Third Serbian (28th Yu) Congress on Theoretical and Applied Mechanics
Vlasina Lake, Serbia, 5-8 July 2011 A-01:1-15

SOME QUADRATURE RULES FOR FINITE ELEMENT METHOD AND BOUNDARY ELEMENT METHOD

G.V. Milovanović[1], T. Igić[2], N. Tončev[2]

[1] Faculty of Computer Sciences, Megatrend University, Bulevar Umetnosti 29, 11070 Novi Beograd, Serbia
e-mail: gvm@megatrend.edu.rs
[2] Faculty of Civil Engineering and Architecture, University of Niš, A. Medvedeva 14, 18000 Niš, Serbia
e-mail: tomislav.igic@gaf.ni.ac.rs

Abstract. The boundary element method (BEM) and the finite element method (FEM) are very popular in many computational applications in engineering. These methods very often require the numerical evaluation of one dimensional or multiple integrals with singular or near singular integrands. Such problems appear in many subjects of mechanics (fracture mechanics, damage mechanics, etc.), as well as in other technical fields. In this paper we give some improvements of quadrature rules for FEM and BEM. Beside of general notions on Gaussian quadratures, we give a construction of weighted Gaussian quadratures for integrals with logarithmic and/or algebraic singularities. Also, we consider generalized quadratures of high degree of precision for Müntz systems. Numerical examples are included.

1. Introduction

The boundary element method (BEM) and the finite element method (FEM) are very popular in many computational applications in engineering, for example, in fracture mechanics, damage mechanics, electromagnetic diffraction, etc. Very often in such applications we need accurate numerical evaluation of one dimensional or multiple integrals with singular kernels and/or singular basis functions. Two kinds of singularities are typical: algebraic and logarithmic. Müntz and Müntz-logarithmic polynomials are typical functions with such properties. Also, an accurate evaluation of nearly singular one and multidimensional integrals is very important. For some additional details see, for example, [1], [9], [7], [8], [10], [11], [17], [18].

In this paper we propose a method for constructing the weighted Gaussian quadrature rules for integrals with algebraic and/or logarithmic singularities. This method gives Gaussian quadratures with a maximal algebraic degree of precision. Also, we mention another approach which enables us to obtain Gaussian quadratures for Müntz systems (for details see Milovanović [13] and Milovanović and Cvetković [15]).

Beside of general notions on quadratures of high algebraic degree of precision, we consider a stable and efficient construction of the weighted Gaussian quadratures for integrals of functions with end-point singularities. Such a construction is based on an application of the `Mathematica` package `OrthogonalPolynomials`, recently developed by Cvetković and Milovanović [2]. Numerical examples are included.

2. Quadrature Formulas for Integrals With Logarithmic Weight Functions

In numerical implementation of the BEM (see [9, Chapters 4 & 5]), quadrature formulas play a very important role, especially for higher order elements. For calculating integrals of the corresponding influence coefficients (for off-diagonal elements and diagonal elements), quadratures of Gaussian type are very appropriate. For sufficiently smooth functions on a finite interval $[a,b]$ a linear transformation to the standard interval $[-1,1]$ can be used and then an application of Gauss-Legendre quadrature formula provides numerical integration with a satisfactory accuracy. However, for integrals with a logaritmic singularity and/or some kind of algebraic singularities the convergence of the corresponding quadrature process is very slow, so that certain weighted quadratures are recommended. In such cases, the weight functions of the corresponding weighted Gaussian quadratures include these “difficult parts (with singularities)” of the integrand. In this section, we consider a few cases of such quadratures on the standard interval $[0,1]$. However, we first give some general notions on Gaussian quadratures.

2.1. General notions of Gaussian quadratures

Let $\mathscr{P}_m$ be a set of all algebraic polynomials of degree at most m. We consider the n-point weighted quadrature formula

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

where the weight function $w(x)$ is such one that all its moments $\mu_k = \int_a^b x^k w(x)\,dx < +\infty$, $k = 0,1,\ldots$, and $\mu_0 > 0$. Quadrature rule (1) is known as *interpolatory* if it is exact for all polynomials of degree at least $n-1$, i.e., if the remainder term $R_n(f) = 0$ for each $f \in \mathscr{P}_{n-1}$.

