



# Generalized Gaussian Quadratures for Integrals with Logarithmic Singularity

Gradimir V. Milovanović<sup>a</sup>

<sup>a</sup>*Serbian Academy of Sciences and Arts, Beograd, Serbia & State University of Novi Pazar, Serbia*

**Abstract.** A short account on Gaussian quadrature rules for integrals with logarithmic singularity, as well as some new results for weighted Gaussian quadrature formulas with respect to generalized Gegenbauer weight  $x \mapsto |x|^\gamma(1-x^2)^\alpha$ ,  $\alpha, \gamma > -1$ , on  $(-1, 1)$ , which are appropriated for functions with and without logarithmic singularities, are considered. Methods for constructing such kind of quadrature formulas and some numerical examples are included.

## 1. Introduction

In this paper we give a short account on Gaussian quadrature rules for integrals with logarithmic singularity, as well as some new results for weighted quadrature rules of Gaussian type with respect to generalized Gegenbauer weight on  $(-1, 1)$  suited for functions with and without logarithmic singularities. Some of these results have been recently presented during author's plenary lecture at the 28th International Conference of The Jangjeon Mathematical Society (May 15–19, 2015, Akdeniz University Antalya, Turkey).

The paper is organized as follows. A history of Gaussian quadrature formulas and basic connection to orthogonal polynomials are presented in Section 2. Gaussian quadratures for non-polynomial systems are introduced in Section 3. Generalized weighted Gaussian formulae for functions with a logarithmic singularity are considered in Section 4. In particular, quadrature rules of Gaussian type with respect to the generalized Gegenbauer weight are presented. Such rules are able to integrate functions with a sufficient accuracy, regardless of whether they are smooth, or contain a logarithmic singularity. Methods of construction such kind of quadratures, as well as some numerical examples, are given in Sections 5 and 6.

## 2. Gaussian quadratures and orthogonal polynomials

One of the most significant event of the 19th century in the field of numerical integration and perhaps in all of numerical analysis was the discovery of Gaussian quadratures in 1814. While interpolatory Newton-Cotes formula for numerical integration from 1676,

$$I(f) = \int_a^b f(x) dx \approx Q_n(f) = \sum_{k=1}^n A_k f(x_k), \quad (1)$$

2010 *Mathematics Subject Classification.* Primary 65D30; Secondary 41A55, 65D32, 33C45.

*Keywords.* numerical integration, singular integrals, Gaussian quadrature, orthogonalization, boundary element method.

Received: 31 August 2015; Accepted: 17 September 2015

Communicated by Gradimir V. Milovanović

The work was supported in part by the Serbian Ministry of Education, Science and Technological Development.

*Email address:* [gvm@mi.sanu.ac.rs](mailto:gvm@mi.sanu.ac.rs) (Gradimir V. Milovanović)

with fixed nodes (usually selected equidistantly on  $[a, b]$ ), is exact only for algebraic polynomials of degree at most  $n - 1$ , the corresponding Gaussain formula is exact for polynomials of degree at most  $2n - 1$ . This dramatically increasing of degree of exactness has enabled by a free choice of nodes  $x_k, k = 1, \dots, n$ . Starting only from the previous work of Newton and Cotes and using his own result on continued fractions associated with hypergeometric series, Gauss proved this result. It is interesting to mention that in the case  $[a, b] = [0, 1]$ , Gauss [5] determined numerical values of the nodes  $x_k$  and the weight coefficients  $A_k, k = 1, \dots, n$ , for all  $n \leq 7$ , with almost 16 significant decimal digits. An alternative (and elegant) derivation of Gaussian formulas was provided by Jacobi [14], using theory of orthogonality. His result can be stated in the form: *Given a positive integer  $m (\leq n)$ , the quadrature rule  $Q_n(f)$  in (1) has an algebraic degree of exactness  $d = n - 1 + m$  if and only if the following conditions are satisfied:*

1° Formula  $Q_n(f)$  is interpolatory;

2° The node polynomial  $q_n(x) = (x - x_1)(x - x_2) \cdots (x - x_n)$  satisfies  $I(q_n p) = 0$  for each polynomial of degree at most  $m - 1$ .

Let  $\mathcal{P}_n$  be the set of all algebraic polynomials of degree at most  $n, \mathcal{P}$  be the set of all algebraic polynomials, and  $(p, q)$  be an inner product defined on  $(a, b) = (-1, 1)$  by  $(p, q) = \int_{-1}^1 p(x)q(x) dx \ (p, q \in \mathcal{P})$ .

According to the previous Jacobi result, an  $n$ -point quadrature formula (1) has the maximal degree of exactness  $2n - 1$ , i.e.,  $m = n$  is optimal, because the higher  $m (> n)$  is impossible. Indeed, the condition 2° for  $m = n + 1$  requires the orthogonality  $I(q_n p) = (q_n, p) = 0$  for all  $p \in \mathcal{P}_n$ , which is impossible when  $p = q_n$ , because  $I(q_n^2) = (q_n, q_n) > 0$ .

Thus, in the case  $m = n$ , the orthogonality condition 2° shows that the node polynomial  $q_n$  must be (monic) orthogonal polynomial on  $(-1, 1)$ , and therefore the nodes  $x_k$  must be zeros of the polynomial  $q_n(x)$ , which is exactly the monic Legendre polynomial of degree  $n$ .

After further contributions by Mehler, Radau, Heine, Posse, etc., Christoffel generalized the previous Gauss-Jacobi theory to arbitrary weighted integrals  $I(f; w) = \int_a^b f(x)w(x) dx$ , and Stieltjes to integrals with respect to the positive measures  $d\mu(x)$  on the real line with finite or unbounded support,  $I(f; d\mu) = \int_{\mathbb{R}} f(x) d\mu(x)$ . A nice survey of Gauss-Christoffel quadrature formulae was given by Gautschi [6].

In this paper we deal with Gauss-Christoffel quadratures

$$\int_a^b f(x)w(x) dx = \sum_{k=1}^n A_k f(x_k) + R_n(w; f), \tag{2}$$

with respect to the weight function  $w : (a, b) \rightarrow \mathbb{R}^+$ , for which all moments  $\mu_k = \int_a^b x^k w(x) dx$  exist and are finite, and  $\mu_0 > 0$ . The quadrature sum will be denoted By  $Q_n(w; f)$  and  $R_n(w; f)$  we denote the quadrature sum and the corresponding remainder term, respectively. Under previous conditions for moments, for each  $n \in \mathbb{N}$ , there exists the  $n$ -point Gauss-Christoffel quadrature formula (2) which is exact for all algebraic polynomials of degree  $\leq 2n - 1$ , i.e.,  $R_n(w; f) = 0$  for each  $f \in \mathcal{P}_{2n-1}$ .

Thus, the Gauss-Christoffel quadrature formula can be characterized as an interpolatory formula for which its *node polynomial*  $q_n(t) = \prod_{k=1}^n (x - x_k)$  is orthogonal to  $\mathcal{P}_{n-1}$  with respect to the inner product defined by

$$(p, q) = \int_a^b p(x)q(x)w(x) dx \quad (p, q \in \mathcal{P}). \tag{3}$$

Therefore, orthogonal polynomials play an important role in the analysis and construction of such quadrature formulas of the maximal algebraic degree of exactness.

The monic polynomials  $\pi_\nu(\cdot) = \pi_\nu(w; \cdot), \nu = 0, 1, \dots$ , orthogonal with respect to (3) satisfy the three-term recurrence relation (cf. [19, p. 97])

$$\begin{aligned} \pi_{\nu+1}(x) &= (x - \alpha_\nu)\pi_\nu(x) - \beta_\nu\pi_{\nu-1}(x), \quad \nu = 0, 1, \dots, \\ \pi_0(x) &= 1, \quad \pi_{-1}(x) = 0, \end{aligned} \tag{4}$$

with recurrence coefficients  $\alpha_v = \alpha_v(w)$  and  $\beta_v = \beta_v(w) > 0$ , and  $\beta_0 = \mu_0 = \int_{-1}^1 w(x) dx$  (by definition).

