FACTA UNIVERSITATIS (NIŠ)
Ser. Math. Inform. 17 (2002), 57–74

NUMERICAL INTEGRATION OF FUNCTIONS WITH LOGARITHMIC END POINT SINGULARITY*

Gradimir V. Milovanović and Aleksandar S. Cvetković

Abstract. In this paper we study some integration problems of functions involving logarithmic end point singularity. The basic idea of calculating integrals over algebras different from the standard algebra $\{1, x, x^2, \ldots\}$ is given and is applied to evaluation of integrals. Also, some convergence properties of quadrature rules over different algebras are investigated.

1. Introduction

The basic motive for our work is a slow convergence of the Gauss-Legendre quadrature rule, transformed to $(0, 1)$,

$$I(f) = \int_0^1 f(x)\,dx \simeq Q_n(f) = \sum_{\nu=1}^{n} A_\nu f(x_\nu), \tag{1.1}$$

in the case when $f(x) = x^x$. It is obvious that this function is continuous (even uniformly continuous) and positive over the interval of integration, so that we can expect a convergence of (1.1) in this case. In Table 1.1 we give relative errors in Gauss-Legendre approximations (1.1), $\text{rel.err}(f) = |(Q_n(f) - I(f))/I(f)|$, for $n = 30$, 100, 200, 300 and 400 nodes.

All calculations are performed in D- and Q-arithmetic, with machine precision $\approx 2.22 \times 10^{-16}$ and $\approx 1.93 \times 10^{-34}$, respectively. (Numbers in parentheses denote decimal exponents.)

Received September 12, 2001. The paper was presented at the International Conference FILOMAT 2001 (Niš, August 26–30, 2001).

2000 *Mathematics Subject Classification.* Primary 65D30, 65D32.

*The authors were supported in part by the Serbian Ministry of Science, Technology and Development (Project #2002: Applied Orthogonal Systems, Constructive Approximation and Numerical Methods).

Table 1.1: The relative errors in Gauss-Legendre approximations $Q_n(x^x)$ with n nodes, in D- and Q-arithmetic

n	30	100	200	300	400
D-arthm.	3.74(−7)	3.14(−9)	1.98(−10)	3.93(−11)	1.24(−11)
Q-arthm.	3.74(−7)	3.14(−9)	1.98(−10)	3.93(−11)	1.24(−11)

From this table we can see that the quadrature rule (1.1) converges, but the convergence is rather slow. The quadrature rule with 300 nodes gives only 11 decimal digits precision in the result like with 400 points, but one can also encounter that increasing in number of nodes after 100, the precision of result increases very slowly. If we further increase the number of points this fact becomes quite evident. We can also see that errors are the same in D- and Q-arithmetic, which means that the problem does not lie in ill-conditioning.

What is presented is a phenomenon called the *saturation*. We say that the integration of the function $x \mapsto x^x$ on $(0,1)$ is saturated with respect to the Gauss-Legendre quadrature rule. We also note that this function has a logarithmic singularity in its derivative at the point 0, and we can say that the problem of slow convergence occurs because of this singularity.

In this paper we consider integration of functions with logarithmic singularities and present some problems involved with them. The next section is devoted to present problem of saturation of Gaussian quadrature rules and also develops theory needed for an application of quadrature rules over different algebras. Section 3 is concerned with different ways to construct three term recurrence coefficients of orthogonal polynomials over different algebras and some examples are given as well. Section 4 is concerned with orthogonal polynomials with respect to the measure $x^x\chi_{(0,1)}dx$, three term recurrence coefficients are given for orthogonal polynomials over algebras generated by x and $-\log(x)$.

2. Saturation Problem and Müntz Systems

Let $\Lambda = \{\lambda_0, \lambda_1, \lambda_2, \ldots\}$ be a real sequence. We consider Müntz polynomials as linear combinations of the Müntz system $\{x^{\lambda_0}, x^{\lambda_1}, \ldots, x^{\lambda_n}\}$. By $M_n(\Lambda)$ we denote the set of all such polynomials, i.e.,

$$M_n(\Lambda) = \operatorname{span}\{x^{\lambda_0}, x^{\lambda_1}, \ldots, x^{\lambda_n}\},$$

where the linear span is over the real numbers. The union of all $M_n(\Lambda)$ is denoted by $M(\Lambda)$, i.e., $M(\Lambda) = \bigcup_{n=0}^{+\infty} M_n(\Lambda)$.

The first considerations of orthogonal Müntz systems were made by the Armenian mathematicians Badalyan [1] and Taslakyan [14]. Recently, it was rediscovered by McCarthy, Sayre and Shawyer [9]. A complete investigation of such systems, including some inequalities of Markov type, was done by Borwein, Erdélyi, and Zhang [3] (see also the book [2]).