However, if the nodes x_k and the weights A_k in (1) are selected so that $R_n(f) = 0$ for each $f \in \mathscr{P}_{2n-1}$, the rule (1) is the *Gaussian* quadrature formula. In that case, the nodes x_k are zeros of the monic orthogonal polynomial $\pi_n(w;x)$ and the corresponding weights A_k (Christoffel numbers) can be expressed by the so-called Christoffel function $\lambda_n(w;x)$ (cf. [12, Chapters 2 & 5]) in the form $A_k = \lambda_n(w;x_k) > 0$, $k = 1,\ldots,n$. Positivity of Christoffel numbers is very important for the convergence of the quadrature formulas. In the special case $w(x) = 1$ on $[-1,1]$, the nodes x_k are zeros of the Legendre polynomial $P_n(x)$. It was originally discovered by Gauss in 1814, of course, without theory of orthogonality.

As we know [12, Chapters 2], the (monic) polynomials $\pi_n(w;x)$ orthogonal with respect to the weight function $w(x)$ on $[a,b]$ satisfy the three-term recurrence equation

$$\begin{aligned} &\pi_{k+1}(t) = (t-\alpha_k)\pi_k(t) - \beta_k\pi_{k-1}(t), \qquad k = 0,1,2,\ldots, \\ &\pi_0(t) = 0, \quad \pi_{-1}(t) = 0, \end{aligned} \tag{2}$$

where $(\alpha_k) = (\alpha_k(w))$ and $(\beta_k) = (\beta_k(w))$ are sequences of recursion coefficients. The coefficient β_0 which is multiplied by $\pi_{-1}(x) = 0$ in the recurrence relation (2) may be arbitrary, but it is convenient to define it by $\beta_0 = \mu_0 = \int_a^b w(x)\,dx$.

For generating Gaussian quadrature rules there are numerical methods, which are computationally much better than a computation of nodes by using Newton’s method and then a direct application of the classical Christoffel’s expressions for the weights (see e.g. Davis and Rabinowitz [3]). The characterization of the Gaussian quadratures via an eigenvalue problem for the Jacobi matrix

$$J_n(w) = \begin{bmatrix} \alpha_0 & \sqrt{\beta_1} & & & \text{O} \\ \sqrt{\beta_1} & \alpha_1 & \sqrt{\beta_2} & & \\ & \sqrt{\beta_2} & \alpha_2 & \ddots & \\ & & \ddots & \ddots & \sqrt{\beta_{n-1}} \\ \text{O} & & & \sqrt{\beta_{n-1}} & \alpha_{n-1} \end{bmatrix} \tag{3}$$

has become the basis of current methods for generating these quadratures. The most popular of them is one due to Golub and Welsch [6]. Their method is based on determining the eigenvalues and the first components of the eigenvectors of a symmetric tridiagonal Jacobi matrix (3), where α_ν and β_ν, $\nu = 0,1,\ldots,n-1$, are the coefficients in the three-term recurrence relation (2) for the monic orthogonal polynomials $\pi_\nu(w;\cdot)$. Namely, *the nodes x_k in the Gaussian quadrature rule (1), with respect to the weight function $w(x)$ on $[a,b]$, are the eigenvalues of the n-th order Jacobi matrix (3). The weights A_k are given by*

$$A_k = \beta_0 v_{k,1}^2, \qquad k = 1,\ldots,n,$$

where $\beta_0 = \mu_0 = \int_a^b w(x)\,dx$ and $v_{k,1}$ is the first component of the normalized eigenvector $\mathbf{v}_k$ corresponding to the eigenvalue x_k,

$$J_n(w)\mathbf{v}_k = x_k\mathbf{v}_k, \qquad \mathbf{v}_k^{\mathrm{T}}\mathbf{v}_k = 1, \qquad k = 1,\ldots,n.$$

Simplifying QR algorithm so that only the first components of the eigenvectors are computed, Golub and Welsch [6] gave an efficient procedure for constructing the Gaussian quadrature rules. This procedure was implemented in several programming packages including our package `OrthogonalPolynomials` realized in `Mathematica` [2].