For numerical construction of quadrature rules (2) of the maximal algebraic degree of exactness, there are several stable algorithms. The most popular of them is one due to Golub and Welsch [12], which is based on determining the eigenvalues and the first components of the corresponding normalized eigenvectors of the following symmetric tridiagonal Jacobi matrix

$$J_n(w) = \begin{bmatrix} \alpha_0 & \sqrt{\beta_1} & & & 0 \\ \sqrt{\beta_1} & \alpha_1 & \sqrt{\beta_2} & & \\ & \sqrt{\beta_2} & \alpha_2 & \ddots & \\ & & \ddots & \ddots & \sqrt{\beta_{n-1}} \\ 0 & & & \sqrt{\beta_{n-1}} & \alpha_{n-1} \end{bmatrix},$$

where  $\alpha_v$  and  $\beta_v, v = 0, 1, \dots, n - 1$ , are the coefficients in the three-term recurrence relation (4) for the monic orthogonal polynomials  $\pi_v(w; \cdot)$ . The nodes  $x_k, k = 1, \dots, n$ , in the weighted Gauss-Christoffel quadrature formula (2) are the eigenvalues of the Jacobi matrix  $J_n(w)$ , and the weights  $A_k$  are given by

$$A_k = \beta_0 v_{k,1}^2, \quad k = 1, \dots, n,$$

where  $\beta_0 = \mu_0$  and  $v_{k,1}$  is the first component of the normalized eigenvector  $\mathbf{v}_k$  corresponding to the eigenvalue  $x_k$ ,

$$J_n(w)\mathbf{v}_k = x_k \mathbf{v}_k, \quad \mathbf{v}_k^T \mathbf{v}_k = 1, \quad k = 1, \dots, n. \tag{5}$$

Thus, if recursive coefficients in the three-term recurrence relation (4) are known explicitly, as for example, in the case of *very classical* orthogonal polynomials, the construction of quadratures with the maximal algebraic degree of exactness (Gaussian quadratures in the polynomial case) can be realized very easy for arbitrary  $n \in \mathbb{N}$ . Orthogonal polynomials for which the recursion coefficients are not known we call *strongly non-classical polynomials*. In such cases we need a stable numerical construction of recursive coefficients based on the Gautschi *constructive theory* of orthogonal polynomials on the real line [7], [8], [9] (see also [10], [19], [21]).

Recent progress in symbolic computation and variable-precision arithmetic now makes it possible to generate the recurrence coefficients  $\alpha_k$  and  $\beta_k$  directly by using the original Chebyshev method of moments, but in a sufficiently high precision arithmetic. Such an approach enables us to overcome the numerical instability. Respectively symbolic/variable-precision software for orthogonal polynomials is available: Gautschi’s package SOPQ in MATLAB and our MATHEMATICA package OrthogonalPolynomials (see [4], [23], [21]), which is downloadable from the web site <http://www.mi.sanu.ac.rs/~gvm/>. A survey on quadrature processes and new applications has been recently given in [20].

Unfortunately, such elegant tools do not exist for non-polynomial basis systems of functions and related Gaussian quadratures, which will be considered in the following section.

### 3. Gaussian quadratures for non-polynomial systems

Gauss’s method can be extended in a natural way to non-polynomial functions, taking a system of linearly independent functions

$$\{\psi_1(x), \psi_2(x), \dots\} \quad (x \in [a, b]), \tag{6}$$

usually chosen to be complete in some suitable space of functions.

If  $w(x)$  is a given nonnegative weight function on  $[a, b]$  and the quadrature rule

$$\int_a^b f(x)w(x) dx = \sum_{k=1}^n A_k f(x_k) + R_n(w; f) \tag{7}$$

is such that it integrates exactly the first  $2n$  functions in (6), then the rule (7) is called *Gaussian with respect to the system* (6). The existence and uniqueness of such a quadrature rule of Gaussian type for a non-polynomial system (6) is always guaranteed if the first  $2n$  functions of this system constitute a Chebyshev system on  $[a, b]$ . In that case, all the weights  $A_1, \dots, A_n$  in (7) are positive. In terms of moment spaces, the Gaussian rule corresponds to the unique *lower principal representation* of the measure  $d\mu(x) = w(x) dx$  (see Karlin and Studden [15]).

A further refinement of the results of existence and uniqueness of Gaussian quadrature rules for non-polynomial systems (in the other words, for the so-called generalized Gaussian quadrature rules) was given by Ma, Rokhlin, and Wandzura [18]. Suppose a system of functions  $\psi_k, k = 1, \dots, 2n$ , which are continuous on  $(a, b]$ , and that there exists a function  $r$  integrable and continuous on  $(a, b]$ , such that  $|\psi_k/r| < +\infty$  as  $x \rightarrow a^+$ . If  $\psi_k/r, k = 1, \dots, 2n$ , constitute the Chebyshev system on  $[a, b]$ , then the generalized Gaussian quadrature formula (7) exists uniquely. Further, assuming that system of functions  $\psi_k, k = 1, \dots, 2n$ , is Chebyshev on the interval  $(a, b]$  and integrable (w.r.t. to a given weight function  $w$ ), we have that Gaussian quadrature rule exists uniquely (see [18]).

The most important systems of functions which satisfy the previous condition are the well known Müntz systems.

A construction of the generalized Gaussian quadrature formula (7) was considered in [18], taking a Chebyshev system of functions  $\{\psi_1, \psi_2, \dots, \psi_{2n}\}$  on  $[a, b]$  with the following properties:

- (1)  $\psi_k \in C^1[a, b]$  ( $k = 1, \dots, 2n$ );
- (2) the determinants

$$D_{2n}(\psi_1, \psi_2, \dots, \psi_{2n}) = \begin{vmatrix} \psi_1(x_1) & \cdots & \psi_1(x_n) & \psi'_1(x_1) & \cdots & \psi'_1(x_n) \\ \psi_2(x_1) & & \psi_2(x_n) & \psi'_2(x_1) & & \psi'_2(x_n) \\ \vdots & & \vdots & \vdots & & \vdots \\ \psi_{2n}(x_1) & & \psi_{2n}(x_n) & \psi'_{2n}(x_1) & & \psi'_{2n}(x_n) \end{vmatrix} \neq 0 \tag{8}$$

for any set of  $n$  points  $x_1, \dots, x_n \in [a, b]$  ( $x_i \neq x_j$  for  $i \neq j$ ).

Such a system will be referred to as an *extended Hermite (EH) system*.

The procedure given in [18] requires the construction of the functions

$$\xi_i(x) = \sum_{j=1}^{2n} \alpha_{ij} \psi_j(x), \quad \eta_i(x) = \sum_{j=1}^{2n} \beta_{ij} \psi_j(x) \quad (i = 1, \dots, n) \tag{9}$$

such that

$$\begin{cases} \xi_i(x_k) = 0, \\ \xi'_i(x_k) = \delta_{ik}, \end{cases} \quad \begin{cases} \eta_i(x_k) = \delta_{ik}, \\ \eta'_i(x_k) = 0, \end{cases} \tag{10}$$

for all  $i = 1, \dots, n$  and all  $k = 1, \dots, n$ . The algorithm is ill conditioned (see [18, Remark 6.2]). In order to obtain the double precision results (REAL\*8), the authors performed the computations in extended precision (Q-arithmetic, i.e., REAL\*16) for generating Gaussian quadratures up to order 20, and in MATHEMATICA (120-digit operations) for generating Gaussian quadratures of higher orders ( $n \leq 40$ ).

In [22], Milovanović and Cvetković proposed a quite different numerical algorithm which is numerically stable and simpler than the previous one. It performs calculations in double precision arithmetic to get double precision results. Otherwise, the generalized Gaussian quadratures for Müntz systems goes back to Stieltjes's paper [29] of 1884. Taking  $\psi_k(x) = x^{\lambda_k}$  on  $[a, b] = [0, 1]$ , where  $0 \leq \lambda_1 < \lambda_2 < \dots$ , he showed the existence of Gaussian formulae. In his short note he considered also Gauss–Radau formulae.

The quadrature formulas (7) possess several properties of the classical Gaussian formulae (for polynomial systems), such as positivity of the weights, rapid convergence, etc. They can be applied to the

wide class of functions, including smooth functions, as well as functions with end-point singularities, such as those in boundary-contact value problems, integral equations, complex analysis, potential theory, and several other fields.