Suppose that $\lambda_k > -1/2$ for every $k \in \mathbb{N}_0$ and let $\Lambda_n = \{\lambda_0, \lambda_1, \ldots, \lambda_n\}$. If Γ is a simple contour surrounding all the zeros of the denominator in the rational function

$$W_n(s) = \prod_{k=0}^{n-1} \frac{s + \lambda_k + 1}{s - \lambda_k} \cdot \frac{1}{s - \lambda_n} \qquad (n \in \mathbb{N}_0), \tag{2.1}$$

then the Müntz-Legendre polynomials are defined by (see [1, 2, 3, 9, 11, 14])

$$P_n(x) = P_n(x; \Lambda_n) = \frac{1}{2\pi i} \oint_\Gamma W_n(s) x^s \, ds. \tag{2.2}$$

In the case $n = 0$, an empty product in (2.1) should be taken to be equal to 1. For the Müntz-Legendre polynomials (2.2) the following orthogonality relation holds:

$$(P_n, P_m) = \int_0^1 P_n(x) \overline{P_m(x)} \, dx = \frac{\delta_{nm}}{2\lambda_n + 1} \, .$$

Returning now to saturation problems, we give the following definition:

Definition 2.1. We will say that some quadrature rule Q_n is saturated of order ϕ_n, where ϕ_n is positive zero converging sequence, if and only if

$$\lim_{n \to +\infty} \frac{\|I(f) - Q_n(f)\|}{\phi_n} = C \neq 0.$$

It is obvious for Gauss-Legendre quadrature rule applied to the function $f(x) = x^x$ that respected sequence ϕ_n converging to zero very slowly.

This is not the first saturation problem encountered. Saturation problems occur regularly in problems of polynomial approximation. A basic tool which is constructed for solving saturation problems are Müntz polynomials or some other kinds of generalized polynomials. A numerical algorithm for the construction of generalized Gaussian quadratures was investigated by Ma, Rokhlin and Wandzura [8], but, in a general case, this algorithm is ill conditioned.

Recently, one of us presented a stable numerical method for constructing the generalized Gaussian quadratures for Müntz polynomials [12]. Our constructive method is based on an application of orthogonal Müntz polynomials, as well as on a numerical procedure for evaluation of such polynomials with a high-precision [11]. However, such construction is not so easy as in the algebraic case (cf. [5]). Usually, it cannot be constructed with a standard software for algebraic Gaussian quadrature rules.

In some special cases, the standard software ([5]) for construction of Gaussian quadrature rule can be applied. In the special case when $\lambda_0 = \lambda_1 = \cdots = \lambda$, i.e., $\Lambda = \Lambda(\lambda) = \{\lambda, \lambda, \ldots\}$, (2.1) reduces to $W_n(s) = (s+\lambda+1)^n/(s-\lambda)^{n+1}$, so that (2.2) becomes

$$P_n(x; \Lambda_n) = x^\lambda L_n(-(2\lambda+1)\log x),$$

where $L_n(x)$ is the Laguerre polynomial orthogonal with respect to e^{-x} on $[0, +\infty)$ and such that $L_n(0) = 1$. Thus, in this special case, Müntz polynomials become Laguerre polynomials in $(-\log x)$. This gives a direction to take an integration with respect to the Gauss-Laguerre quadrature rule. We can either transform integral over $(0,1)$ to $(0,+\infty)$ or to take an integration over $(0,1)$, in which case we have to transform nodes of the Gauss-Laguerre quadrature rule (with the exponential weight e^{-x}),

$$\int_0^{+\infty} g(t)e^{-t}\,dt \simeq \sum_{\nu=1}^{n} A_k g(\tau_\nu),$$

from the interval $(0,+\infty)$ to $(0,1)$. Since $\int_0^{+\infty} g(t)e^{-t}\,dt = \int_0^1 g(-\log x)\,dx$, we get the quadrature formula

$$\int_0^1 f(x)\,dx \simeq Q_n^L(f) = \sum_{\nu=1}^{n} A_k f(x_\nu), \quad x_k = e^{-\tau_k}, \quad k = 1, \ldots, n, \tag{2.3}$$

which is exact for all $f \in M_{2n-1}(\Lambda(0)) = \operatorname{span}\{1, \log x, \log^2 x, \ldots, \log^{2n-1} x\}$. In this case, our quadrature rule (2.3) is constructed with standard software tools. An application of this quadrature to $f(x) = x^x$ gives results with relative errors displayed in Table 2.1.

There is also a quadrature rule which can further decrease number of points in which function is evaluated, but this quadrature rule requires algebra different then algebra constructed over $\{(-\log x)^k,\ k \in \mathbb{N}_0\}$.

