INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2000; **00**:1–6 *Prepared using* nmeauth.cls *[Version: 2002/09/18 v2.02]*

Design of quadrature rules for Müntz and Müntz-logarithmic polynomials using Monomial Transformation

Guido Lombardi*

Dipartimento di Elettronica, Politecnico di Torino, 10129 Torino, Italy.

SUMMARY

A method for constructing the *exact* quadratures for Müntz and Müntz-logarithmic polynomials is presented. The algorithm does permit to anticipate the precision (machine precision) of the numerical integration of Müntz-logarithmic polynomials in terms of number of Gauss-Legendre quadrature samples and monomial transformation order. To investigate in depth the properties of classical Gauss-Legendre quadrature we present new optimal asymptotic estimates for the remainder. In boundary element integrals this quadrature rule can be applied to evaluate singular functions with end-point singularity, singular kernel as well as smooth functions. The method is numerically stable, efficient, easy to be implemented. The rule has been fully tested and several numerical examples are included.

KEY WORDS: *numerical integration; error estimates; Müntz polynomials; nonlinear coordinate transformation; singular integrals; boundary element method; finite element method*

1. INTRODUCTION

The Boundary Element Method (BEM) and Finite Element Method (FEM) often require the numerical evaluation of multiple integrals with singular kernels and/or singular basis functions, see for example fracture mechanics and electromagnetic diffraction. The singularity is typically of two kinds: logarithmic functions and/or nonclassical polynomials as Müntz polynomials, see [1, 2] and references therein.

The Gauss-Legendre quadrature is not efficient in the evaluation of such singular integrals, therefore specific quadrature have been studied.

The classical Gauss quadrature method (1814) has been extended to non-polynomials functions, *i.e.* a system of linearly independent functions. The systematic treatment of arbitrary weight functions using orthogonal polynomials and the definition of the generalized Gaussian quadrature was developed by Christoffel in 1877 and by Stieltjes in 1884 [3]. Several authors

*Correspondence to: Guido Lombardi, Dipartimento di Elettronica, Politecnico di Torino, Corso Duca degli Abruzzi, 24, 10129 Torino, Italy, e-mail: guido.lombardi@polito.it

Contract/grant sponsor: NATO Science for Peace Programme grant CBP.MD.SFPP 982376

Received May 14, 2008
Revised

have defined algorithms to obtain the generalized Gaussian quadrature of important systems of functions, see for example [4]-[9].

The algorithms of generalized Gaussian quadrature for nonclassical polynomials (in particular logarithmic functions or Müntz polynomials) are often ill conditioned, see [7], or in general need some effort to be implemented, see [9]. In particular the algorithm of [9] is stable and it is based on the high precision numerical evaluation of orthogonal Müntz polynomials to safely construct quadrature rules. However it is practically impossible to list all the tables of nodes and weights for all kind of Müntz systems and the computational cost of run-time generalized Gaussian quadrature is high.

Our paper aims to present a new strategy to define quadratures readily available for BEM/FEM codes where the integrands are Müntz polynomials or/times higher order logarithmic functions.

This method (in the following called *monomial quadrature rule*) is based on the study of the capabilities of the classical Gauss-Legendre quadrature to integrate nonclassical (singular) polynomial functions using monomial transformations [16]-[28] of noninteger transformation order. We propose a quadrature scheme that allows to anticipate the precision of the numerical integration of Müntz and Müntz-logarithmic polynomials (see section 2.1 for definitions) in terms of number of Gauss-Legendre quadrature samples and monomial transformation order.

To investigate in depth the properties of classical Gauss-Legendre quadrature we present new optimal asymptotic estimates for the remainder [10]-[16].

The proposed quadrature can be applied to a 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, finite methods, etc. These functions are efficiently approximated using polynomial series, logarithmic functions and/or Müntz polynomials.

The proposed quadrature shows rapid convergence, positive weights and quick implementation without high computational cost, even if they are not "optimal" in the sense of generalized Gaussian quadrature. We have extensively used this technique in computational electromagnetics for diffraction problems, see for example [29].

This paper is organized as follows. Section 2 presents the Müntz polynomials in $(0,1)$, their extension and the basic definitions to represent the remainder for Gaussian quadrature. Asymptotic estimates of the remainder for Gauss-Legendre quadrature are presented in section 3 with integrands of logarithmic and/or Müntz kinds. Section 4 is devoted to the design of the *monomial quadrature rule*. Finally, in section 5 we present several numerical tests to validate the proposed quadrature scheme. In the Appendix we report some special sets of Müntz polynomials with their properties.