#### 4. Generalized weighted Gaussian formulae for functions with a logarithmic singularity

##### 4.1. Quadrature rules for integrals in boundary element method

Quadrature formulas play a very important role in numerical implementation of the boundary element method (BEM), especially for higher order elements (see [16, Chapters 4 & 5] and [1, Chapter 6]). For calculating integrals of the corresponding influence coefficients (for off-diagonal elements and diagonal elements), quadratures of Gaussian type are very appropriate. In particular, for sufficiently smooth functions on a finite interval  $[a, b]$ , an application of Gauss-Legendre quadrature formula (after a linear transformation to the standard interval  $[-1, 1]$ ) provides numerical integration with a satisfactory accuracy. However, for integrals with a logarithmic singularity the convergence of the corresponding quadrature process is very slow, so that some weighted quadratures of Gaussian type are recommended, e.g.,

$$\int_0^1 f(x) \log \frac{1}{x} dx \approx \sum_{k=1}^n w_k^L f(x_k^L). \tag{11}$$

In such kind of Gaussian quadratures, the weight functions include these “difficult parts (with singularities)” of the integrand. Integrals with nearby and strong singularities was recently considered by Tsamasphyros and Theotokoglou [30].

Regarding the previous fact, it would be very useful to have some kind of quadrature rules suited for functions with and without logarithmic singularities. In other words, such universal (direct) quadrature formulae need to be able to integrate functions with a sufficient accuracy, regardless of whether they are smooth, or contain a logarithmic singularity. This would avoid the separation of a function into singular and non-singular parts, as well as use two different quadrature formulas.

An approach for constructing such universal quadrature formulae which integrate both kind of functions, smooth and ones with a logarithmic singularity, was considered by Nahlik and Białeckı [25]. They assumed that the integrand behaves as a logarithm near zero, i.e.,

$$f(x) = C_1 \log |x| + C_2 + C_3x + C_4x^2 + \dots \quad (C_k \text{ are constants}),$$

and constructed symmetric interpolatory quadrature formulas of the form

$$\int_{-1}^1 f(x) dx = \sum_{k=1}^{2m} A_k f(x_k) + R_n(f), \tag{12}$$

taking zeros of

$$g_4(x) = \frac{1}{4} \sqrt{\frac{5}{2}} (-2 \log |x| + 9x^2 - 5), \quad g_6(x) = \frac{3}{32 \sqrt{2}} (12 \log |x| + 175x^4 - 180x^2 + 37),$$

$$g_8(x) = \frac{1}{64} \sqrt{\frac{13}{2}} (-20 \log |x| + 1078x^6 - 1575x^4 + 630x^2 - 69),$$

as nodes in (12), for  $m = 2, 3$ , and  $4$ , respectively. The functions  $\{g_\nu\}_{\nu=1}^8$  are orthonormal with respect to the usual inner product  $(f, g) = \int_{-1}^1 f(x)g(x) dx$ , obtained from  $\{\log |x|, 1, x, x^2, x^3, x^4, x^5, x^6\}$  by the standard Gram–Schmidt procedure. These zeros are real, distinct and are symmetrically distributed in  $(-1, 1)$  (see graphics of functions  $g_4, g_6$ , and  $g_8$  in Figure 1), but the obtained quadratures are not of Gaussian type.

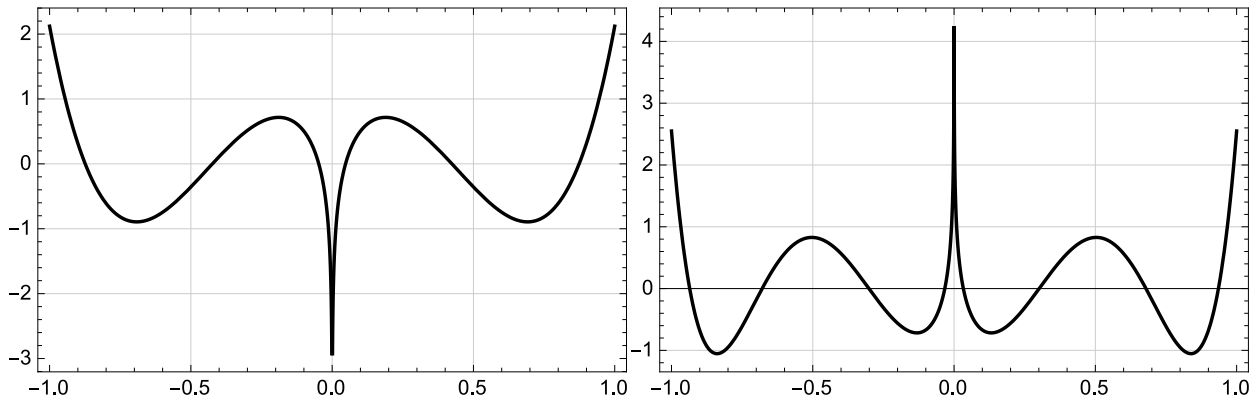


Figure 1: Graphics of orthonormal functions  $g_6$  (left) and  $g_8$  (right)

Another approach for weakly singular logarithmic integrals, which appear in two-dimensional BEM problems, was considered by Smith [28]. He discussed some direct Gaussian rules for logarithmic singularities on isoparametric (quadratic and cubic) elements.

Recently, Milovanović, Igić an Turnić [24] have developed an efficient method for constructing a class of generalized quadrature formulae of Gaussian type on  $(-1, 1)$  for integrals  $\int_{-1}^1 f(x) dx$ , with logarithmic singularities. Also, they have presented several special cases for such  $n$ -point quadratures, which are exact on the both of spaces  $\mathcal{P}_{2n-2\ell-1}[-1, 1]$  (the space of algebraic polynomials of degree at most  $2n - 2\ell - 1$ ) and  $\mathcal{L}_{2\ell-1}[-1, 1] = \text{span}\{x^k \log|x|\}_{k=0}^{2\ell-1}$  (the logarithmic space), where  $1 \leq \ell \leq n$ . The construction of such quadratures is based on solving systems of nonlinear equations, using orthogonal system of basis functions. The last provides the well conditioned matrices in the corresponding iterative procedure.

In the sequel we consider  $(2m)$ -point symmetric quadrature rules of Gaussian type

$$\int_{-1}^1 f(x)|x|^\gamma(1-x^2)^\alpha \log \frac{1}{|x|} dx = \sum_{k=1}^m A_k^L(f(x_k^L) + f(-x_k^L)) + R_{2m}(w^L; f) \tag{13}$$

and

$$\int_{-1}^1 f(x)|x|^\gamma(1-x^2)^\alpha dx = \sum_{k=1}^m A_k(f(x_k) + f(-x_k)) + R_{2m}(w; f), \tag{14}$$

where  $w(x) = |x|^\gamma(1-x^2)^\alpha$  and  $w^L(x) = |x|^\gamma(1-x^2)^\alpha \log(1/|x|)$ ,  $\alpha, \gamma > -1$ . The first of them will be Gaussian rule with an algebraic degree of exactness  $2(2m) - 1 = 4m - 1$ , and the second one will be exact for all functions from the set

$$\mathcal{A}^L = \{1, x, \dots, x^{2m-1}, \log|x|, x \log|x|, \dots, x^{2m-1} \log|x|\}, \quad \dim \mathcal{A}^L = 4m. \tag{15}$$

Otherwise, these symmetric formulas (13) and (14) are true for any odd function.

#### 4.2. Generalized Gegenbauer orthogonal polynomials on $(-1, 1)$

In the quadrature formulas (13) and (14) we use the generalized Gegenbauer weight function

$$w(x) = |x|^\gamma(1-x^2)^\alpha \quad (\alpha, \gamma > -1). \tag{16}$$

The monic polynomials  $W_\nu^{(\alpha, \beta)}(x)$ ,  $\nu = 0, 1, \dots$ , orthogonal with respect to the weight function (16), where  $\beta = (\gamma - 1)/2$ , were introduced by Lašćenov [17] (see, also, [2, pp. 155–156] and [19, pp. 147–148]) and they

can be expressed in terms of the Jacobi polynomials  $P_v^{(\alpha,\beta)}(x)$ ,  $v = 0, 1, \dots$ , which are orthogonal on  $(-1, 1)$  with respect to the weight function  $w^{(\alpha,\beta)}(x) = (1-x)^\alpha(1+x)^\beta$ ,  $\alpha, \beta > -1$ . Thus, we have

$$W_{2k}^{(\alpha,\beta)}(x) = \frac{k!}{(k + \alpha + \beta + 1)_k} P_k^{(\alpha,\beta)}(2x^2 - 1), \quad W_{2k+1}^{(\alpha,\beta)}(x) = \frac{k!}{(k + \alpha + \beta + 2)_k} x P_k^{(\alpha,\beta+1)}(2x^2 - 1). \quad (17)$$

We can also see that  $W_{2k+1}^{(\alpha,\beta)}(x) = xW_{2k}^{(\alpha,\beta+1)}(x)$ . These orthogonal polynomials satisfy the following three-term recurrence relation

$$\begin{aligned} W_{v+1}^{(\alpha,\beta)}(x) &= xW_v^{(\alpha,\beta)}(x) - \beta_v W_{v-1}^{(\alpha,\beta)}(x), \quad v = 0, 1, \dots, \\ W_{-1}^{(\alpha,\beta)}(x) &= 0, \quad W_0^{(\alpha,\beta)}(x) = 1, \end{aligned}$$

with recursive coefficients

$$\beta_{2k} = \frac{k(k + \alpha)}{(2k + \alpha + \beta)(2k + \alpha + \beta + 1)}, \quad \beta_{2k-1} = \frac{(k + \beta)(k + \alpha + \beta)}{(2k + \alpha + \beta - 1)(2k + \alpha + \beta)}, \quad k = 1, 2, \dots,$$

except when  $\alpha + \beta = -1$ ; then  $\beta_1 = (\beta + 1)/(\alpha + \beta + 2)$ . It is convenient to put  $\beta_0 = \int_{-1}^1 |x|^\nu (1-x^2)^\alpha dx = B(\alpha + 1, \beta + 1)$ , where  $B(\alpha, \beta) = \Gamma(\alpha)\Gamma(\beta)/\Gamma(\alpha + \beta)$  is the beta function.

