrod quadrature formula. We are not aware of any theoretical results concerning this problem for quadrature formulas (2.1) of Kronrod-type. It seems that this is a very difficult problem from the theoretical point of view.

An alternative approach are the Anti-Gaussian formulas introduced by Laurie [22], which have been slightly generalized in [9] and in Spalević's paper [38]. Such formulas always exist and are positive. A modification of the Anti-Gaussian formulas that leads in a special case to symmetric Gauss-Lobatto formulas has been considered by Calvetti and Reichel in [4].

In [38] a very simple numerical method is proposed for the construction of the averaged Gaussian quadrature formulas. In [39], effort has been taken in order to determine whether the averaged Gaussian formulas are an adequate alternative to the corresponding Gauss-Kronrod quadrature formulas to estimate the remainder term of a Gaussian rule.

The ADP of the generalized averaged Gaussian quadrature formulas proposed in [38] is in general $\text{ADP} = 2n + 2$. Now let the quadrature formula (2.3) be the generalized averaged quadrature with respect to the weight function $(\pi_n)^{2s}\omega$ constructed in [38]. It has $\text{ADP} = 2n + 2$, and according to Theorem 2.3, the quadrature formula of the form (2.4), namely the quadrature formula (2.1), has $\text{ADP} = 2ns + 2n + 2$. In this case we have coefficients and nodes $\hat{H}_j = C_j$, $\hat{\tau}_j = \hat{\tau}_j$, $j = 1, 2, \dots, n + 1$. Again, we use the methods from [28, 37] in order to compute π_n and then construct three-term recurrence coefficients subject to the weight function $(\pi_n)^{2s}\omega$. These are needed for the implementation of Spalević's method [38], which uses moment-based methods such as the Chebyshev algorithm. The weights in the quadrature formula (2.1) can be calculated using the method from [28] for interpolatory quadrature formulas with multiple nodes with variable multiplicities.

4. Numerical results. A very popular method for obtaining a practical error estimate in the numerical integration by standard quadrature is to use two quadrature formulae A and B , where the nodes in formula B form a proper subset of those in formula A , and where the rule A is of higher degree of precision. Kronrod originated this method; see [20]. For more details concerning this theory for standard quadrature formulas see, e.g., [22, 29, 30, 38, 39]. The difference $|A(f) - B(f)|$, where f is the integrand, is usually quite a good estimate of the error for the rule B . Here, let the rule B be the Gauss-Turán quadrature rule (1.2) and the rule A be an extension (2.1), i.e., a Kronrod extension or an extension obtained using generalized averaged Gaussian quadrature formulas.

For the construction of (2.1), the Mathematica package `OrthogonalPolynomials` presented in [8, 26] is used. We demonstrate the construction of the quadrature formula for the Legendre weight function with $n = 8$ and $s = 2$. First we use the function `aTuránNodes`, which computes the nodes of the Gauss-Turán quadrature rule for the given measure. The format of the function call `aTuránNodes` is as follows.

TABLE 4.1

Nodes and weights in the formula (2.1) for the Legendre weight function with $n = 8$ and $s = 2$. Numbers in parentheses represent exponents in the base 10.

nodes		
	τ_ν	τ_ν
Gauss-Turán	$\pm.97351341389220656237$	$\pm.81887893183941851633$
	$\pm.54466490130324525777$	$\pm.19085290843984904479$
	$\hat{\tau}_\nu$	$\hat{\tau}_\nu$
Kronrod	$\pm.99822290664226966427$	$\pm.91312742966813842749$
	$\pm.69453148147735680361$	$\pm.37468411056055517193$
	0	

weights		
i	$B_{i,\nu}$	$B_{i,\nu}$
0	$.58386175714956789764(-1)$	$.15089370723807276066$
0	$.22074279238149145536$	$.25860501785904752596$
1	$\mp.39014622164968142227(-3)$	$\mp.83891790441786127084(-3)$
1	$\mp.82125588688818075943(-3)$	$\mp.33535947356365552771(-3)$
2	$.71919503334752507270(-5)$	$.11322886240186814938(-3)$
2	$.35125693562506992715(-3)$	$.56357216956268884559(-3)$
3	$\mp.23883755204324643150(-7)$	$\mp.34341240517473823534(-6)$
3	$\mp.71953009372658394643(-6)$	$\mp.40311953782588033969(-6)$
4	$.11870030713306244483(-9)$	$.13777802473954649246(-7)$
4	$.92258193720323290033(-7)$	$.20363103482496320695(-6)$
	C_j	C_j
	$.52356392061325635414(-2)$	$.48662615167843382200(-1)$
	$.86208495603928504804(-1)$	$.11125708395348622280$
	$.12001694575008158983$	

```
aTuranNodes[8, {aLegendre}, 2, WorkingPrecision->50,
Precision->45, AlgorithmSigma->IncreaseDegree]
```