The reader could first read sections 4.2 and 5 to immediately appreciate the advantages of the *monomial quadrature rule* for Müntz and Müntz-logarithmic polynomials (see the other sections for definitions).

2. Mathematical background

2.1. Müntz and Müntz-logarithmic polynomials

We recall the definition of Müntz polynomials $M(\Lambda_n)$ which are linear combinations (*span*) of a Müntz system for a given sequence of real numbers Λ_n where $\lambda_k \leq \lambda_{k+1}$, $k \in \mathbb{N}_0, k \leq n$, see

for a full description [1, 2] and references therein:

$$\Lambda_n = \{\lambda_1, \lambda_2, ..., \lambda_n\}, \quad \lambda_k > -1, \quad k = 1..n \tag{1}$$

$$M(\Lambda_n) = span\{x^{\lambda_1}, x^{\lambda_2}, ..., x^{\lambda_n}\} \tag{2}$$

We define $\lambda_{\min} = \min(\lambda_k) > -1$ and $\lambda_{\max} = \max(\lambda_k)$ for a given Λ_n.

We define the Müntz-logarithmic polynomials as the combination of two sub-systems. The first subset is classical Müntz system, the second subset is constituted by the linear combinations of Müntz system times $\log(x)$ function (where log is the natural logarithm); *i.e.* the Müntz-logarithmic polynomials are constituted by terms of the following kind: $\{x^\lambda, x^\lambda \log(x)\}$. We can extend the Müntz-logarithmic polynomials to higher order logarithmic terms, *i.e.* the generalized monomial is: $x^\lambda[\log(x)]^\mu$ with $\lambda > -1$ and $\mu \in \mathbb{N}$.

We define the extended Müntz-logarithmic polynomials $M(\Lambda_{n,m})$ which are linear combinations (*span*) of a system for a given sequence of pairs of number (one real and one integer) $\Lambda_{n,m}$:

$$\Lambda_{n,m} = \{(\lambda_1, \mu_1), ..., (\lambda_k, \mu_h), ..., (\lambda_n, \mu_m)\} \tag{3}$$

where $\lambda_k > -1, \quad k = 1..n$ and $\mu_h \in \mathbb{N}, \quad h = 1..m$.

$$M(\Lambda_{n,m}) = span\{x^{\lambda_1}[\log(x)]^{\mu_1}, ..., x^{\lambda_k}[\log(x)]^{\mu_h}, ..., x^{\lambda_n}[\log(x)]^{\mu_m}\} \tag{4}$$

We define $\lambda_{\min} = \min(\lambda_k)$, $\lambda_{\max} = \max(\lambda_k)$, $\mu_{\min} = \min(\mu_h)$ and $\mu_{\max} = \max(\mu_h)$ for a given $\Lambda_{n,m}$. Note that the Müntz-logarithmic polynomials and the Müntz polynomials are particular cases of the extended Müntz-logarithmic polynomials.

In this paper we intend to construct an efficient quadrature algorithm for these polynomials and we observe that the polynomials presented here include and extend the functions presented in [7, 9] where the generalized Gaussian quadrature is applied.

2.2. Definitions for quadrature rules

The problem of numerical quadrature is to evaluate efficiently the integral $I(f)$ by a sum $I_n(f)$ that satisfies a target precision in terms of relative error $R_n(f)$ (for example, the machine-double precision *d.p.*[†]).

$E_n(f)$ is the remainder, *i.e.* the error committed in integrating an analytic function $f(x)$ times a non-negative weight-function $w(x)$ by an arbitrary integration rule given in terms of n weights w_k and n sample points x_k.

$$I(f) = \int_a^b w(x)f(x)dx = \sum_{k=1}^{n} w_k f(x_k) + E_n(f) \tag{5}$$

[†]In the following, without loss of generality, we consider the machine precision the one defined for IEEE Double-Precision Format [30] in 32-bit programming which yields a floating-point relative accuracy $eps = 2^{-52} = 2.2204 * 10^{-16}$, $\log_{10}(eps) = -15.6536$.