#### 4.3. Some basic facts on the quadrature rule (14)

We study the generalized Gaussian quadrature rule (14), which is exact for all functions from  $\mathcal{A}^L$ . The corresponding orthonormal system of functions

$$\{\psi_1(x), \psi_2(x), \dots, \psi_{4m}(x)\}, \quad (18)$$

can be obtained from (15) by some of orthogonalization processes, for example, by the Gram–Schmidt procedure, so that

$$(\psi_\nu, \psi_\mu) = \int_{-1}^1 \psi_\nu(x)\psi_\mu(x)w(x) dx = 0 \quad (\nu \neq \mu), \quad \|\psi_\nu\|^2 = (\psi_\nu, \psi_\nu) = 1,$$

where  $w$  is the generalized Gegenbauer weight function given by (16).

Notice that the first  $2m$  functions in (18) are algebraic polynomials; in fact, they are orthonormal generalized Gegenbauer polynomials,

$$\psi_k(x) = \frac{W_{k-1}^{(\alpha,\beta)}(x)}{\|W_{k-1}^{(\alpha,\beta)}\|}, \quad k = 1, \dots, 2m.$$

Also, we note that  $\{\psi_{2\nu-1}\}_{\nu=1}^{2m}$  are even, and  $\{\psi_{2\nu}\}_{\nu=1}^{2m}$  are odd functions.

As we mentioned before, the elegant tools (like orthogonal polynomials) do not exist for non-polynomial basis systems of functions and related Gaussian quadratures. Thus, in construction of such symmetric rules we need to solve the following system of nonlinear equations

$$\sum_{k=1}^m A_k \psi_{2\nu-1}(x_k) = \int_0^1 \psi_{2\nu-1}(x)w(x) dx, \quad \nu = 1, \dots, 2m. \quad (19)$$

taking only even basis functions  $\{\psi_{2\nu-1}\}_{\nu=1}^{2m}$ . Since  $\psi_1(x) = 1/\sqrt{\beta_0}$ , where  $\beta_0 = B(\alpha + 1, \beta + 1)$ , because of orthogonality, the right-hand side in (19) reduces to

$$\int_0^1 \psi_{2\nu-1}(x)w(x) dx = \begin{cases} \frac{1}{2} \sqrt{\beta_0}, & \nu = 1, \\ 0, & \nu > 1, \end{cases}$$

In the sequel we need an auxiliary result for the sequence of only even functions  $\{\psi_{2\nu-1}\}_{\nu=1}^{2m}$  from (18). This result enables us to get a simpler method for constructing generalized Gaussian quadratures of the mentioned form.

**Lemma 4.1.** Let  $\{\psi_{2\nu-1}\}_{\nu=1}^{2m}$  be a system of only even functions orthogonal with respect to the inner product  $(f, g) = \int_{-1}^1 f(x)g(x)w(x) dx$ . Then the functions  $\varphi_\nu(t) = \psi_{2\nu-1}(\sqrt{t})$ ,  $\nu = 1, \dots, 2m$ , are orthogonal on  $(0, 1)$  in the following sense

$$\langle \varphi_\nu, \varphi_\mu \rangle = \int_0^1 \varphi_\nu(t)\varphi_\mu(t) \frac{w(\sqrt{t})}{\sqrt{t}} dt = 0 \quad (\nu \neq \mu). \tag{20}$$

*Proof.* Let  $\nu \neq \mu$ . Since

$$\int_{-1}^1 \psi_{2\nu-1}(x)\psi_{2\mu-1}(x)w(x) dx = 2 \int_0^1 \psi_{2\nu-1}(x)\psi_{2\mu-1}(x)w(x) dx = 0,$$

we have, by a change of variables  $x^2 = t$ ,

$$\int_0^1 \psi_{2\nu-1}(\sqrt{t})\psi_{2\mu-1}(\sqrt{t}) \frac{w(\sqrt{t})}{\sqrt{t}} dt = 0 \quad (\nu \neq \mu).$$

Thus, the system of functions  $\varphi_\nu(t) = \psi_{2\nu-1}(\sqrt{t})$ ,  $\nu = 1, \dots, 2m$ , satisfies the orthogonality relation (20).  $\square$

Otherwise, the construction of symmetric quadrature rules on  $(-1, 1)$  can be significantly simplified by a transformation from  $(-1, 1)$  to the interval  $(0, 1)$  and we will use it in the rest of this work.

4.4. Two equivalent weighted quadrature rules

Now, we consider  $(2m)$ -point symmetric quadrature with respect to an even weight function  $x \mapsto W(x)$  on  $(-1, 1)$ ,

$$\int_{-1}^1 f(x)W(x) dx = \sum_{k=1}^m A_k(f(x_k) + f(-x_k)) + R_{2m}^{[-1,1]}(f), \tag{21}$$

where  $0 < x_1 < \dots < x_m \leq 1$ . Evidently, such a quadrature is exact for every odd function.

Now, we want to transform this quadrature (21) to the interval  $(0, 1)$ . Namely, because of  $W(-x) = W(x)$ , we have

$$\int_{-1}^1 f(x)W(x) dx = \int_0^1 (f(x) + f(-x))W(x) dx,$$

and then, by changing variables  $x = \sqrt{t}$ , we obtain

$$\int_{-1}^1 f(x)W(x) dx = \int_0^1 (f(\sqrt{t}) + f(-\sqrt{t})) \frac{W(\sqrt{t})}{\sqrt{t}} dt = \int_0^1 g(t) \frac{W(\sqrt{t})}{\sqrt{t}} dt,$$

where the function  $g$  is defined by

$$g(t) = \frac{1}{2} (f(\sqrt{t}) + f(-\sqrt{t})).$$

Then, the right-hand side of (21) reduces to

$$\sum_{k=1}^m A_k(f(x_k) + f(-x_k)) + R_{2m}^{[-1,1]}(f) = \sum_{k=1}^m B_k g(\tau_k) + R_m^{[0,1]}(g), \tag{22}$$

where we put

$$B_k = 2A_k, \quad \tau_k = x_k^2, \quad k = 1, \dots, m, \tag{23}$$



and  $R_m^{[0,1]}(g) = R_{2m}^{[-1,1]}(f)$ , so that the right-hand side in (22) corresponds to the transformed integral, i.e.,

$$\int_0^1 g(t) \frac{W(\sqrt{t})}{\sqrt{t}} dt = \sum_{k=1}^m B_k g(\tau_k) + R_m^{[0,1]}(g). \tag{24}$$

In this way, we have proved the following result:

**Lemma 4.2.** *Under conditions (23), the weighted quadrature formulas (21) and (24) are equivalent.*

In the sequel we use this lemma in construction of quadrature formulas on  $(-1, 1)$ , with  $2m$  nodes, using a construction of the corresponding rules on  $(0, 1)$ , with only  $m$  nodes.