This function constructs nodes for the Gauss-Turán quadrature rule (1.2) for $\omega = 1$, $n = 8$, and $s = 2$. It applies an algorithm presented in [28]. It should be noted that variable precision arithmetic is used. The reason for employing variable precision arithmetic is the application of the Chebyshev algorithm, which we discuss below. Table 4.1 shows the Gauss-Turán nodes obtained using the previous function call. Note that the nodes are given with 20 decimal digits precision. Alternatively to the algorithm in [28] used here, the method presented in [37] can be used with `AlgorithmSigma->Homotopy`, but this is more time-consuming although it is more general and can be applied to arbitrary weight functions.

When the nodes of the Gauss-Turán quadrature rule have been constructed, we can compute moments of the weight function $(\pi_n)^{2s}\omega$. In order to apply the Chebyshev algorithm for the computation of the three-term recurrence relation coefficients, we need to calculate the moments up to the order $2n + 1$. The moments can be computed using Gaussian quadrature rules for the weight function ω . If the weight function ω belongs to the class of those for which a three-term recurrence is known in analytic form—as it is the case for the Legendre weight function, then we can apply the function `aGaussianNodesWeights` as follows.

```
{nG, wG}=aGaussianNodesWeights[n, {aLegendre},
WorkingPrecision->50, Precision->50]
```

Here nG and wG are the computed Gaussian nodes and weights. As can be seen, we have to work in variable precision arithmetic. In our case we used 50 decimal digits for the mantissa, which is in accordance with the number of decimal digits employed in the computation of Gauss-Turán nodes. Also note that we compute a Gaussian quadrature rule having $ADP = ns + n + 1$. If $x_i^G, w_i^G, i = 1, \dots, ns + n + 1$, are the constructed nodes and weight of the Gaussian quadrature rule for the weight function ω , we compute

$$\int_a^b t^k (\pi_n(t))^{2s} \omega(t) dt = \sum_{i=1}^{ns+n+1} w_i^G (x_i^G)^k (\pi_n(x_i^G))^{2s}, \quad k = 0, \dots, 2n + 1.$$

When moments are computed, we use the function `aChebyshevAlgorithm`, to find the coefficients of the three-term recurrence relation.

```
aChebyshevAlgorithm[mom, WorkingPrecision->50]
```

The application of Chebyshev's algorithm requires high precision arithmetic. Actually, it can be observed that we have used 50 decimal digits precision in the calls to the functions `aTuranNodes` and `aChebyshevAlgorithm` in order to obtain 20 decimal digits of the coefficients of the three-term recurrence. This is generally the case in computations involving Chebyshev's algorithm when the number of requested three term recurrence coefficients is larger than ten. It should be mentioned here that there exists a modified Chebyshev algorithm (see [13, p. 76]) which has—with good tuning—a much better numerical characteristics than the ordinary Chebyshev algorithm. However, there is a serious problem finding good sequences of modified moments. The interested user is referred to [1] for more information. An implementation of the modified Chebyshev algorithm can be found in both software packages. However, due to limited space, we are not going into details.

However, there is an alternative approach which can be used and is based on the application of Christoffel modifications of the measure with the quadratic factors. For further information, the reader might consult [13, pp. 135] or the original works of Christoffel presented in [6, 7], which are reformulated as algorithms in [11, 12]. Let ω be the weight function with three-term recurrence coefficients α_k and $\beta_k, k \in \mathbb{N}_0$, then the three-term recurrence coefficients $\hat{\alpha}_k, \hat{\beta}_k, k \in \mathbb{N}_0$, for the weight $(t - c)^2 \omega(t)$ can be computed numerically stable using the above mentioned Christoffel algorithm. We can construct three-term recurrence coefficients for the weight function $(\pi_n)^{2s} \omega$ using ns applications of Christoffel modifications. Assume that the three-term recurrence coefficients for the measure ω are known and set $w := \omega$. We apply a series of Christoffel modifications to compute the three-term recurrence coefficients for the weights $w(t) := (t - \tau_\nu)^2 w(t), \nu = 1, \dots, n, j = 1, \dots, s$. The package `OrthogonalPolynomials` has implemented the Christoffel modification with a quadratic factor in the function `aChristoffelAlgorithm`. For example, the following sequence of code can be used for the construction of three-term recurrence coefficients of the weight $(\pi_n)^{2s} \omega_H$, where ω_H is Hermite weight function.

```
{alH, beH} = Transpose[aHermite["ttr"]/@Table[k, {k, 0, nn}]];
For[i=1, i<=n, ++i,
  For[j=1, j<=s, ++j,
    {alH, beH} = aChristoffelAlgorithm[nn--,
      {aQuadraticReal, nT[[i]]},
      alH, beH,
      WorkingPrecision->$MachinePrecision];
  ];
];
```