$$I_n(f) = \sum_{k=1}^{n} w_k f(x_k) \tag{6}$$

$$E_n(f) = I(f) - I_n(f) \tag{7}$$

$$R_n(f) = \frac{E_n(f)}{I(f)} \tag{8}$$

In the framework of Gaussian quadrature formulas a basic expression for the remainder is provided in [10] as a contour integral (9):

$$E_n(f) = \frac{1}{2\pi i} \oint_C \frac{\Pi_n(s)}{P_n(s)} f(z) ds \tag{9}$$

where the contour C is chosen to enclose the open interval (a, b), $P_n(s)$ are the orthogonal polynomials with weight function $w(s)$ and

$$\Pi_n(s) = \int_a^b \frac{P_n(t) w(t)}{t - s} dt \quad s \in \mathbb{C} \backslash [a, b] \tag{10}$$

This formula is extended and applied to regular functions and functions with end-point singularity, see [11]-[18] and references therein.

3. Asymptotic estimates of the remainder for Gauss-Legendre quadrature rule

In this section we review the asymptotic estimates of the remainder for Gauss-Legendre quadrature available in literature, see for example [10]-[16]. In particular we define estimates that are valid for Müntz-logarithmic polynomials with non-integer degree λ up to and higher than the Gaussian limit $2n - 1$ (where n is the number of sample points).

We consider Gaussian quadrature in the interval $(a, b) = (0, 1)$ with weight-function $w(x) = 1$. It yields that $P_n(x)$ and $\Pi_n(x)$ are respectively the Legendre polynomials and the Legendre functions of the second kind, besides the $\{w_k, x_k\}$ are referred to the classical Gauss-Legendre quadrature formula in $(0, 1)$. We assume that the integrand function $f(x)$ is a Müntz or a Müntz-logarithmic polynomial, see section 2.1. Since the analytical continuation of the integrand $f(x)$ has a branch point singularity at the origin and $\Pi_n(s)/P_n(s)$ is analytic in $\mathbb{C}\backslash[0, 1]$, we can evaluate (9) along the plane cut constituted by the real axis from 0 to $-\infty$.

In the following we assume the problem of the quadrature in the interval $(0, 1)$ for two reasons: 1) the Müntz polynomials are defined for $x > 0$ and their quadrature is usually considered in $(0, 1)$, 2) the numerical quadrature of nonclassical singular polynomials suffers from the effect of numerical cancellation when the singularity is located out of the origin, as the quadrature points tends to be clustered around the singular point.

3.1. Müntz polynomials

By using the procedure proposed in [12, 16] we present a new estimate of the remainder (9) for Müntz monomial. First of all we reframe the evaluation of the integral of a monomial x^λ

in $(0,1)$ in the interval $(-1,1)$ by using the real mapping $x=(1-y)/2$:

$$I(x^\lambda)=\int_0^1 x^\lambda dx = 2^{-(1+\lambda)}\int_{-1}^1 (1-y)^\lambda dy = \frac{1}{1+\lambda}, \qquad x=\frac{1-y}{2} \tag{11}$$

We observe that, due to the change of interval, the integral and the remainder of the Müntz monomial are $2^{-(1+\lambda)}$ times the one presented in [16], but, of course, it yields the same relative error (8).

In fact, from (9) we obtain:

$$E_n(x^\lambda)=\frac{1}{2\pi\imath}\oint_{Cs}\frac{\Pi_n(s)}{P_n(s)}s^\lambda ds = \frac{2^{-(1+\lambda)}}{2\pi\imath}\oint_{Cz}\frac{\Pi_n(z)}{P_n(z)}(1-z)^\lambda dz \tag{12}$$

where $x=Re[s]$, $y=Re[z]$, $s=(1-z)/2$ is the complex mapping and the contours path C_s and C_z are reported in Figure 1.