Thus, we need the recursion coefficients α_k and β_k, $k \le N-1$, for the monic polynomials $\pi_\nu(w;\cdot)$, in order to construct the n-point Gauss-Christofell quadrature formula, with respect to the weighte $w(x)$, for each $n \le N$. These coefficients are known explicitly for the classical orthogonal polynomials (see [12, Chapters 2]). In other cases we need an additional numerical construction of recursion coefficients, using the method of moments or the so-called discretized Stieltjes procedure (see [12, § 2.4.8]).

2.2. Gaussian formulas for the weight $w(x) = (1-x)^\alpha x^\beta \log(1/x)$

We consider the n-point quadrature formula

$$\int_0^1 f(x)(1-x)^\alpha x^\beta \log\frac{1}{x}\,dx = \sum_{k=1}^n A_k f(x_k) + R_n(f), \tag{4}$$

with parameters $\alpha,\beta > -1$ in the weight function $w(x) = (1-x)^\alpha x^\beta \log(1/x)$. Piessens and Branders [19] considered cases when $\alpha = 0$ and $\beta = 0, \pm 1/2, \pm 1/3, -1/4, -1/5$ (see also Gautschi [4] and [5]). Quadrature parameters for $n \le 8$ was given in Katsikadelis [9, pp. 297–298] in the case $\alpha = \beta = 0$.

Using symbolic integration we find the moments $\mu_k = \mu_k(\alpha,\beta)$ in terms of the gamma function and harmonic numbers,

$$\mu_k(\alpha,\beta) = \int_0^1 x^k w(x)\,dx = \int_0^1 (1-x)^\alpha x^{k+\beta}\log\frac{1}{x}\,dx$$

$$= \frac{\Gamma(\alpha+1)\Gamma(\beta+k+1)}{\Gamma(\alpha+\beta+k+2)}\,[H(\alpha+\beta+k+1)-H(\beta+k)]\,. \qquad (5)$$

For example, for $\alpha=\beta=0$ it reduces to $\mu_k(0,0)=1/(k+1)^2$, $k\geq 0$.

The standard meaning of the k-th harmonic number H_k is the sum of the reciprocals of the first k natural numbers, i.e.,

$$H_k = H(k) = \sum_{\nu=1}^{k}\frac{1}{\nu}\,,$$

and its representation is given by Euler in the form

$$H(k) = \int_0^1 \frac{1-t^k}{1-t}\,dt = \sum_{\nu=1}^{k}(-1)^{\nu-1}\frac{1}{\nu}\binom{k}{\nu}.$$

Taking a fractional argument x between 0 and 1, the harmonic number $H(x)$ is defined by the previous integral, where k is simply replaced by x. Then it can be generated by $H(x)=H(x-1)+x^{-1}$ or

$$H(1-x)-H(x) = \pi\cot(\pi x)-\frac{1}{x}+\frac{1}{1-x}.$$

More generally, for every $x>0$ (integer or not), the harmonic number is determined by

$$H(x) = x\sum_{k=1}^{+\infty}\frac{1}{k(x+k)} = \psi(x+1)+\gamma,$$

where $\psi(x)=\Gamma'(x)/\Gamma(x)$ is the so-called *digamma* function, i.e., the logarithmic derivative of the gamma function $\Gamma(x)$ and $\gamma=0.577215664901532\ldots$ is the Euler-Mascheroni constant.

Using the `Mathematica` package `OrthogonalPolynomials` [2] and the first $2N$ moments μ_k, $k=0,1,\ldots,2N-1$, given by (5), we get the first N coefficients α_k and β_k, $k=0,1,\ldots,N-1$, in the recurrence relation (2). It enables us to obtain quadrature parameters in (4) for any $n\leq N$.

Remark. In order to overcome the severe ill-conditioning in obtaining the recursion coefficients with a satisfactory accuracy, a multi-precision arithmetic can be used. For example, in the simplest case $\alpha=\beta=0$, taking 55-decimal-digit arithmetic we get the first $N=50$ recursion coefficients to about 20 decimal digits.