Note that we are working in double precision arithmetic, and we assume that the variable `nT` holds a list of Turán nodes for the Hermite weight function. Assuming this form of the

TABLE 4.2

Error estimation using the difference of the Gauss-Turán quadrature rule and its Kronrod extension for the integral (4.1).

$s = 2$			$n = 6$		
n	estimate	error	s	estimate	error
6	.35174(-5)	.35253(-5)	2	.35174(-5)	.35253(-5)
10	.95272(-10)	.952815(-10)	4	.95027(-9)	.95177(-9)
14	.23752562(-14)	.2375268(-14)	6	.28504(-12)	.28539(-12)
18	.5796464(-19)	.5796466(-19)	8	.90176(-16)	.90264(-16)
22	.1401631557(-23)	.1401631596(-23)	10	.29464(-19)	.29488(-19)

construction of three-term recurrence coefficients with $n = 8$ and $s = 2$, we obtain three-term recurrence coefficients with 14 decimal digits precision while working with double precision numbers. In the same manner, even better results can be obtained for the Legendre weight function.

Nowadays, an application of higher precision arithmetic is simply a matter of choice. There is wide variety of software packages which implement arbitrary precision arithmetic. Starting with high level languages such as Mathematica and ending with C software packages such as the GNU multiprecision package (GMP). More about GMP can be found at <http://gmplib.org>. GMP is actually the working horse used in Mathematica for the implementation of arbitrary precision arithmetic. According to these trends, we decided to present a construction based on Chebyshev's algorithm since it is quite simple and gives some sense of uniformity. This is especially convenient for the non-specialists in the area of the computational theory of orthogonal polynomials.

We just need to apply Laurie's algorithm for the construction of Gauss-Kronrod quadrature rules (see [23]) to compute the Jacobi matrix, whose eigenvalues are nodes for the quadrature rule (2.3). The package `OrthogonalPolynomials` can be used for this purpose with the command

```
aLaurie[n, a1, be, WorkingPrecision->50]
```

where `a1` and `be` are three-term recurrence coefficients for the weight $(\pi_n)^{2s}\omega$. After an application of Laurie's algorithm, we can construct the missing nodes for the Gauss-Kronrod quadrature rule by using the function `azero`, which computes eigenvalues of the Gauss-Kronrod matrix. The missing nodes of the Gauss-Kronrod quadrature rule are presented in the Table 4.1. As for the Gaussian nodes, the nodes are given with 20 decimal digits precision.

Finally, we can compute weights of the quadrature rule (2.4) using an interpolation property. For example, the function `aInterpolationWeights` in the Mathematica package `OrthogonalPolynomials` can be used. The computed weights are given in the Table 4.1. Numbers in parentheses represent exponents in the base 10.

To illustrate the possibility of an error estimation, we calculate the integral

$$(4.1) \quad I_1 = \int_{-1}^1 \frac{dt}{t + \frac{11}{10}} = \log 21 \approx 3.0445224377234229965005979803657054343\dots$$

Table 4.2 displays the results of the error estimation using differences between Gauss-Turán quadrature rule represented by equation (1.2) and Kronrod extension of the Gauss-Turán quadrature rule represented by equation (2.1). In the first two columns, we present results for variable n and fixed $s = 2$; the second two columns show the behavior of the error estimate for fixed $n = 6$ and variable s . As can be seen, the error estimate is quite accurate.