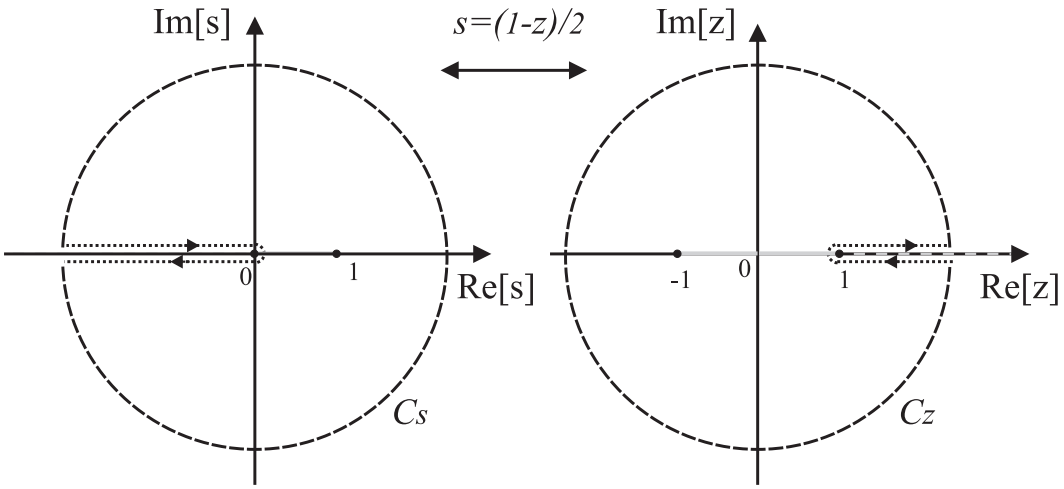

Figure 1. The contours C_s and C_z used to evaluate the remainder (9) for Müntz polynomials.

The integrand $(1-z)^\lambda$ has a branch point singularity at the point 1 and $\Pi_n(z)/P_n(z)$ is analytic in $\mathbb{C}\backslash[-1,1]$. We evaluate (12) along the plane cut constituted by the real axis ($y=Re[z]$) from 1 to $+\infty$, see [12, 16]. We recall and reframe (7) of [16]:

$$E_n(f) = -2^{-(1+\lambda)}\,2\sin(\pi\lambda)J_1 \tag{13}$$

where:

$$J_1=\frac{1}{2\pi}\int_1^{+\infty}\frac{\Pi_n(y)}{P_n(y)}(y-1)^\lambda dy \simeq J_{1a}=\int_1^{+\infty}(y+\sqrt{y^2-1})^{-(2n+1)}(y-1)^\lambda dy \tag{14}$$

$$J_{1a}=\int_0^{+\infty} e^{-(2n+1)\xi}(2(\sinh^2(\xi/2))^\lambda \sinh\xi d\xi \tag{15}$$

with $y = \cosh(\xi)$. The fundamental assumptions in (14) are $y > 1$ and $n \gg 1$ in order to obtain the one term approximation for the ratio $\Pi_n(z)/P_n(z)$ reported in [17]. Note that:

$$\frac{1}{2\pi}\frac{\Pi_n(y)}{P_n(y)} \simeq e^{-(2n+1)\xi} \tag{16}$$

We observe that J_{1a} converges if $\lambda < 2n$. While [16] suggests to use $\sinh(\xi) \approx \xi$ to estimate (15) by obtaining

$$J_{1a}^{(JE)} = 2^{-\lambda}(1+2n)^{-2-2\lambda}\Gamma(2+2\lambda) \tag{17}$$

we evaluate exactly (15) without approximations:

$$J_{1a} = 2^{-\lambda}\,\lambda\left(\frac{Beta(2\lambda, 2\mathrm{n}-\lambda)}{2\mathrm{n}+\lambda} - \frac{Beta(2\lambda, 2+2\mathrm{n}-\lambda)}{2+2\mathrm{n}+\lambda}\right) \tag{18}$$

where $Beta(\cdot,\cdot)$ is the special function:

$$Beta(z,w) = \int_0^1 t^{z-1}(1-t)^{w-1}dt = \frac{\Gamma(z)\Gamma(w)}{\Gamma(z+w)} \tag{19}$$

By using (18), we obtain a new estimate of the remainder (13) that shows a wider λ-range of validity even for modest values of n.

Figure 2 shows the relative error $R_n(f)$ (8) derived from (13) with (18), the actual relative error using tabulated Gauss-Legendre formulas and the relative error derived from the remainder presented in [16] (labelled J-E) using (17). In order to show the convergence it has been used absolute values, 31-digits precision (Quad-Precision Format) and decimal logarithmic scale.

The new estimate (13) defined with (18) is valid for $\lambda > -1$ and in particular for $\lambda > 2n-1$ which is the Gaussian limit for *exact* integration. Note that the integral (15) does not converge for $\lambda \geq 2n$, therefore we extend the domain of J_{1a} using the closed-form explicit expression (18).