The following code in the `Mathematica` package `OrthogonalPolynomials` [2] generates recursion coefficients for $k\leq 2N-1=99$ and quadrature parameters (nodes and weights) to 20 decimal digits for $n=10(10)50$:

```
In[1]:= << orthogonalPolynomials`

In[2]:= w[t_, a_, b_] := (1 - t)^a t^b Log[1 / t]

In[3]:= mom = Integrate[t^k w[t, 0, 0], {t, 0, 1}]; moments = Table[mom, {k, 0, 99}];

In[4]:= {alpha, beta} = aChebyshevAlgorithm[moments, WorkingPrecision → 55];

In[5]:= param = Table[aGaussianNodesWeights[n, alpha, beta,
             Precision → 20, WorkingPrecision → 20], {n, 10, 50, 10}];
```

For example, the obtained nodes and weights for $n=10$ are given in the following list:

In[6]:= `param[[1]]`

```
Out[6]= {{0.009042630962199650636 9, 0.053971266222500629504,
  0.13531182463925077487, 0.24705241628715982422, 0.38021253960933233397,
  0.52379231797184320116, 0.66577520551642459722, 0.79419041601196621736,
  0.89816109121900353817, 0.96884798871863353939},
 {0.12095513195457051499, 0.18636354256407187033, 0.19566087327775998271,
  0.17357714218290692084, 0.13569567299548420167, 0.093646758538110525987,
  0.055787727351415874076, 0.027159810899233331146,
  0.0095151826028485149993, 0.0016381576335982632549}}
```

Example 1. Consider

$$I = \int_0^1 \frac{(1-x)^{-1/2}x^{-1/2}\log(1/x)}{\sqrt{1+x}}\,dx, \tag{6}$$

which value is known (cf. [5])

$$I = \frac{\sqrt{2\pi}}{8}\Gamma\left(\frac{1}{4}\right)^2 = 4.11871837492687201436674 0\ldots$$

By the linear transformation $2x-1=t$, this integral reduces to

$$I = \int_{-1}^1 \sqrt{\frac{2}{3+t}}\log\frac{2}{1+t}\frac{dt}{\sqrt{1-t^2}}.$$

An application of the standard Gauss-Legendre quadrature gives a very slow convergence. Realtive errors $r_n(\mathrm{GL})$ for $n = 10(10)100$ are presented in Table 1. Numbers in parentheses denote decimal exponents. Slightly better results can be obtained by using Gauss-Chebyshev quadratures with respect to the weight function $w(t) = (1-t^2)^{-1/2}$. The corresponding relative errors $r_n(\mathrm{GC})$ are also displayed in the same table.

However, we can directly apply the quadrature formula (4) to integral (6). Let

$$Q_n^{(\alpha,\beta)} = \sum_{k=1}^n A_k f(x_k) \quad \text{and} \quad r_n^{(\alpha,\beta)} = |(Q_n^{(\alpha,\beta)} - I)/I|.$$

Taking the quadrature formula with the logarithmic weight $w(x) = \log(1/x)$, the corresponding function in (6) is $f(x) = 1/\sqrt{x(1-x^2)}$. The convergence of this rule is again very slow. Relative errors $r_n^{(0,0)}$ are given in Table 1.

But, if we include also algebraic singularities in the weight, i.e., if we take $w(x) = (1-x)^{-1/2}x^{-1/2}\log(1/x)$ $(\alpha = \beta = -1/2)$, the convergence becomes very fast. Gaussian approximations $Q_n^{(-1/2,-1/2)}$ and relative errors are given in the second part of Table 1 for small values of $n \le 10$. Incorrect decimal digits are underlined. As we can see, 17 exact decimal digits are obtained using Gaussian rule with only $n = 10$ digits.

The same method enables us to include also a logarithmic singularity at $x = 0$. Thus, we can consider the weight function

$$w(x) = w^{(\alpha,\beta)}(x) = (1-x)^\alpha x^\beta \log\frac{1}{x(1-x)}, \quad \alpha,\beta > -1.$$

In Fig. 1 we present this weight function for $\alpha = 0$ and three selected values of the parameter β.