Table 2.1: The relative errors in quadrature approximations $Q_n^L(x^x)$ with n nodes in D-arithmetic

n	40	50	60	70	80
D-arthm.	7.07(−11)	1.94(−12)	7.04(−14)	3.16(−15)	1.69(−16)

Our integrand can also be written in the form $x^x = e^{x \log x}$, which is entire function in $x \log x$. If we calculate moments for algebra generated by $x \log x$, with respect to the Legendre measure, we can use the Chebyshev algorithm to construct the coefficients of three term recurrence relation, which can be used for construction of orthogonal polynomials in $x \log x$ over $(0, 1)$, with respect to Legendre measure. Using QR algorithm, the Gaussian quadrature rule can be constructed. This formula is exact for all $f \in \operatorname{span}\{1, x \log x, x^2 \log^2 x, \ldots, x^{2n-1} \log^{2n-1} x\}$. Applying this quadrature rule to the function $f(x) = x^x$ we have the rate of convergence given in Table 2.2. The exact result in D-arithmetic is achieved with only 5 points in the corresponding Gaussian quadrature rule.

Table 2.2: The relative errors in quadrature approximations for $I(x^x)$ by the quadrature formula obtained using algebra over $x \log(x)$

n	3	4	5
D-arthm.	1.28(−9)	1.96(−13)	1.11(−16)

Let $x \mapsto \psi(x)$ be a continuous function defined over a closed interval I, and such that all its powers $(\psi(x))^k$, $k = 0, 1, \ldots,$ be mutually linearly independent. Without loss of generality, we take $I = [0, 1]$. We can construct an algebra of functions with mother function $\psi(x)$ in the following manner taking all powers of the function $\psi(x)$ from 0 to n ($\in \mathbb{N}_0$). Then, we take all linear combinations over the set of powers with real numbers and obtain a real vector space of dimension n, denoted by $U_n \equiv U_n(\psi)$. If we take the union of all such linear spaces over $n \in \mathbb{N}_0$ we get the linear space $U = \bigcup_{n=1}^{+\infty} U_n$ in which the multiplication is a closed operation.

The set U can be understood as an algebra of polynomials in $\psi(x)$. Since all linear combinations over set of powers are in U, the elements in U are polynomials in $\psi(x)$. The algebra constructed in this manner is isomorphic with standard polynomial algebra over power set $\{1, x, x^2, x^3, \ldots\}$ and the linear mapping $L(\psi(x)^k) = x^k$, $k \in \mathbb{N}_0$, $L(\alpha_1 f_1 + \alpha_2 f_2) = \alpha_1 L(f_1) + \alpha_2 L(f_2)$, constitutes this isomorphism.

This means that we can use the Euclid's algorithm for polynomial division or to find a common divisor, etc. We can say that this isomorphism allows us to have no difference between standard algebra of polynomials and algebra U when we are interested only in properties with respect to arithmetic operations of addition and multiplication.

Our interest is now in some approximation properties of our algebra U. For this purposes we need the following theorem of Stone (see [4]):

Theorem 2.1. *A function $g \in C(I)$ is approximable by functions of real algebra U if and only if*

(a) $h(x) = 0$ *for all* $h \in U$ *implies* $g(x) = 0$, *and*

(b) *The algebra U separates any two points separated by g.*

In our algebra U, there are no point x at which all $h \in U$ are zero, because we have in U the function $\psi(x)^0 = 1$, which has no zeroes on any closed interval I.

We say that function g separates two different points x and y, if and only if $g(x)$ is not equal with $g(y)$. We say that algebra separates two different points x and y, if and only if there is $h \in U$ which separate points x and y. To be approximable means that the function g can be approximated arbitrarily close, with respect to uniform norm, with polynomials from the algebra U.

This theorem states that if any two points of an interval are separated by the algebra U and there is no point which is zero for all functions in U, then the algebra is dense in $C(I)$. A separation property of the algebra U, constructed in the previous manner, for two points x and y, is exactly the same as the question weather mother function ψ separates those points, since the constant function $\psi(x)^0$ does not separate any two points. In another words, if our mother function separates any two points, then they are separated by the algebra U, and if the algebra U separates any two points then also the mother function separates them.

If our mother function does not separate all points in an interval I then we still can approximate functions but only those functions which are in the uniform closure of the algebra U, and which also do not separate points not separated by the mother function ψ.

Let the uniform closure of U be denoted by $C(U)$. Then by theorem of Stone we can state that every function in $C(U)$ is approximable by polynomials in ψ. It is easy to check that $C(U)$ is a Banach function space with the uniform norm.

Lemma 2.1. *The function space $C(U)$ is a Banach function space with respect to the uniform norm.*

It is a linear space, and it is subset of the space $C(I)$. This means that any sequence of functions from $C(U)$ converge to some function from $C(I)$ in the uniform norm. Since the convergence is in uniform norm, for any given $x \in I$, any sequence of points $\psi_k(x)$ converge to $\psi(x)$ pointwise. If the algebra U does not separate points x and y, then from the poinwise convergence it is obvious that the function ψ cannot separate points x and y. The previous means that our space is closed.

We also need an interpolation property of our set of powers. Let the closed interval $[\min \psi(x), \max \psi(x)]$ be denoted by F, where min and max is taken over interval I. Then we can think of our interpolation problem for polynomials in $\psi(x)$ as an interpolation problem for polynomials of standard algebra over the interval F. Since the system of functions $\{1, x, x^2, x^3, \ldots, x^n\}$ is a Haar system on any closed interval F, the interpolation problem has the unique solution.