The estimate is zero for $0 \leq \lambda \leq 2n-1, \lambda \in \mathbb{N}$, thus it agrees with the theory of Gauss-Legendre quadrature (see negative peaks in Figure 2).

We observe that for an appropriate range of n the relative error is lower than the (IEEE Double-Precision Format, see footnote † at the beginning of section 2.2) floating point relative accuracy for non-integer values of $\lambda \in [0, 2n-1]$ and even for non-integer values of $\lambda > 2n-1$. The estimate of the remainder can be used to find efficiently the range of λ as a function of n to obtain a fixed target precision of relative error.

This property is used to design *exact* quadrature rule for Müntz polynomials using the monomial transformation, see section 4. The exactness is establish for a fixed working precision (for example, we consider IEEE double precision).

3.2. Müntz-logarithmic polynomials

In this section we extend the procedure to find new optimal estimates to Müntz-logarithmic polynomials revising the estimate proposed in [16]. First of all we reframe the evaluation of the integrals of the monomial $x^\lambda \log(x)$ over $x \in (0,1)$ by taking the integration interval $y \in (-1,1)$:

$$I(x^\lambda \log(x)) = \int_0^1 x^\lambda \log(x)dx = 2^{-(1+\lambda)}\int_{-1}^1 (1-y)^\lambda\left[\log(1-y) - \log(2)\right]dy = -\frac{1}{(1+\lambda)^2} \tag{20}$$

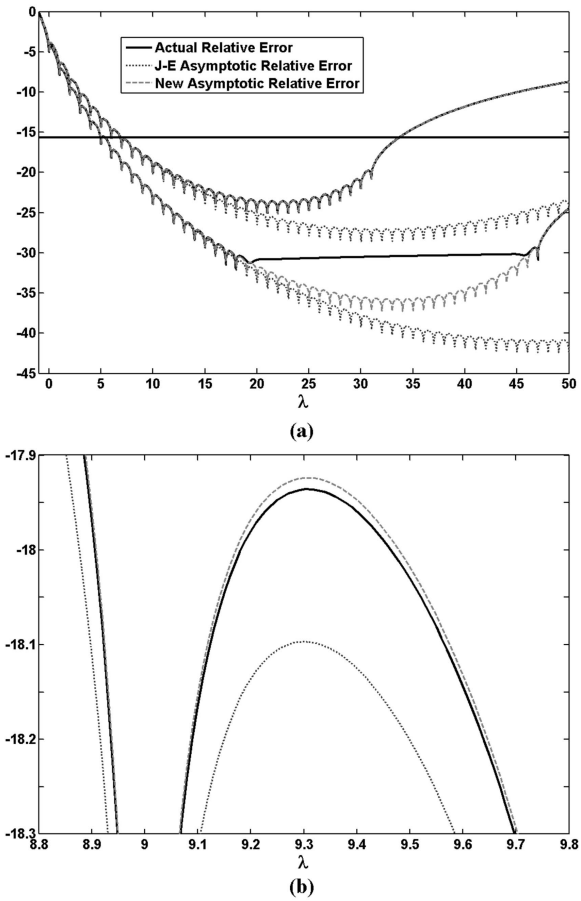

Figure 2. Müntz polynomials: a) Plots of the relative error and its estimates are reported in $\log_{10}$ scale for $n = 16, 24$. We have reported the actual value, the estimated value derived from (13) with (17) [16, 18] (with J-E label) and the value obtained by (13) with (18). We observe that the actual value and the estimate using (18) have approximately coincident values for any λ, as clearly shown from the detailed view reported in b). In a) the constant black solid line is the double precision (IEEE Format) relative accuracy level $log_{10}(eps) = -15.6536$, $eps = 2^{-52}$. The actual value is limited to 31-digits precision.

where $x = (1-y)/2$. The remainder can be written as the sum of two terms:

$$E_n(x^\lambda \log(x)) = \frac{2^{-(1+\lambda)}}{2\pi\imath} \oint_C \frac{\Pi_n(z)}{P_n(z)} \log(1-z)(1-z)^\lambda dz - \frac{2^{-(1+\lambda)}\log(2)}{2\pi\imath} \oint_C \frac{\Pi_n(z)}{P_n(z)} (1-z)^\lambda dz \quad (21)$$