TABLE 4.3

Quadrature rule (2.1), $n = 8, s = 2$, for the Hermite weight function based on the Anti-Gaussian quadrature formula.

		nodes	
Gauss-Turán		τ_ν	τ_ν
		$\pm.51681136909405826789(1)$	$\pm.34836231485926327975(1)$
		$\pm.20317347623297887471(1)$	$\pm.66893558851736928468$
Anti-Gaussian		$\hat{\tau}_\nu$	$\hat{\tau}_\nu$
		$\pm.63342943088252741257(1)$	$\pm.42938819278505047838(1)$
		$\pm.27471245421883808223(1)$	$\pm.13464957031164851511(1)$
		0	
		weights	
i		$B_{i,\nu}$	$B_{i,\nu}$
0		.36175515137157045962(-10)	.16750466449875911971(-4)
0		.22298920872663214573(-1)	.59073792759942283728
1		$\mp.10156104835511174186(-10)$	$\mp.41614132181940995042(-5)$
1		$\mp.41438045323230060806(-2)$	$\mp.42505466030715357444(-1)$
2		.12253977340516115977(-11)	.54025501379426880713(-6)
2		.67462832585667770868(-3)	.16841301453891941760(-1)
3		$\mp.73364318178749481946(-13)$	$\mp.34862739064811316798(-7)$
3		$\mp.38565517760469799699(-4)$	$\mp.41767093230057802043(-3)$
4		.19217294376648553599(-14)	.13257166120337859521(-8)
4		.24969557547643169509(-5)	.81926585192629552560(-4)
		C_j	C_j
		.33157509907727129273(-17)	.50643909703274783901(-8)
		.23467083711191469645(-3)	.68090277501526014491(-1)
		.40969674615003533584	

The construction of the quadrature formula (2.1) under the assumption that the underlying formula is Anti-Gaussian can be achieved using similar techniques. The Package `OrthogonalPolynomials` has the function `aAntiGaussianNodesWeights`, which can be used for the construction of the nodes and weights in the formula (2.3). For the successful construction, we need the three-term recurrence coefficients for the weight function $(\pi_n)^{2s}\omega$. These recurrence coefficients can be constructed using similar methods as above. Once the nodes $\tau_\nu, \nu = 1, \dots, n, \tilde{\tau}_j, j = 1, \dots, n + 1$, are found, we can apply the function `aInterpolationWeights` to compute the missing weights for the quadrature rule (2.4). Table 4.3 displays the results of the construction for the Hermite weight function.

The feasibility of an error estimation in Gauss-Turán quadrature rules is presented in Table 4.4. The integral used for this demonstration is

$$(4.2) I_2 = \int_{-\infty}^{\infty} \frac{e^{-t^2}}{1+t^2} dt = e\pi \operatorname{Erfc}(1) \approx 1.34329342164673517043712359441059\dots$$

As can be seen from the table, the error estimate is quite accurate, as it happened with the previous quadrature formula.

During the performed numerical constructions with the Hermite weight function, one interesting theoretical result appeared. We present it in the next theorem.

TABLE 4.4

Error estimation using the difference of the Gauss-Turán quadrature rule and its Anti-Gaussian extension for the integral (4.2).

$s = 2$			$n = 6$		
n	estimate	error	s	estimate	error
6	.6819(-3)	.6734(-3)	2	.6819(-3)	.6734(-3)
10	.2403(-4)	.2377(-4)	4	.1954(-3)	.1957(-3)
14	.1498(-5)	.1484(-5)	6	.9441(-4)	.9530(-4)
18	.1333(-6)	.1323(-6)	8	.5847(-4)	.5931(-4)
22	.1521(-7)	.1511(-7)	10	.4159(-4)	.4233(-4)

THEOREM 4.1. Let $n \in \mathbb{N}$ be fixed. Let $\pi_k, k \in \mathbb{N}_0$, be the sequence of monic s -orthogonal polynomials with respect to the weight function $(\pi_n(t))^{2s} e^{-t^2}$ supported on the real line and let $\alpha_k = 0$ and $\beta_k, k \in \mathbb{N}_0$, be the coefficients in the recurrence relation

$$\pi_{k+1}(t) = t\pi_k(t) - \beta_k\pi_{k-1}(t), \quad k \in \mathbb{N}.$$

Then

$$\beta_n = n \left(s + \frac{1}{2} \right).$$

Proof. The proof relies on the fact that the weight function $(\pi_n(t))^{2s} e^{-t^2}$ is semiclassical. It can be easily shown that

$$\left((\pi_n(t))^{2s+1} e^{-t^2} \right)' = ((2s+1)(\pi_n)'(t) - 2t\pi_n(t))(\pi_n(t))^{2s} e^{-t^2}.$$