This situation can be further elaborated in the following manner. Two sets of interpolation points $\{x_k\}$, $\{y_k\}$, $k \in K$, are called equivalent if sets $\{f(x_k)|\ k \in K\}$ and $\{f(y_k)|\ k \in K\}$ are the same. It is evident that if the interpolation problem has a solution over either set of points that it has solution over other set of points.

Also, it is obvious that if some set of interpolation points has two points x and y which are not separated by ψ, then the interpolation problem has no a solution or this solution is not given uniquely. If we take the set of interpolation points x_k of n points which has no points which are not separated by function ψ, then set of points $\psi(x_k)$ has also n different points which all belong to the interval F and because the system of functions $\{1, x, x^2, x^3, \ldots, x^n\}$ is a Haar system over any closed interval, we have following lemma:

Lemma 2.2. *For every set of interpolation points, which has no points not separated by function ψ, the interpolation problem has a unique solution.*

This lemma gives an opportunity to construct the Lagrange interpolation polynomial over algebra U. Further, we can construct an interpolatory quadrature rule for our algebra U, which can have U-algebraic degree of exactness. Since there is an isomorphism between the polynomial algebra and our algebra U, we can also construct a sequence of orthogonal polynomials with respect to some linear functional over $C(U)$, if such polynomials exist. They satisfy a three term recurrence relation, because of the isomorphism. Thus, if we have the coefficients in three term recurrence relation, we are able to construct Gaussian quadrature rules, because operations have the same algebraic property as they have for the traditional set of powers $\{1, x, x^2, x^3, \ldots\}$. We can use the standard software for this construction, too. This means that the construction of Gaussian quadrature rules over algebra U can be done using QR algorithm (see [7]).

One can also wonder weather our quadrature rule constructed for some linear functional over $C(U)$ converge to the value of linear functional. The answer to this question can be found in a modification of standard theorems which solve the question of convergence in the standard quadrature rules. So, we have the following lemma:

Lemma 2.3. *Let $C(U)$ denote the uniform closure of the algebra U over a closed interval I. Further, let the quadrature rules Q_n converge to the value of the linear functional L for every element of the algebra U and let all functionals Q_n are uniformly bounded over $C(U)$. Then and only then the sequence $Q_n(f)$ converges to $L(f)$ for every $f \in C(U)$.*

This can be proved using Banach-Steinhaus theorem over Banach space $C(U)$.

The main advantage of using algebras instead of Müntz polynomials for saturation problems is the existence of software which can construct Gaussian quadrature rules, easily. We can use, for example, the Chebyshev algorithm to construct three term recurrence coefficients from the moments and then use QR algorithm to construct the Gaussian quadrature rule.

In this paper, we consider algebras generated with mother functions $\psi(x) = x^p(-\log x)^q$, where $p, q \geq 0$ and $p + q > 0$. We see that our mother function, for $p, q > 0$, cannot separate all points from the interval $[0, 1]$. It can be easily shown that in these cases there are pairs of points x and y which are not separated. The only point which is separated is a point for which the mother function achieves the maximum value, since the minimum value, which is 0, is achieved in two points 0 and 1 at the both ends of interval $[0, 1]$.

In the sequel, we give three term recurrence coefficients for orthogonal polynomials in $\psi(x) = x^p(-\log x)^q$ over $(0, 1)$, with respect to the weight function $x \mapsto x^\alpha(-\log x)^\beta$, $\alpha, \beta > -1$, and also we solve some integrals using Gaussian quadrature rules over this algebras.

For this purpose the following integral is needed

$$\int_0^1 x^r(-\log x)^s dx = \frac{\Gamma(s+1)}{(r+1)^{s+1}}, \quad r, s > -1, \tag{2.4}$$

which can be evaluated by using an exponential transformation for the variable x to the interval $(0, +\infty)$.

All moments with respect to the weight $x \mapsto x^\alpha(-\log x)^\beta$ $(\alpha, \beta > -1)$ can be expressed in the following manner

$$\begin{aligned} \mu_k &= \int_0^1 x^\alpha(-\log x)^\beta [x^p(-\log x)^q]^k\, dx \\ &= \int_0^1 x^{\alpha+pk}(-\log x)^{\beta+qk} dx = \frac{\Gamma(\beta+qk+1)}{(\alpha+pk+1)^{\beta+qk+1}}. \end{aligned} \tag{2.5}$$

The special case for $p = 0$ and $q = 1$ corresponds to the generalized Laguerre polynomials. Namely, we have

$$\int_0^1 x^\alpha(-\log x)^\beta L_n^\beta(-(1+\alpha)\log x) L_k^\beta(-(1+\alpha)\log x) dx = 0, \quad k \neq n, \tag{2.6}$$

where L_n^β denotes the monic generalized Laguerre polynomial of degree n and with parameter β. In this special case, three term recurrence coefficients are known. The convergence of the quadrature rules is assured from the convergence of respective Gauss-Laguerre quadrature rules, which are introduced taking substitution $x = e^{-t}$.