Table 1. Relative errors of quadrature sums for $n = 10(10)100$ and Gaussian approximations with respect to logarithmic weight and the corresponding relative errors for $n = 1(1)10$

n	$r_n(\mathrm{GL})$	$r_n(\mathrm{GC})$	$r_n^{(0,0)}$	n	$Q_n^{(-1/2,-1/2)}$	$r_n^{(-1/2,-1/2)}$
10	1.84(−1)	5.29(−2)	1.42(−1)	1	$4.\underline{0801983843688532}$	9.35(−3)
20	1.08(−1)	2.64(−2)	8.69(−2)	2	$4.11\underline{79039770237825}$	1.98(−4)
30	7.80(−2)	1.76(−2)	6.41(−2)	3	$4.118\underline{6986430715864}$	4.79(−6)
40	6.17(−2)	1.32(−2)	5.14(−2)	4	$4.11871\underline{78694526636}$	1.23(−7)
50	6.17(−2)	1.06(−2)	4.31(−2)	5	$4.1187183\underline{615750484}$	3.24(−9)
60	5.14(−2)	8.81(−3)	3.73(−2)	6	$4.118718374\underline{5672496}$	8.73(−11)
70	4.41(−2)	7.55(−3)	3.30(−2)	7	$4.1187183749\underline{170540}$	2.38(−12)
80	3.88(−2)	6.61(−3)	2.96(−2)	8	$4.118718374926\underline{6013}$	6.57(−14)
90	3.47(−2)	5.87(−3)	2.69(−2)	9	$4.1187183749268\underline{644}$	1.83(−15)
100	2.87(−2)	5.29(−3)	2.47(−2)	10	$4.11871837492687\underline{18}$	5.10(−17)

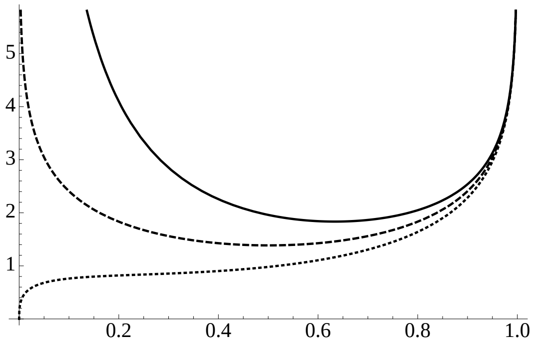

Figure 1. Graphs of the weight functions $w(x)$ for $\alpha = 0$ and $\beta = -1/2$ (solid line), $\beta = 0$ (dashed line) and $\beta = 1/2$ (dotted line).

Similarly as before we find the corresponding moments

$$\begin{aligned}\mu_k(\alpha,\beta) &= \int_0^1 (1-x)^\alpha x^{k+\beta} \log \frac{1}{x(1-x)}\, dx \\ &= \frac{\Gamma(\alpha+1)\Gamma(\beta+k+1)}{\Gamma(\alpha+\beta+k+2)} \left[2H(\alpha+\beta+k+1) - H(\beta+k) - H(\alpha)\right].\end{aligned}$$

Example 2. For $\alpha = -1/4$ and $\beta = -1/2$ compute

$$I_k = \int_{-1}^1 f_k(x) w^{(\alpha,\beta)}(x)\, dx \approx Q_n^{(\alpha,\beta)}(f_k), \qquad k = 1, 2,$$

where $f_1(t) = \sin(10\pi x)$ and $f_2(t) = \sin(20\pi x^2)$.

As before, by using the `Mathematica` package `OrthogonalPolynomials` [2] we obtain recursion coefficients and parameters of the Gaussian rules with respect to the weight $w^{(-1/4,-1/2)}(x)$, and then we apply them to given integrals.

In the first case we get results presented in Table 2, including the corresponding relative errors. Incorrect decimal digits are underlined.