If we integrate the previous equation multiplied by $t^k, k = 0, \dots, n$, over the real line, we get, due to orthogonality,

$$\begin{aligned} \int_{-\infty}^{\infty} ((2s+1)(\pi_n)'(t) - 2t\pi_n(t)) t^k (\pi_n(t))^{2s} e^{-t^2} dt &= \int_{-\infty}^{\infty} \left((\pi_n(t))^{2s+1} e^{-t^2} \right)' t^k dt \\ &= -k \int_{-\infty}^{\infty} \pi_n(t) t^{k-1} (\pi_n(t))^{2s} e^{-t^2} dt = 0, \quad k = 0, \dots, n. \end{aligned}$$

Hence, it must hold that

$$(2s+1)(\pi_n)' - 2t\pi_n = -2\pi_{n+1} = -2(t\pi_n - \beta_n\pi_{n-1}), \quad (2s+1)(\pi_n)' = 2\beta_n\pi_{n-1}.$$

Using this equation we get $\beta_n = (2s+1)n/2$. \square

This theorem can be used to measure the efficiency of the Chebyshev algorithm since we can compare results of the Chebyshev algorithm and the exact value for β_n for the Hermite weight function.

Acknowledgment. We would like to thank the anonymous referees and Lothar Reichel for helpful comments.