In [5], the weight $x \mapsto x^\sigma(-\log x)$ is considered, also numerical values for three term recurrence coefficients are given, since analytical expressions are not known. Our weight function is more general and analytical expressions for three term recurrence coefficients are known, if polynomials are constructed over the algebra $(-\log x)^k$, $k \in \mathbb{N}_0$.

We illustrate now the efficiency of a such quadrature rule.

Example 2.1. In this example we take the entire function $f(x) = e^x$. For the weight with parameters $\alpha = 2$ and $\beta = 10$, the relative errors in quadrature approximations for $n = 10, 20, 30$ are given in Table 2.3. We get results with the machine precision in D-arithmetic with 30 points.

Example 2.2. This example involves function with a logarithmic singularity. We evaluate integral of function $f(x) = x^x$ with respect to weight $x^3(-\log x)^{10}$. The corresponding results are given in Table 2.4.

Table 2.3: The relative errors in quadrature approximations for $I(x^2(-\log x)^{10}e^x)$

n	10	20	30
D-arthm.	2.11(−8)	2.97(−13)	1.11(−16)

Table 2.4: The relative errors in quadrature approximations for $I(x^2(-\log x)^{10}x^x)$

n	10	20	30
D-arthm.	8.75(−9)	1.47(−13)	1.11(−16)

Later it is shown that there is more efficient Gaussian quadrature rule, for solving this integral, namely the machine precision can be achieved with only five points.

Example 2.3. We consider a function with singularity in the complex plain near the interval of integration, i.e., $f(x) = 100/(100x^2 + 1)$, with respect to the weight function $x^3(-\log x)^2$. The corresponding relative errors are presented in Table 2.5.

Table 2.5: The relative errors in quadrature approximations for $I(x^3(-\log x)^2 f(x))$, $f(x) = 100/(100x^2 + 1)$

n	30	40	50	60	70	80	90
D-arthm.	2(−9)	2(−11)	1(−12)	2(−13)	5(−15)	2(−15)	1(−16)

As we can see the saturation is evident. This means that our algebra is not suitable for integration of this function.

Now, a comparison is given between Gauss-Legendre quadrature rule and quadrature rule in $(-\log x)$, with $\alpha = 0$ and $\beta = 0$, for the previous function. For Gauss-Legendre quadrature rules the results are given in Table 2.6. The corresponding results for Gauss-Laguerre quadrature rules are given in Table 2.7.

As we can see, our quadrature rule is saturated but not as heavily as the Legendre quadrature rule (see Table 1.1). This also suggests that there are no quadrature rule which has well behavior for all types of integrals in saturation sense.

Table 2.6: Gauss-Legendre quadrature rule

n	30	40	50
D-arthm.	3.87(−12)	3.79(−16)	2.22(−16)

Table 2.7: Gauss-Laguerre quadrature rule

n	40	70	100	200	300
D-arthm.	5.28(−6)	3.20(−8)	1.07(−8)	2.45(−12)	3.88(−15)

3. Three term recurrence coefficients

Except for $\beta = 0,\ p = 1,\ q = 0$, when we have another special case which is connected with Jacobi polynomials, for general moments given by (2.5), we are not able to calculate three term recurrence coefficients in an analytic form.

Table 3.1: Three term recurrence coefficients in the case $\alpha = \beta = 0,\ p = q = 1$

k	α_k	β_k
0	$\frac{1}{4}$	1
1	$\frac{7}{40}$	$\frac{5}{432}$
2	$\frac{1734889}{9561160}$	$\frac{239029}{27000000}$
3	$\frac{224751722550116062327633 63}{1229629452608471349397513 60}$	$\frac{14468256301374835983}{168046486959520225 0000}$

It is, however, easy to calculate three term recurrence coefficients using some programming language such as Mathematica which can compute with rational numbers and symbols. We give some exact values for three term recurrence coefficients. In the case $\alpha = 0,\ \beta = 0,\ p = 1,\ q = 1$, the three term recurrence coefficients are given in Table 3.1.

It can be seen that three term recurrence coefficients are rational numbers, which numerators and denominators increase very fast with k. Similar behavior can be seen for any other combination of $\alpha,\ \beta,\ p$ and q, when they are integers, but when some or all of them are rational, then according to (2.5), the moments cannot even be calculated as rational numbers.