Table 2. Gaussian quadrature sums $Q_n^{(-1/4,-1/2)}(f_k)$, with corresponding relative errors $r_n(f_k),\ k=1,2$

n	$Q_n^{(-1/4,-1/2)}(f_1)$	$r_n(f_1)$	$Q_n^{(-1/4,-1/2)}(f_2)$	$r_n(f_2)$
10	$0.5\underline{022466846173798}$	$5.53(-2)$		
20	$0.531643144401\underline{4815}$	$3.32(-13)$		
30	0.5316431444016578	$2.88(-29)$	$0.4466\underline{5240303668106222}$	$6.99(-5)$
40			$0.446621201\underline{69680683776}$	$3.43(-11)$
50			$0.44662120168147791\underline{272}$	$1.14(-19)$
60			0.44662120168147791267	$2.06(-29)$

Grapfs of the second function $f_2(x)$ and the integrand $F_2(x) = f_2(x)w^{(-1/4,-1/2)}(x)$ are displayed in Fig. 2. Because of oscillatory integrand we need more nodes in integration and therefore we start with $n = 30$ points. Results are given in the same table.

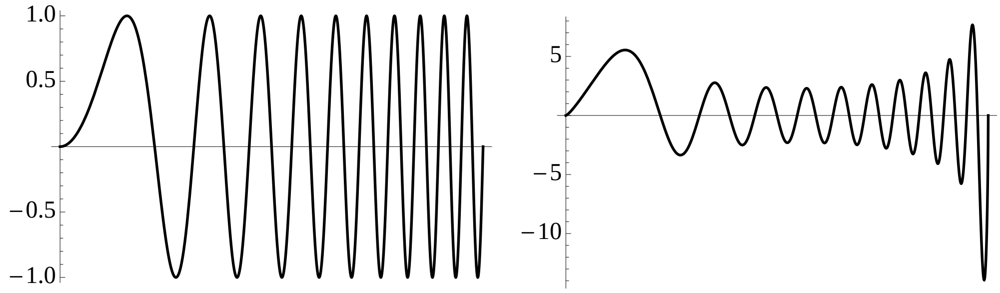

Figure 2. Graphs of functions $f_2(x) = \sin(20\pi x^2)$ (left) and $F_2(x) = f_2(x)w^{(-1/4,-1/2)}(x)$ (right).

3. Some Remarks on Gaussian Quadrature Rules for Müntz Systems

Gaussian integration can be extended in a natural way to non-polynomial functions, taking a system of linearly independent functions

$$\{P_0(x), P_1(x), P_2(x), \ldots\} \qquad (x \in [a,b]), \tag{7}$$

usually chosen to be complete in some suitable space of functions. If $w(x)$ is a given nonnegative weight on $[a,b]$ and the quadrature rule

$$\int_a^b f(x)w(x)\,dx = \sum_{k=1}^{n} A_k f(x_k) + R_n(f) \tag{8}$$

is such that it integrates exactly the first $2n$ functions in (7), we call the rule (8) as *Gaussian with respect to the system* (7). The existence and uniqueness of a Gaussian quadrature rule (8) with respect to the system (7), or shorter a *generalized Gaussian formula*, is always guaranteed if the first $2n$ functions of this system constitute a Chebyshev system on $[a,b]$. Then, all the weights $A_1, \ldots, A_n$ in (8) are positive.

The generalized Gaussian quadratures for Müntz systems goes back to Stieltjes [20] in 1884. Taking $P_k(x) = x^{\lambda_k}$ on $[a,b] = [0,1]$, where $0 \le \lambda_0 < \lambda_1 < \cdots$, Stieltjes showed the existence of Gaussian formulae.

A numerical algorithm for constructing the generalized Gaussian quadratures was investigated by Ma, Rokhlin and Wandzura [11], but their algorithm is ill conditioned (see [11, Remark 6.2]). In [15], Milovanović and Cvetković presented an alternatively numerical method for constructing the generalized Gaussian quadrature (8) for Müntz polynomials, which is exact for each $f \in M_{2n-1}(\Lambda) = \mathrm{span}\{x^{\lambda_0}, x^{\lambda_1}, \ldots, x^{\lambda_{2n-1}}\}$. The method is rather stable and simpler than the previous one, since it is based on construction and stable computation of orthogonal Müntz systems, previously developed in [13]. The method performs calculations in double precision arithmetics in order to get double precision results. For details see [15]. An application in numerical inversion of the Laplace transform was given in [16].

Some transformation methods for integrals with Müntz polynomials can be found in [14] and [10].