### 5. Construction of Gaussian quadrature rule (12)

In order to construct the quadrature formula (13), i.e.,

$$\int_{-1}^1 f(x) |x|^\gamma (1-x^2)^\alpha \log \frac{1}{|x|} dx = \sum_{k=1}^m A_k^L (f(x_k^L) + f(-x_k^L)) + R_{2m}(w^L; f),$$

we start with Lemma 4.2 and construct first the quadrature rule (24) with  $m$  nodes. In this case (24) should be a weighted quadrature rule of Gaussian type, with respect to the weight function

$$w_0(t) = \frac{W(\sqrt{t})}{\sqrt{t}} = \frac{1}{2} t^\beta (1-t)^\alpha \log \frac{1}{t}, \quad \beta = \frac{\gamma-1}{2},$$

because

$$W(x) = |x|^\gamma (1-x^2)^\alpha \log \frac{1}{|x|} = \frac{1}{2} |x|^\gamma (1-x^2)^\alpha \log \frac{1}{|x|^2}.$$

The moments of this weight function  $m_k = \int_0^1 t^k w_0(t) dt$  can be exactly calculated in terms of the so-called digamma function (i.e., the logarithmic derivative of the gamma function)  $\psi(z) = \Gamma'(z)/\Gamma(z)$ . So, we have (cf. [26, p. 490])

$$m_k = -\frac{1}{2} \int_0^1 t^{k+\beta} (1-t)^\alpha \log t dt = \frac{1}{2} B(\alpha+1, \beta+k+1) [\psi(\alpha+\beta+k+2) - \psi(\beta+k+1)], \quad k \geq 0,$$

which is enough for finding recursion coefficients in the three-term recurrence relation for polynomials orthogonal with respect to the weight function  $w_0$  on  $(0, 1)$ , as well as for the nodes  $\tau_k$  and the weights  $B_k, k = 1, \dots, m$ , in the Gaussian quadrature formula (24), by our MATHEMATICA package `OrthogonalPolynomials` (see [4] and [23]), because MATHEMATICA evaluates  $\psi(z)$  to arbitrary numerical precision, using the function `PolyGamma[z]`. Then, by (23) we obtain the parameters  $x_k^L$  and  $A_k^L$  in the rule (13).

For example, using our MATHEMATICA package `OrthogonalPolynomials` (see [4] and [23]) and executing the following commands:

```
<< orthogonalPolynomials'
mom =Table[Gamma[al+1]Gamma[be+k+1]/(2Gamma[al+be+k+2]) (PoyGamma[al+be+k+2]
-PolyGamma[be+k+1]), {k,0,39}]/. {al -> 0, be -> -1/2};
{alpha, beta} = aChebyshevAlgorithm[mom, WorkingPrecision -> 35];
{alpha1, beta1} = aChebyshevAlgorithm[mom, WorkingPrecision -> 50];
N[Max[Abs[alpha1/alpha-1], Abs[beta/beta1-1]], 3]
```

we obtain the first 20 recurrence coefficients (for  $\alpha = 0$  and  $\beta = -1/2$  in the weight  $w_0$ ) with the maximal relative error  $5.50 \times 10^{-30}$ , using the working precision of 35 decimal digits. Notice that for calculating this maximal relative error in recursive coefficients we have to compute them with some better precision (in this case we used 50 decimal digits).

Now, we can calculate Gaussian parameters (nodes and weights) in (24) for each  $m \leq 20$ , as well as the parameters in the rule (13). For example, for  $m = 10$  we print the parameters  $x_k^L$  and  $A_k^L$  in the rule (13), with 20 decimal digits.

```
PGQ[m_] := aGaussianNodesWeights[m, alpha, beta, WorkingPrecision -> 30, Precision -> 25]
{tau, B} = PGQ[10]; {xL, AL} = {Sqrt[tau], B/2};
Print["nodes = ", N[xL, 20]]; Print["weights = ", N[AL, 20]];
```

```
nodes = {0.058684713389643455348, 0.20085590974338672323, 0.34102346665699661535,
0.47499755685218773898, 0.59933105712959326204, 0.71098630175917040439,
0.80729488074114865580, 0.88597995933835106670, 0.94519165897723331249,
0.98353845345472765660}
```

```
weights = {0.40077930960551466949, 0.22788836401795107375, 0.14812360023344707567,
0.096547816551015655829, 0.060649749122039659929, 0.035609402612321070364,
0.018803993942658554996, 0.0083797607340355934749, 0.0027604277995848179492,
0.00045757538143182854107}
```

Similarly, for  $\alpha = \beta = -1/2$  and  $m = 5$ , we get

```
nodes = {0.11002372481701129812, 0.38352164350631725367, 0.63216422494287654400,
0.82962888620377180320, 0.95635128465561669026}
```

```
weights = {0.61005071778693605514, 0.27611428447805086073, 0.13427676855303110178,
0.055126965252488246396, 0.013224309081294801204}
```

**Example 5.1.** For functions  $x \mapsto f_1(x) = \cos(10x)$  and  $x \mapsto f_2(x) = \cos(50x^2)$  we consider the integrals

$$I(f_1) = \int_{-1}^1 \cos(10x) \log \frac{1}{|x|} dx = \frac{1}{5} \text{Si}(10) = 0.3316695188437748098661943758779 \dots$$

and

$$I(f_2) = \int_{-1}^1 \cos(50x^2) \log \frac{1}{|x|} dx = \frac{1}{5} \left( 2000 {}_2F_3 \left( \frac{5}{4}, \frac{5}{4}; \frac{3}{2}, \frac{9}{4}, \frac{9}{4}; -625 \right) + \sqrt{\pi} C \left( \frac{10}{\sqrt{\pi}} \right) \right) \\ = 0.659723294638846952127061737394 \dots,$$

where

$$\text{Si}(z) = \int_0^z \frac{\sin t}{t} dt \quad \text{and} \quad C(z) = \int_0^z \cos \left( \frac{\pi t^2}{2} \right) dt$$

are the sine integral function and the Fresnel integral, respectively, and  ${}_2F_3$  is the hypergeometric function, defined by

$${}_2F_3(a_1, a_2; b_1, b_2, b_3; z) = \sum_{k=0}^{+\infty} \frac{(a_1)_k (a_2)_k}{(b_1)_k (b_2)_k (b_3)_k} \cdot \frac{z^k}{k!},$$

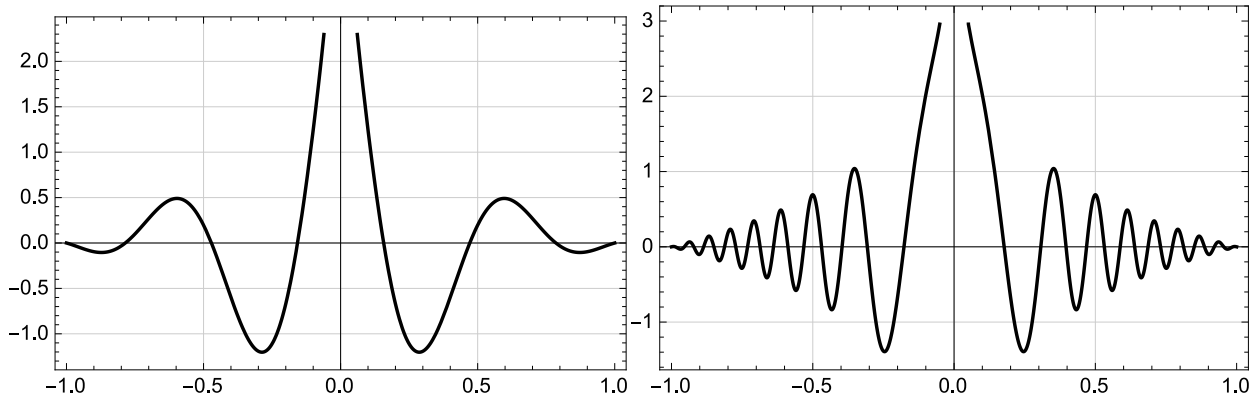


Figure 2: Graphics of the integrands  $x \mapsto \cos(10x) \log(1/|x|)$  (left) and  $x \mapsto \cos(50x^2) \log(1/|x|)$  (right)

Table 1: Quadrature sums  $Q_{2m}(f_j)$  and their relative errors for integrals  $I(f_j)$ ,  $j = 1, 2$

$m$	$Q_{2m}(f_1)$	$err_{2m}(f_1)$	$m$	$Q_{2m}(f_2)$	$err_{2m}(f_2)$
4	0.33009775491530849648801487374	4.74(-3)	12	0.857128301	2.99(-1)
6	0.33166949067521330644013062255	8.49(-8)	14	0.674875330	2.30(-2)
8	0.33166951884374388926728589986	9.32(-14)	16	0.659977439	3.58(-4)
10	0.33166951884377480986183339753	1.31(-20)	18	0.659716096	1.09(-5)
12	0.33166951884377480986619437576	3.70(-28)	20	0.659722976	4.83(-7)

where  $(a)_k$  is the Pochhammer symbol,  $(a)_k = a(a + 1) \cdots (a + k - 1) = \Gamma(a + k)/\Gamma(a)$ . Integrands in  $I(f_1)$  and  $I(f_2)$  are presented in Figure 2.