Table 3.2: Three term recurrence coefficients for $\alpha = \beta = 0,\ p = q = 1$

k	α_k	β_k
0	.2500000000000000	1.0000000000000000
1	.1750000000000000	.01157407407407407
2	.1814517276146409	.008852925925925926
3	.1827800416404630	.008609674955513954
4	.1832701352351433	.008538162504297512
5	.1835041444127862	.008507622973183079
6	.1836338776671062	.008491794720322833
7	.1837132309050915	.008482542087505247
8	.1837652815357174	.008476668007965964
9	.1838012574120895	.008472707068501194
10	.1838271555258116	.008469910324691129
11	.1838464161023932	.008467862508424024
12	.1838611274586809	.008466318278465908
13	.1838726168568256	.008465125115133252
14	.1838817606155675	.008464184154779261
15	.1838891562089048	.008463429033512966
16	.1838952223385164	.008462813860948587
17	.1839002594805140	.008462306083234165
18	.1839044877920375	.008461882088706323
19	.1839080715605580	.008461524414527530
20	.1839111353914029	.008461219920776388
21	.1839137751766370	.008460958567166738
22	.1839160656830992	.008460732575356799
23	.1839180659000882	.008460535843948855
24	.1839198228719387	.008460363532677109
25	.1839213744869127	.008460211762089140
26	.1839227515350129	.008460077393463645
27	.1839239792458009	.008459957865369290
28	.1839250784511384	.008459851070797856
29	.1839260664738652	.008459755263755947
30	.1839269578138276	.008459668987511629
31	.1839277646823939	.008459591018943195
32	.1839284974225275	.008459520324989171
33	.1839291648415835	.008459456028283241
34	.1839297744769544	.008459397379825226
35	.1839303328096127	.008459343737088712
36	.1839308454369102	.008459294546363598
37	.1839313172132825	.008459249328422600
38	.1839317523654927	.008459207666815415
39	.1839321545875477	.008459169198254124

If we use the fact that max machine number for D-arithmetic is of order 10^{308}, it is evident that even if analytic expressions are known for three term recurrence coefficients, they cannot be used for exact calculations. We can still use Chebyshev algorithm, which is, however, ill-conditioned.

We have used Chebyshev algorithm in arithmetic with 90 decimal digits arithmetic to obtain the first 40 three term recurrence coefficients with 34 decimal digits precision, for the case $\alpha = 0$, $\beta = 0$, $p = 1$, $q = 1$, the results are shown in Table 3.2.

Example 3.4. We use the three term recurrence coefficients from Table 3.2 to evaluate some integrals. The results are presented in Table 3.3.

Table 3.3: Some integrals calculated using algebra generated by $\psi(x) = x(-\log x)$

$f(x)$	$n = 5$	$n = 10$	$n = 15$	$n = 25$
$\dfrac{x^x}{100x^{2x}+1}$	8.49(−15)	1.11(−16)	1.11(−16)	1.11(−16)
$\dfrac{x(-\log x)}{100x^{2x}+1}$	9.47(−13)	1.11(−16)	1.11(−16)	1.11(−16)
$\dfrac{\sin(x(-\log x))}{100x^{-2x}+1}$	2.79(−7)	1.85(−15)	2.22(−16)	2.22(−16)
$\dfrac{\sin(100x(-\log x))}{100x^{-2x}+1}$	1.52(0)	5.37(−2)	7.43(−5)	3.55(−15)

The convergence is really fast and we need only 25 first three term recurrence coefficients to evaluate almost all of these integrals. This means that in a wide variety of applications, the Chebyshev algorithm is enough if it is used in Q-arithmetic, and also for some integrals it would be enough to use it in D-arithmetic if convergence is extremely fast, like in the first case in Table 3.3.

Example 3.5. For delicate integrals from Table 3.4, we use only three term recurrence coefficients obtained using Chebyshev algorithm in Q-arithmetic. Here, J_k denotes the Bessel function of the first kind and order k. For the weight function $w(x) = (-\log x)^3$ and $\psi(x) = x(-\log x)$, the relative errors are given in Table 3.4.

Example 3.6. We consider now some examples when p and q in the function $\psi(x) = x^p(-\log x)^q$ are not integers. The corresponding relative errors in such examples are given in Table 3.5, for $w(x) = (-\log x)^{10}$ and $p = 1/10$,

Table 3.4: The case $w(x) = (-\log x)^3$ and $\psi(x) = x(-\log x)$

$f(x)$	$n = 5$	$n = 10$	$n = 15$	$n = 20$
$\dfrac{\sin(70x^x)}{100x^x + 1}$	6.24(0)	7.05(−5)	4.45(−10)	3.55(−15)
$\dfrac{\sin(70\sin(x(-\log x)))}{100x^x + 1}$	4.34(0)	6.87(−4)	2.34(−8)	3.55(−16)
$J_0\Big(\dfrac{x(-\log x)}{200 - x^x}\Big)$	1.16(−16)	1.16(−16)	1.16(−16)	1.16(−16)
$J_{20}\Big(\dfrac{x(-\log x)}{200 - x^x}\Big)$	0.25(0)	3.53(−11)	1.11(−16)	1.11(−16)