4. Conclusion

In this paper we propose a method for construction weighted Gaussian quadrature rules for integrals with algebraic and/or logarithmic singularities, which appear in many applications of BEM and FEM in computational problems in engineering. Also we give some remarks on generalized Gaussian formulas for Müntz systems.

Acknowledgement. The authors were supported in part by the Serbian Ministry of Education and Science.

References

[1] Brebbia C A and Dominguez J (1998) *Boundary Elements: An Introductory Course*, Second Edition, WIT Press, Boston.

[2] Cvetković A S and Milovanović G V (2004) The Mathematica Package “OrthogonalPolynomials”, *Facta Univeritatis Series Mathematics & Informatics*, **19**, pp. 17–36.

[3] Davis P J and Rabinowitz P (1984) *Methods of Numerical Integration* (2nd edn.), Computer Science and Applied Mathematics, Academic Press Inc., Orlando, FL.

[4] Gautschi W (1990) Computational aspects of orthogonal polynomials, *Orthogonal Polynomials (Columbus, OH, 1989)*, P. Nevai (Ed.), pp. 181–216, NATO Adv. Sci. Inst. Ser. C: Math. Phys. Sci., 294, Kluwer, Dordrecht.

[5] Gautschi W (2010) Gauss quadrature routines for two classes of logarithmic weight functions, *Numerical Algorithms*, **55**, pp. 265–277.

[6] Golub G H and Welsch J H (1969) Calculation of Gauss quadrature rules, *Mathematics of Computation*, **23**, pp. 221–230.

[7] Johnston P R and Elliott D (2005) A sinh transformation for evaluating nearly singular boundary element integrals, *International Journal for Numerical Methods in Engineering*, **62**, pp. 564–578.

[8] Johnston B M, Johnston P R, and Elliott D (2007) A sinh transformation for evaluating two-dimensional nearly singular boundary element integrals, *International Journal for Numerical Methods in Engineering*, **69**, pp. 1460–1479.

[9] Katsikadelis J T (2002) *Boundary Elements: Theory and Applications*, Elsevier, Amsterdam.

[10] Lombardi G (2009) Design of quadrature rules for Müntz and Müntz-logarithmic polynomials using monomial transformation, *International Journal for Numerical Methods in Engineering*, **80**, pp. 1687–1717.

[11] Ma J, Rokhlin V, and Wandzura S (1996) Generalized Gaussian quadrature rules for systems of arbitrary functions, *SIAM Journal on Numerical Analysis*, **33**, pp. 971–996.

[12] Mastroianni G and Milovanović G V (2008) *Interpolation Processes – Basic Theory and Applications*, Springer Monographs in Mathematics, Springer – Verlag, Berlin – Heidelberg.

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

[14] Milovanović G V and Cvetković A S (2002) Numerical integration of functions with logarithmic end point singularity, *Facta Univeritatis Series Mathematics & Informatics*, **17**, pp. 57–74.

[15] Milovanović G V and Cvetković A S (2005) Gaussian type quadrature rules for Müntz systems, *SIAM Journal on Scientific Computing* **27**, pp. 893–913.

[16] Milovanović G V and Cvetković A S (2005) Numerical inversion of the Laplace transform, *Facta Univeritatis Series Electronics & Energetics*, **18**, pp. 515–530.

[17] Milovanović G V and Velicković D M (2001) Quadrature processes for several types of quasi-singular integrals appearing in the electromagnetic field problems, *5th Intern. Conf. Applied Electromagnetics PES 2001, Proceedings of Full Papers (Niš, 2001)*, D M Velicković, (Ed.), pp. 117–122, Faculty of Electronic Engineering, Niš.

[18] Oliveira S P, Madureira A L, and Valentin F (2009) Weighted quadrature rules for finite element methods, *Journal of Computational and Applied Mathematics*, **227**, pp. 93–101.

[19] Piessens R and Branders M (1975) *Tables of Gaussian Quadrature Formulas*, University of Leuven, Leuven.

[20] Stieltjes T J (1884) Sur une généralisation de la théorie des quadratures mécaniques, *Comptes Rendus Mathématique, Académie des Sciences, Paris*, **99**, pp. 850–851.