In order to calculate these integrals we use the quadrature rule (13) (with the parameters  $\alpha = 0$  and  $\beta = -1/2$ ). In Table 1 we present quadrature approximations and the corresponding relative errors,

$$Q_{2m}(f) = \sum_{k=1}^m A_k^L (f(x_k^L) + f(-x_k^L)), \quad err_{2m}(f) = \left| \frac{Q_{2m}(f) - I(f)}{I(f)} \right|.$$

Digits in error are underlined, and numbers in parenthesis indicate the decimal exponents.

As we can see, in the case of  $I(f_1)$ , the convergence of the quadrature rule (13) is very fast.

### 6. Construction of Gaussian quadrature rule (13)

We return now to construction of a direct (universal) quadrature of Gaussian type (14), i.e.,

$$\int_{-1}^1 f(x) |x|^\gamma (1 - x^2)^\alpha dx = \sum_{k=1}^m A_k (f(x_k) + f(-x_k)) + R_{2m}(w; f), \tag{25}$$

where  $w(x) = |x|^\gamma (1 - x^2)^\alpha$ ,  $\alpha, \gamma > -1$ . As we have seen earlier, a nonlinear system of equations for finding nodes  $x_k$  and weight coefficients  $A_k$ ,  $k = 1, \dots, m$ , is given by (19). Using Lemmas 4.1 and 4.2, this system of equations becomes

$$\sum_{k=1}^m B_k \varphi_\nu(\tau_k) = \sqrt{\beta_0} \delta_{\nu,1}, \quad \nu = 1, \dots, 2m, \tag{26}$$

where  $\{\varphi_\nu\}_{\nu=1}^{2m}$  is a system of orthogonal functions on  $(0, 1)$  with respect to the weight function  $t \mapsto w(\sqrt{t})/\sqrt{t} = t^\beta (1 - t)^\alpha$ ,  $\beta = (\gamma - 1)/2$ , and  $\delta_{\nu,1}$  is the Kronecker’s delta. Evidently, this system of equations gives a characterization for the quadrature formula (24) to be Gaussian on  $(0, 1)$ .

In our construction of orthonormal system of functions  $\{\varphi_1, \varphi_2, \dots, \varphi_{2m}\}$ , we use an orthogonalization process described in [19, pp. 75–77], starting from the system of  $2m$  linearly independent functions  $U = \{1, t, \dots, t^{m-1}, \log t, t \log t, \dots, t^{m-1} \log t\}$ . Then, for  $k \in \mathbb{N}_0$  and  $\alpha, \beta > -1$ , we need an integral of the form

$$\int_0^1 t^{\beta+k}(1-t)^\alpha dt = B(\alpha+1, \beta+k+1) = \frac{\Gamma(\alpha+1)\Gamma(\beta+k+1)}{\Gamma(\alpha+\beta+k+2)},$$

as well as the following integrals (cf. [26, p. 490])

$$\int_0^1 t^{\beta+k}(1-t)^\alpha \log t dt = B(\alpha+1, \beta+k+1)[\psi(\beta+k+1) - \psi(\alpha+\beta+k+2)]$$

and

$$\int_0^1 t^{\beta+k}(1-t)^\alpha \log^2 t dt = B(\alpha+1, \beta+k+1)\{[\psi(\beta+k+1) - \psi(\alpha+\beta+k+2)]^2 + \psi'(\beta+k+1) - \psi'(\alpha+\beta+k+2)\}.$$

We mention that MATHEMATICA evaluates derivatives  $\psi^{(n)}(z)$  to arbitrary numerical precision, using the function PolyGamma[n, z]. Otherwise, the first  $m$  functions of this orthogonal system are, in fact, the orthonormal polynomials (see Subsections 4.2 and 4.3)

$$\varphi_{k+1}(t) = \sqrt{\frac{k!\Gamma(k+\alpha+\beta+1)}{\Gamma(2k+\alpha+\beta+1)}} \cdot \frac{P_k^{(\alpha,\beta)}(2t-1)}{\sqrt{B(k+\alpha+1, k+\beta+1)}}, \quad k = 0, 1, \dots, m-1.$$

Thus, our method for constructing quadrature rules of the form (24) is based on solving system of nonlinear equations (26), i.e.,

$$f_v(\mathbf{x}) = f_v(\mathbf{B}, \mathbf{t}) = B_1\varphi_v(\tau_1) + B_2\varphi_v(\tau_2) + \dots + B_m\varphi_v(\tau_m) = \sqrt{\beta_0} \delta_{v,1}, \quad v = 1, \dots, 2m, \tag{27}$$

by the standard Newton–Kantorovich method. Here,  $\mathbf{x}$  is  $(2m)$ -dimensional vector, which coordinates are weights and nodes in the quadrature rule (24), i.e.,  $\mathbf{x} = [B_1 \dots B_m \tau_1 \dots \tau_m]^T$ . Also, we introduce  $m$ -dimensional vectors  $\mathbf{B} = [B_1 \dots B_m]^T$  and  $\mathbf{t} = [\tau_1 \dots \tau_m]^T$ .

The Jacobian for the system of equations (27) can be easily calculated in the form

$$J(\mathbf{x}) = \begin{bmatrix} \frac{\partial f_1}{\partial B_1} & \dots & \frac{\partial f_1}{\partial B_m} & \frac{\partial f_1}{\partial \tau_1} & \dots & \frac{\partial f_1}{\partial \tau_m} \\ \frac{\partial f_2}{\partial B_1} & \dots & \frac{\partial f_2}{\partial B_m} & \frac{\partial f_2}{\partial \tau_1} & \dots & \frac{\partial f_2}{\partial \tau_m} \\ \vdots & & & & & \\ \frac{\partial f_{2m}}{\partial B_1} & \dots & \frac{\partial f_{2m}}{\partial B_m} & \frac{\partial f_{2m}}{\partial \tau_1} & \dots & \frac{\partial f_{2m}}{\partial \tau_m} \end{bmatrix} = \begin{bmatrix} U_m(\mathbf{t}) & Y_m(\mathbf{B}, \mathbf{t}) \\ V_m(\mathbf{t}) & Z_m(\mathbf{B}, \mathbf{t}) \end{bmatrix} = W_{2m}(\mathbf{B}, \mathbf{t}), \tag{28}$$

where the matrices  $U_m = U_m(\mathbf{t})$ ,  $V_m = V_m(\mathbf{t})$ ,  $Y_m = Y_m(\mathbf{B}, \mathbf{t})$ , and  $Z_m = Z_m(\mathbf{B}, \mathbf{t})$  are given by

$$U_m = \begin{bmatrix} \varphi_1(\tau_1) & \varphi_1(\tau_2) & \dots & \varphi_1(\tau_m) \\ \varphi_2(\tau_1) & \varphi_2(\tau_2) & & \varphi_2(\tau_m) \\ \vdots & & & \\ \varphi_m(\tau_1) & \varphi_m(\tau_2) & & \varphi_m(\tau_m) \end{bmatrix}, \quad V_m = \begin{bmatrix} \varphi_{m+1}(\tau_1) & \varphi_{m+1}(\tau_2) & \dots & \varphi_{m+1}(\tau_m) \\ \varphi_{m+2}(\tau_1) & \varphi_{m+2}(\tau_2) & & \varphi_{m+2}(\tau_m) \\ \vdots & & & \\ \varphi_{2m}(\tau_1) & \varphi_{2m}(\tau_2) & & \varphi_{2m}(\tau_m) \end{bmatrix},$$

$$Y_m = \begin{bmatrix} B_1\varphi'_1(\tau_1) & B_2\varphi'_1(\tau_2) & \dots & B_m\varphi'_1(\tau_m) \\ B_1\varphi'_2(\tau_1) & B_2\varphi'_2(\tau_2) & & B_m\varphi'_2(\tau_m) \\ \vdots & & & \\ B_1\varphi'_m(\tau_1) & B_2\varphi'_m(\tau_2) & & B_m\varphi'_m(\tau_m) \end{bmatrix}, \quad Z_m = \begin{bmatrix} B_1\varphi'_{m+1}(\tau_1) & B_2\varphi'_{m+1}(\tau_2) & \dots & B_m\varphi'_{m+1}(\tau_m) \\ B_1\varphi'_{m+2}(\tau_1) & B_2\varphi'_{m+2}(\tau_2) & & B_m\varphi'_{m+2}(\tau_m) \\ \vdots & & & \\ B_1\varphi'_{2m}(\tau_1) & B_2\varphi'_{2m}(\tau_2) & & B_m\varphi'_{2m}(\tau_m) \end{bmatrix}.$$

In matrix notation, the system of equations (27) has the following form

$$\mathbf{f}(\mathbf{x}) = \Phi(\mathbf{B}, \mathbf{t}) = \begin{bmatrix} U_m(\mathbf{t})\mathbf{B} - \sqrt{\beta_0}\mathbf{e}_1 \\ V_m(\mathbf{t})\mathbf{B} \end{bmatrix} = \mathbf{0}, \tag{29}$$

where  $\mathbf{e}_1$  is the first coordinate  $m$ -dimensional vector,  $\mathbf{e}_1 = [1 \ 0 \ \dots \ 0]^T$ .