Table 3.5: The case $w(x) = (-\log x)^{10}$, $p = 1/10$, $q = 1/20$

$f(x)$	$n = 5$	$n = 10$	$n = 15$	$n = 18$
$\dfrac{\sin((\sqrt{x}(-\log x))^{1/10})}{x^{1/10}(-\log x)^{1/5}}$	1.22(−1)	2.34(−6)	3.50(−12)	5.68(−14)
$\dfrac{\sin(30x^{1/20}(-\log x)^{1/10})}{100x^{1/10}(-\log x)^{1/5} + 1}$	5.97(0)	4.13(−4)	2.79(−10)	2.14(−13)
$\dfrac{x^{-x^{(-3/10)(-\log x)^{-7/10}}}}{30(\cos(x^{1/20(-\log x)^{1/10}}))^2 + 1}$	5.92(−7)	1.57(−14)	2.22(−16)	2.22(−16)
$\dfrac{J_0(x^{1/20(-\log x)^{1/10}})}{30(\cos(x^{1/20}(-\log x)^{1/10}))^2 + 1}$	2.84(−2)	2.55(−7)	1.17(−14)	1.50(−16)

$q = 1/20$. We emphasize that the three term recurrence coefficients, obtained in Q-arithmetic, are exact with 17 decimal digits till 18-th coefficient. For more then 18-point quadrature rule, the Chebyshev algorithm must be run in arithmetic with more decimal digits precision than it is Q-arithmetic.

Example 3.7. We give, also, some examples with $p = q = 1$ and $\alpha = 3$, $\beta = 10$. In this case we have relative errors as in Table 3.6. For the first function in Table 3.6, we get the machine precision only with 5-point rule, which is less then 30 points we needed when $\alpha = 3$, $\beta = 10$, $p = 0$ and $q = 1$.

Table 3.6: The case $w(x) = x^3(-\log x)^{10}$, $p = q = 1$

$f(x)$	$n = 5$	$n = 10$	$n = 15$	$n = 20$
x^x	1.11(−16)	1.11(−16)	1.11(−16)	1.11(−16)
$\dfrac{\sin(70x(-\log x))}{100x^{10x}+1}$	2.83(1)	1.26(−2)	6.31(−8)	2.84(−14)
$J_{10}(70x(-\log x))$	8.17(−1)	7.80(−5)	3.31(−10)	7.35(−15)

4. Weight Function $\mathbf{w(x) = x^x}$ on $[\mathbf{0}, \mathbf{1}]$

We tried to evaluate three term recurrence coefficients for the weight function $w(x) = x^x$ using Stieltjes and Lanczos algorithm, with Laguerre measure as a discretization measure (see [5], [10], [6]).

For the sake of completeness, we recall that in Stieltjes algorithm, the integrals representing three term recurrence coefficients

$$\alpha_k = \frac{\int wxp_k^2 dx}{\int wp_k^2 dx}, \quad \beta_k = \frac{\int wp_k^2 dx}{\int wp_{k-1}^2 dx}, \tag{4.1}$$

are calculated using some approximation by a quadrature rule. The main point is to find some weight $\hat{w}$, which is not *very different* from w, for which a quadrature rule can be constructed easily. Then we approximate the integrals in (4.1), by quadrature rules with respect to $\hat{w}$, i.e.,

$$\int wpdx = \int \hat{w}\frac{w}{\hat{w}}pdx \approx \sum_{k=1}^{n} A_k^{\hat{w}} \left(\frac{w}{\hat{w}}p\right)(x_k^{\hat{w}}). \tag{4.2}$$

The measure of these quadrature rules is called, usually, the discretized measure. The Lanczos algorithm performs something similar, but in some cases it shows a better numerical stability.

Since we use the Laguerre measure as a discretized measure $\hat{w}$, it is obvious that this discretization procedure uses an algebra over $\psi(x) = (-\log x)$, in order to evaluate integrals in powers of x. We cannot evaluate integrals efficiently in Stieltjes procedure, because of saturation problem. It happens that, if we have nodes and weights of the Gauss-Laguerre quadrature, for example, with 16 decimal digits precision, we can only construct the first 14 three term recurrence coefficients in D-arithmetic with the 600-point Gauss-Laguerre quadrature rule.

Table 4.1: Three term recurrence coefficients α_k, β_k, $k = 0, 1, \ldots, 19$, for the weight $w(x) = x^x$ on $[0, 1]$ over algebra generated by $\psi(x) = x$ (left) and $\psi(x) = -\log x$ (right)