REFERENCES

- [1] B. BECKERMANN AND E. BOURREAU, *How to choose modified moments?*, J. Comput. Appl. Math., 98 (1998), pp. 81–98.
- [2] B. BOJANOV AND G. PETROVA, *Quadrature formulae for Fourier coefficients*, J. Comput. Appl. Math., 231 (2009), pp. 378–391.
- [3] D. CALVETTI, G. H. GOLUB, W. B. GRAGG, AND L. REICHEL, *Computation of Gauss-Kronrod rules*, Math. Comp., 69 (2000), pp. 1035–1052.
- [4] D. CALVETTI AND L. REICHEL, *Symmetric Gauss-Lobato and modified anti-Gauss rules*, BIT, 43 (2003), pp. 541–554.
- [5] L. CHAKALOV, *General quadrature formulae of Gaussian type*, Bulgar. Akad. Nauk. Izv. Mat. Inst., 1 (1954), pp. 67–84.
- [6] E. B. CHRISTOFFEL, *Über die Gaußsche Quadratur und eine Verallgemeinerung derselben*, J. Reine Angew. Math., 55 (1858), pp. 61–82.
- [7] ———, *Sur une classe particulière de fonctions entières et de fractions continues*, Ann. Mat. Pura Appl., 8 (1877), pp. 1–10.
- [8] A. S. CVETKOVIĆ AND G. V. MILOVANOVIĆ, *The Mathematica package “OrthogonalPolynomials”*, Facta Univ. Ser. Math. Inform., 19 (2004), pp. 17–36.
- [9] S. EHRICH, *On stratified extensions of Gauss-Laguerre and Gauss-Hermite quadrature formulas*, J. Comput. Appl. Math., 140 (2002), pp. 291–299.
- [10] H. ENGELS, *Numerical Quadrature and Cubature*, Academic Press, London, 1980.
- [11] D. GALANT, *An implementation of Christoffel’s theorem in the theory of orthogonal polynomials*, Math. Comp., 25 (1971), pp. 111–113.
- [12] W. GAUTSCHI, *A survey of Gauss-Christoffel quadrature formulae*, in E. B. Christoffel. The Influence of his Work on Mathematics and the Physical Sciences, P. L. Butzer and F. Fehér, eds., Birkhäuser, Basel, 1981, pp. 72–147.
- [13] ———, *Orthogonal Polynomials: Computation and Approximation*, Oxford University Press, New York, 2004.
- [14] ———, *OPQ: a Matlab suite of programs for generating orthogonal polynomials and related quadrature rules*, Walter Gautschi 2002 Code Archive, (2002).
www.cs.purdue.edu/archives/2002/wxg/codes/OPQ.html.
- [15] W. GAUTSCHI AND G. V. MILOVANOVIĆ, *S-orthogonality and construction of Gauss-Turán-type quadrature formulae*, J. Comput. Appl. Math., 86 (1997), pp. 205–218.
- [16] A. GHIZZETTI AND A. OSSICINI, *Quadrature Formulae*, Akademie-Verlag, Berlin, 1970.
- [17] ———, *Sull’ esistenza e unicità delle formule di quadratura gaussiane*, Rend. Mat., 8 (1975), pp. 1–15.
- [18] G. H. GOLUB AND J. KAUTSKY, *Calculation of Gauss quadratures with multiple free and fixed knots*, Numer. Math., 41 (1983), pp. 147–163.
- [19] D. K. KAHANER AND G. MONEGATO, *Nonexistence of extended Gauss-Laguerre and Gauss-Hermite quadrature rules with positive weights*, Z. Angew. Math. Phys., 29 (1978), pp. 983–986.
- [20] A. S. KRONROD, *Integration with control of accuracy*, Soviet Physics Dokl., 9 (1964), pp. 17–19.
- [21] A. KROÓ AND F. PEHERSTORFER, *Asymptotic representation of L_p -minimal polynomials*, $1 < p < \infty$, Constr. Approx., 25 (2007), pp. 29–39.
- [22] D. P. LAURIE, *Anti-Gaussian quadrature formulas*, Math. Comp., 65 (1996), pp. 739–747.
- [23] ———, *Calculation of Gauss-Kronrod quadrature rules*, Math. Comp., 66 (1997), pp. 1133–1145.
- [24] S. LI, *Kronrod extension of Turán formula*, Studia Sci. Math. Hungarica, 29 (1994), pp. 71–83.
- [25] G. V. MILOVANOVIĆ, *Quadratures with multiple nodes, power orthogonality, and moment-preserving spline approximation*, J. Comput. Appl. Math., 127 (2001), pp. 267–286.
- [26] G. V. MILOVANOVIĆ AND A. S. CVETKOVIĆ, *Special classes of orthogonal polynomials and corresponding quadratures of Gaussian type*, Math. Balkanica, 26 (2012), pp. 169–184.
- [27] G. V. MILOVANOVIĆ AND M. M. SPALEVIĆ, *Kronrod extensions with multiple nodes of quadrature formulas for Fourier coefficients*, Math. Comp., to appear in print, published online on August 28, 2013.
- [28] G. V. MILOVANOVIĆ, M. M. SPALEVIĆ, AND A. S. CVETKOVIĆ, *Calculation of Gaussian-type quadratures with multiple nodes*, Math. Comput. Modelling, 39 (2004), pp. 325–347.
- [29] G. MONEGATO, *Stieltjes polynomials and related quadrature rules*, SIAM Rev., 24 (1982), pp. 137–158.
- [30] ———, *An overview of the computational aspects of Kronrod quadrature rules*, Numer. Algorithms, 26 (2001), pp. 173–196.
- [31] F. PEHERSTORFER, *Gauss-Turán quadrature formulas: asymptotics of weights*, SIAM J. Numer. Anal., 47 (2009), pp. 2638–2659.
- [32] F. PEHERSTORFER AND K. PETRAS, *Ultraspherical Gauss-Kronrod quadrature is not possible for $\lambda > 3$* , SIAM J. Numer. Anal., 37 (2000), pp. 927–948.
- [33] ———, *Stieltjes polynomials and Gauss-Kronrod quadrature for Jacobi weight functions*, Numer. Math., 95 (2003), pp. 689–706.

- [34] T. POPOVICIU, *Sur une généralisation de la formule d'intégration numérique de Gauss*, Acad. R. P. Romîne Fil. Iași Stud. Cerc. Ști., 6 (1955), pp. 29–57.
- [35] Y. G. SHI, *Generalized Gaussian Kronrod-Turán quadrature formulas*, Acta Sci. Math. (Szeged), 62 (1996), pp. 175–185.
- [36] ———, *Christoffel type functions for m -orthogonal polynomials*, J. Approx. Theory, 137 (2005), pp. 57–88.
- [37] Y. G. SHI AND G. XU, *Construction of σ -orthogonal polynomials and Gaussian quadrature formulas*, Adv. Comput. Math., 27 (2007), pp. 79–94.
- [38] M. M. SPALEVIĆ, *On generalized averaged Gaussian formulas*, Math. Comp., 76 (2007), pp. 1483–1492.
- [39] ———, *A note on generalized averaged Gaussian formulas*, Numer. Algorithms, 46 (2007), pp. 253–264.
- [40] P. TURÁN, *On the theory of the mechanical quadrature*, Acta Sci. Math. (Szeged), 12 (1950), pp. 30–37.