Following [22] let  $\Delta\mathbf{B} = \mathbf{B} - \hat{\mathbf{B}}$  and  $\Delta\mathbf{t} = \mathbf{t} - \hat{\mathbf{t}}$ , where  $(\hat{\mathbf{B}}, \hat{\mathbf{t}})$  is the unique solution of the nonlinear system of equations (29). Then, we can formulate the Newton–Kantorovich method, using the following linearized system of equations obtained by truncating the Taylor expansion at the point  $(\mathbf{B}, \mathbf{t})$ ,

$$U_m\Delta\mathbf{B} + Y_m\Delta\mathbf{t} = U_m\mathbf{B} - \sqrt{\beta_0}\mathbf{e}_1,$$

$$V_m\Delta\mathbf{B} + Z_m\Delta\mathbf{t} = V_m\mathbf{B},$$

or in the matrix form

$$W_{2m} \begin{bmatrix} \Delta\mathbf{B} \\ \Delta\mathbf{t} \end{bmatrix} = \begin{bmatrix} U_m\mathbf{B} - \sqrt{\beta_0}\mathbf{e}_1 \\ V_m\mathbf{B} \end{bmatrix}. \tag{30}$$

Taking the solution  $(\hat{\mathbf{B}}, \hat{\mathbf{t}})$  of (30) to be the next approximation, our iterative process becomes

$$\left. \begin{aligned} \hat{\mathbf{t}} &= \mathbf{t} + \sqrt{\beta_0} \left[ V_m U_m^{-1} Y_m - Z_m \right]^{-1} V_m U_m^{-1} \mathbf{e}_1 \\ \hat{\mathbf{B}} &= U_m^{-1} \left[ Y_m (\mathbf{t} - \hat{\mathbf{t}}) + \sqrt{\beta_0} \mathbf{e}_1 \right] \end{aligned} \right\}. \tag{31}$$

We can prove that this process is quadratic convergent, providing sufficiently good starting values.

For solving the equation  $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ , with a continuously differentiable mapping  $\mathbf{f}$ , it is well known (cf. [11, pp. 284–287]) that there exists some  $\varepsilon > 0$ , such that for each  $\mathbf{x}_0$  with the property  $\|\mathbf{x}_0 - \hat{\mathbf{x}}\| \leq \varepsilon$ , the sequence  $\{\mathbf{x}_k\}_{k=0}^{+\infty}$ , generated by the Newton-Kantorovich method

$$\mathbf{x}_{k+1} = \mathbf{x}_k - (J(\mathbf{x}_k))^{-1} \mathbf{f}(\mathbf{x}_k), \quad k = 0, 1, \dots,$$

tends to the solution  $\hat{\mathbf{x}}$  with a quadratic convergence  $\|\mathbf{x}_{k+1} - \hat{\mathbf{x}}\| = O(\|\mathbf{x}_k - \hat{\mathbf{x}}\|^2)$ . In our case we have the following result:

**Lemma 6.1.** *For sufficiently good starting values, which can be provided by the method of continuation, the presented form of the Newton-Kantorovich method (31) is quadratic convergent.*

*Proof.* Since the system of functions  $\{\varphi_\nu\}_{\nu=1}^{2m}$  is an extended Hermite (EH) system and the weights  $B_k$ ,  $k = 1, \dots, m$ , are positive, from (28) and (8) we see that

$$\det W_{2m}(\hat{\mathbf{B}}, \hat{\mathbf{t}}) = (B_1 \cdots B_m) D_{2m}(\varphi_1, \varphi_1, \dots, \varphi_{2m}) \neq 0,$$

and therefore the Jacobian has full rank. Also, the system  $\{\varphi_\nu\}_{\nu=1}^m$  is Chebyshev, and therefore the matrix  $U_m$  is invertible. Thus, it is enough to apply the theorem on the matrix inversion in the block form presented in [13, p. 13] and [31, p. 201] to finish the proof.  $\square$

**Remark 6.2.** For numerical applications of weighted-residual finite element methods, J.A. Crow [3] presented a quadrature rule  $\int_0^1 g(t) dt = \sum_{k=1}^m w_k g(x_k) + R_m(g)$ , which is exact for integrands of the form  $t \mapsto p(t) + q(t) \log t$ , where  $p$  and  $q$  are polynomials of degree at most  $m - 1$ . In his construction, a selection of power functions on  $[0, 1]$  (monomials) as a natural basis leads to a very ill-conditioned Vandermonde system of equations. For example, in the case  $m = 5$ , condition numbers of this matrix during the iterative process varied between  $10^8$  and  $10^{15}$ , which means a loss of about 8 to 15 decimal digits in computations. Because of that the author used the shifted Legendre polynomials on  $[0, 1]$  as a better selection of basis functions, as well as Smith’s FM multiple precision software [27] to perform all computations using about 50 significant decimal digits.

In our method, in order to get well-conditioned matrices during the numerical procedure, we use the orthonormal system as a basis of functions, so that the method is stable. The main problem with the NewtonKantorovich method is how to provide sufficiently good starting values. Our strategy in the construction is based on the method of continuation, starting from the corresponding standard Gaussian formula (with a polynomial degree of exactness). The convergence is quadratic.

Table 2: Quadrature parameters in (25) for  $\alpha = \beta = -1/2$  and  $m = 2(1)5$

$m$	$k$	$x_k$	$A_k$
2	1	1.5687932220133083737(-1)	4.9023770831392127704(-1)
	2	8.4597799643685904352(-1)	1.0805586184809753422
3	1	7.4330911998429636136(-2)	2.3536348707782220590(-1)
	2	4.9067720343094899076(-1)	5.9474998941244362493(-1)
	3	9.3020808634535177637(-1)	7.4068285030463078840(-1)
4	1	4.3043445929275625351(-2)	1.3725763004144154931(-1)
	2	3.0329580016011919663(-1)	3.7180716249909688774(-1)
	3	6.8411187359652082596(-1)	5.0325894862261033029(-1)
	4	9.6067483342440115801(-1)	5.5847258563174785190(-1)
5	1	2.7998919084010510299(-2)	8.9632942674696156504(-2)
	2	2.0329461303212981917(-1)	2.5274310616604007523(-1)
	3	4.9319306890293015213(-1)	3.6032635832736034825(-1)
	4	7.8928360355099509366(-1)	4.2078564339874673998(-1)
	5	9.7486024601950383507(-1)	4.4730827622805329928(-1)

Nodes and weights in the quadrature formula (25) for  $\alpha = \beta = -1/2$  ( $\gamma = 0$ ) and  $m = 2, 3, 4$ , and 5 are presented in Table 2. For  $m = 1$  it is easy to find

$$x_1(\alpha, \beta) = \exp\left(\frac{1}{2}(\psi(\beta + 1) - \psi(\alpha + \beta + 2))\right), \quad A_1(\alpha, \beta) = \frac{1}{2} B(\alpha + 1, \beta + 1).$$