k	α_k	β_k	α_k	β_k
0	.5144482362005029	7.834305107121344(−1)	1.000000000000000	7.834305107121344(−1)
1	.4910266828809244	9.034273259880220(−2)	3.123988632158376	1.131472030307515
2	.5010811468982506	6.532422170358885(−2)	5.065492874394690	4.109275203714620
3	.5002274192432331	6.398294411189513(−2)	7.046670661021021	9.137614227774804
4	.5001097742225770	6.337428105285331(−2)	9.038416364518349	1.615930292656526(1)
5	.5000619392155803	6.307148321180321(−2)	1.103376873870124(1)	2.517395555615978(1)
6	.5000384116391916	6.290245968483624(−2)	1.303055223587346(1)	3.618646786463015(1)
7	.5000254623819853	6.279868054508732(−2)	1.502809201194203(1)	4.919844292536450(1)
8	.5000177418984606	6.273043434239860(−2)	1.702612350326998(1)	6.421009204062693(1)
9	.5000128531031649	6.268317240486356(−2)	1.902450682219748(1)	8.122135322370709(1)
10	.5000096073910574	6.264909512729829(−2)	2.102315261616556(1)	1.002321802204323(2)
11	.5000073684406322	6.262371688303956(−2)	2.302199905005075(1)	1.212425730399470(2)
12	.5000057742875211	6.260430952745989(−2)	2.502100194331687(1)	1.442525585971080(2)
13	.5000046085827927	6.258913641984857(−2)	2.702012914235859(1)	1.692621729794647(2)
14	.5000037365620693	6.257704941012625(−2)	2.901935684755667(1)	1.962714521331669(2)
15	.5000030712879705	6.256726476410453(−2)	3.101866712622798(1)	2.252804283309976(2)
16	.5000025549616926	6.255923266918557(−2)	3.301804621856896(1)	2.562891294688314(2)
17	.5000021481367535	6.255255817912250(−2)	3.501748337991205(1)	2.892975794430127(2)
18	.5000018232652519	6.254695166351771(−2)	3.701697008299243(1)	3.243057988004892(2)
19	.5000015607129029	6.254219681578086(−2)	3.901649946093514(1)	3.613138053574687(2)

Instead of the previous procedure, we use Chebyshev algorithm in Q-arithmetic and we obtained the first 20 three term recurrence coefficients in D-arithmetic with moments evaluated using the previous quadrature rules (see Table 4.1(left)).

If we seek for three term recurrence coefficients with respect to the weight $w(x) = x^x$ on $(0, 1)$ over algebra generated by $\psi(x) = (-\log x)$, the Stieltjes algorithm is quite stable. Applying Stieltjes algorithm we get the first 20 coefficients, which are shown in Table 4.1.

The recursion coefficients were created with the 200-point Gauss-Laguerre quadrature rule, which is used as discretization rule in Stieltjes procedure given in (4.2).

REFERENCES

1. G. V. Badalyan: *Generalisation of Legendre polynomials and some of their applications.* Akad. Nauk. Armyan. SSR Izv. Ser. Fiz.-Mat. Estest. Tekhn. Nauk **8** (5) (1955) 1–28 and **9** (1) (1956) 3–22 (Russian, Armanian summary).
2. P. Borwein and T. Erdélyi: *Polynomials and Polynomial Inequalities.* Springer Verlag, New York – Berlin – Heidelberg, 1995.
3. P. Borwein, T. Erdélyi and J. Zhang: *Müntz systems and orthogonal Müntz-Legendre polynomials.* Trans. Amer. Math. Soc. **342** (1994), 523–542.
4. R. A. DeVore and G. G. Lorentz: *Constructive Approximation.* Grundlehren der mathematischen Wissenschaften, Vol. 303, Springer-Verlag, Berlin – Heidelberg – New York, 1993.
5. 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.
6. W.B. Gragg and W.J. Harrod: *The numerically stable reconstruction of Jacobi matrices from spectral data.* Numer. Math. **44** (1984), 317–335.
7. G.H. Golub and J.H. Welsch: *Calculation of Gauss quadrature rule.* Math. Comput. **23** (1986), 221–230.
8. J. Ma, V. Rokhlin and S. Wandzura: *Generalized Gaussian quadrature rules for systems of arbitrary functions.* SIAM J. Numer. Anal. **33** (1996), 971–996.
9. P. C. McCarthy, J. E. Sayre and B. L. R. Shawyer: *Generalized Legendre polynomials.* J. Math. Anal. Appl. **177** (1993), 530–537.
10. G. V. Milovanović: *Numerical Analysis.* Naučna knjiga, Beograd, 1991 (Serbian).

11. G. V. Milovanović: *Müntz orthogonal polynomials and their numerical evaluation.* In: Applications and Computation of Orthogonal Polynomials (W. Gautschi, G.H. Golub, G. Opfer, eds.), ISNM, Vol. **131**, Birkhäuser, Basel, 1999, pp. 179–202.

12. G. V. Milovanović: *Construction of Gaussian type quadrature formulas for Müntz systems.* Invited Lecture at the International Meeting on Approximation Theory and Numerical Analysis (dedicated to 60th birthday of Professor G. Mastroianni), Vico Equense, Italy (August 31 – September 1, 1999).

13. G. Szegő: *Orthogonal Polynomials.* Amer. Math. Soc. Colloq. Publ., **23**, 4th ed., Amer. Math. Soc., Providence, R. I., 1975.

14. A. K. Taslakyan: *Some properties of Legendre quasi-polynomials with respect to a Müntz system. Mathematics, Èrevan University, Èrevan* **2** (1984) 179–189 (Russian, Armenian summary).

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