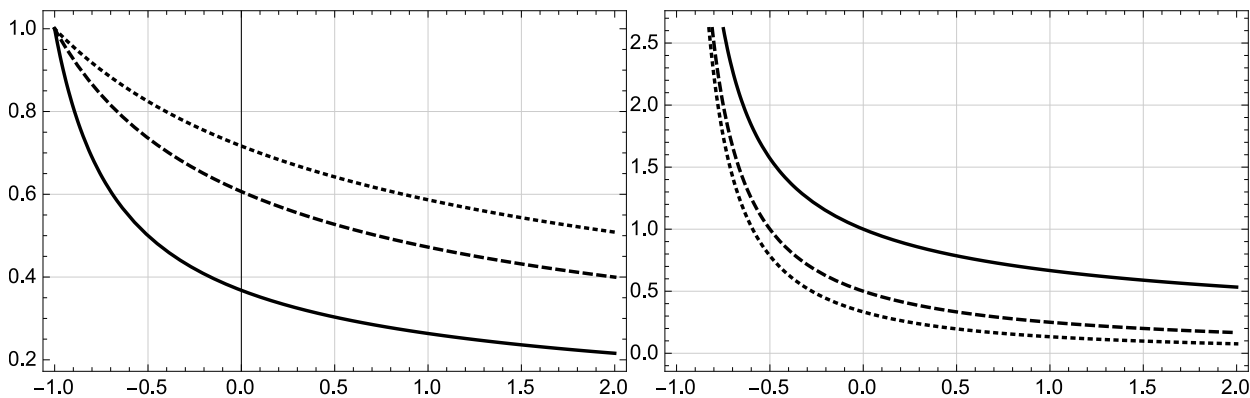


Figure 3: The functions  $\alpha \mapsto x_1(\alpha, \beta)$  (left) and  $\alpha \mapsto A_1(\alpha, \beta)$  (right) in quadrature rule (25) for  $m = 1$ , when  $\beta = -1/2$  (solid line),  $\beta = 0$  (dashed line), and  $\beta = 1/2$  (dotted line)

The behaviour of  $x_1(\alpha, \beta)$  and  $A_1(\alpha, \beta)$  for  $-1 < \alpha < 2$  and some selected values of  $\beta$  are presented in Figure 3. For example, for  $\alpha = \beta = -1/2$  we have  $x_1 = 1/2$  and  $A_1 = \pi/2$ .

**Example 6.3.** We consider the integral

$$I = \int_{-1}^1 \frac{e^x \log x^2 + \cos x}{\sqrt{1-x^2}} dx = -2.2656196675547437914 \dots$$

Applying the quadrature rule (25) to this integral we obtain approximations

$$Q_{2m} = \sum_{k=1}^m A_k (f(x_k) + f(-x_k)),$$

with relative errors  $err_{2m}$  presented in Table 3. Digits in error are underlined.

Table 3: Quadrature sums  $Q_{2m}$  and their relative errors for the rule (25), as well as relative errors for the Gauss-Legendre and Gauss-Chebyshev rules

$m$	$Q_{2m}$	$err_{2m}$	$2m$	$err_{2m}^L$	$err_{2m}^{Ch}$
1	-2. <u>1539983044439</u>	4.93(-2)	20	7.34(-2)	9.60(-2)
2	-2. <u>2732059700768</u>	3.35(-3)	40	3.72(-2)	4.80(-2)
3	-2. <u>2656304863533</u>	4.78(-6)	60	2.49(-2)	3.20(-2)
4	-2. <u>2656197046209</u>	1.64(-8)	80	1.87(-2)	2.40(-2)
5	-2. <u>2656196675727</u>	7.93(-12)	100	1.50(-2)	1.92(-2)

The last two columns give relative errors in the corresponding approximations for the Gauss-Legendre and Gauss-Chebyshev rules with  $2m = 20(20)100$  points. As we can see these rules are very slow.

## References

- [1] G. Beer, I. Smith, Ch. Duenser, The Boundary Element Method with Programming – For Engineers and Scientists, Springer, Wien – New York, 2008.
- [2] T.S. Chihara, An Introduction to Orthogonal Polynomials, Gordon and Breach, New York, 1978.
- [3] J.A. Crow, Quadrature of integrands with a logarithmic singularity, Mathematics of Computation 60 (1993), 297–301.
- [4] A.S. Cvetković, G.V. Milovanović, The MATHEMATICA package “OrthogonalPolynomials”, Facta Universitatis. Series: Mathematics and Informatics 19 (2004), 17–36.
- [5] C.F. Gauss, Methodus nova integralium valores per approximationem inveniendi, Commentationes Societatis Regiae Scientiarum Göttingensis Recentiores 3 (1814) [Werke III, pp. 163–196].
- [6] 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, F. Fehér, eds.), Birkhäuser, Basel, 1981, pp. 72–147.
- [7] W. Gautschi, On generating orthogonal polynomials, SIAM Journal on Scientific and Statistical Computing 3 (1982), 289–317.
- [8] W. Gautschi, Algorithm 726: ORTHPOL – a package of routines for generating orthogonal polynomials and Gauss-type quadrature rules, ACM Transactions on Mathematical Software 20 (1994), 21–62.
- [9] W. Gautschi, Orthogonal polynomials: applications and computation, Acta Numerica (1996), 45–119.
- [10] W. Gautschi, Orthogonal Polynomials: Computation and Approximation, Clarendon Press, Oxford 2004.
- [11] W. Gautschi, Numerical Analysis, Second Edition, Birkhäuser, Boston, 2012.
- [12] G. Golub, J.H. Welsch, Calculation of Gauss quadrature rules, Mathematics of Computation 23 (1969), 221–230.
- [13] A.S. Householder, The Theory of Matrices in Numerical Analysis, Dover, New York, 1975.
- [14] C.G.J. Jacobi, Über Gaußs neue Methode, die Werte der Integrale näherungsweise zu finden, Journal für die Reine und Angewandte Mathematik 30 (1826), 301–308.
- [15] S. Karlin, W. J. Studden, Tchebycheff Systems: With Applications in Analysis and Statistics, Pure and Applied Mathematics XV, Wiley Interscience, New York, 1966.
- [16] J. T. Katsikadelis, Boundary Elements: Theory and Applications, Elsevier, Amsterdam, 2002.
- [17] K.V. Laščenov, On a class of orthogonal polynomials, Učen. Zap. Leningrad. Gos. Ped. Inst. 89 (1953), 167–189 (Russian).
- [18] J. Ma, V. Rokhlin, S. Wandzura, Generalized Gaussian quadrature rules for systems of arbitrary functions, SIAM Journal on Numerical Analysis 33 (1996), 971–996.
- [19] G. Mastroianni, G.V. Milovanović, Interpolation Processes – Basic Theory and Applications, Springer Monographs in Mathematics, Springer – Verlag, Berlin – Heidelberg, 2008.
- [20] G.V. Milovanović, Quadrature processes and new applications, Bulletin, Classe des Sciences Mathématiques et Naturelles, Sciences mathématiques 38 (2013), 83–120.
- [21] G. V. Milovanović, Construction and applications of Gaussian quadratures with nonclassical and exotic weight function, Studia Universitatis Babeş-Bolyai Mathematica 60 (2015), 211–233.
- [22] G.V. Milovanović, A.S. Cvetković, Gaussian-type quadrature rules for Müntz systems, SIAM Journal on Scientific Computing 27 (2005), no. 3, 893–913.
- [23] G.V. Milovanović, A.S. Cvetković, Special classes of orthogonal polynomials and corresponding quadratures of Gaussian type, Mathematica Balkanica 26 (2012), 169–184.
- [24] G. V. Milovanović, T. S. Igić, D. Turnić, Generalized quadrature rules of Gaussian type for numerical evaluation of singular integrals, Journal of Computational and Applied Mathematics 278 (2015), 306–325.

- [25] R. Nahlik, R. Bałeckı, A new method of numerical evaluation of singular integrals occurring in two-dimensional BIEM, *Applied Mathematical Modelling* 7 (1983), no. 3, 169–172.
- [26] A. P. Prudnikov, Yu. A. Brychkov, O. I. Marichev, *Integrals and Series, Vol. 1: Elementary Functions*, Gordon and Breach, Newark, NJ, 1998.
- [27] D.M. Smith, Algorithm 693: A FORTRAN package for floating-point multiple-precision arithmetic, *ACM Transactions on Mathematical Software* 17 (1991), 273–283.
- [28] R.N.I. Smith, Direct Gauss quadrature formulae for logarithmic singularities on isoparametric elements, *Engineering Analysis with Boundary Elements* 24 (2000), 161–167.
- [29] T. J. Stieltjes, Sur une généralisation de la théorie des quadratures mécaniques, *Comptes Rendus de l'Académie des Sciences Paris* 99 (1884), 850–851.
- [30] G. Tsamasphyros, E. E.Theotokoglou, A quadrature formula for integrals with nearby singularities, *International Journal for Numerical Methods in Engineering* 67 (2006), 1082–1093.
- [31] C.W. Ueberhuber, *Numerical Computation 2: Methods, Software, and Analysis*, Springer-Verlag, Berlin, 1997.