While the second integral has been estimated in the previous sub-section, we need to find an optimal estimate for the first one:

$$\frac{1}{2\pi\imath} \oint_C \frac{\Pi_n(z)}{P_n(z)} \log(1-z)(1-z)^\lambda dz \quad (22)$$

The integrand $\log(1-z)(1-z)^{\lambda}$ has a branch point singularity at the point 1 and $\Pi_n(z)/P_n(z)$ is analytic in $\mathbb{C}\backslash[-1,1]$. We evaluate (22) along the plane cut constituted by the real axis ($y = Re[z]$) from 1 to $+\infty$ and it becomes:

$$-\cos(\pi\lambda)\int_1^{+\infty}\frac{\Pi_n(y)}{P_n(y)}(y-1)^{\lambda}dy-\frac{\sin(\pi\lambda)}{\pi}\int_1^{+\infty}\frac{\Pi_n(y)}{P_n(y)}\log(y-1)(y-1)^{\lambda}dy \tag{23}$$

It yields:

$$E_n(x^{\lambda}\log(x)) = 2^{-(1+\lambda)}\{-2\pi\cos(\pi\lambda)J_1 - 2\sin(\pi\lambda)J_2\} - 2^{-(1+\lambda)}\log(2)\{-2\sin(\pi\lambda)J_1\} \tag{24}$$

where the brackets highlight the first and the second part of (21), J_1 is defined in (14), and

$$J_2 = \frac{1}{2\pi}\int_1^{+\infty}\frac{\Pi_n(y)}{P_n(y)}\log(y-1)(y-1)^{\lambda}dy \simeq J_{2a} = \int_1^{+\infty}(y+\sqrt{y^2-1})^{-(2n+1)}\log(y-1)(y-1)^{\lambda}dy \tag{25}$$

J_2 is approximated using the one term approximation for the ratio $\Pi_n(z)/P_n(z)$ when $y > 1$ and $n \gg 1$ [17].

It is useful to apply the change of variable $y = \cosh(\xi)$ to (25):

$$J_{2a} = \int_0^{+\infty} e^{-(2n+1)\xi}\log(2\sinh^2(\xi/2))(2(\sinh^2(\xi/2))^{\lambda}\sinh\xi d\xi \tag{26}$$

We observe that J_{2a} converges if $\lambda < 2n$.

We propose also a compact approximated formulation of (24) which uses the definition of $\widetilde{J}_{2a}$ instead of J_{2a}:

$$\widetilde{J}_{2a} = J_{2a} - J_{1a}\log(2) = \int_0^{+\infty} e^{-(2n+1)\xi}\log(\sinh^2(\xi/2))(2(\sinh^2(\xi/2))^{\lambda}\sinh\xi d\xi \tag{27}$$

and it yields:

$$E_n(x^{\lambda}\log(x)) \simeq 2^{-(1+\lambda)}\{-2\pi\cos(\pi\lambda)J_1 - 2\sin(\pi\lambda)\widetilde{J}_{2a}\} \tag{28}$$

Also $\widetilde{J}_{2a}$ converges if $\lambda < 2n$.

We propose two approximations of the remainder $E_n(x^{\lambda}\log(x))$ with different range of validity: 1) type A is valid for "small" λ ($\lambda \lesssim n$), 2) type B is valid for "non-small" λ ($\lambda \gtrsim n$).

Type A: by applying the approximation $\sinh\xi \simeq \xi$ to (26) and using the definition (17) we obtain

$$J_{2a}^{(A)} = 2^{-\lambda+1}\int_0^{+\infty} e^{-(2n+1)\xi}\log(\xi)\xi^{2\lambda+1}d\xi - J_{1a}^{(JE)}\log(2) = \tag{29}$$

$$J_{2a}^{(A)} = -2^{1-\lambda}(1+2n)^{-2-2\lambda}\Gamma(2+2\lambda)\left[\log(1+2n) - \Psi(2+2\lambda) + \frac{1}{2}\log(2)\right] \tag{30}$$

where $\Psi(z)$ denotes the digamma function. Using (24), (30) and (18) we obtain the the following approximation of the remainder (type A):

$$E_n^{(A)}(x^\lambda \log(x)) = 2^{-(1+\lambda)}\{[-2\pi\cos(\pi\lambda) + 2\log(2)\sin(\pi\lambda)]J_{1a} - 2\sin(\pi\lambda)J_{2a}^{(A)}\} \tag{31}$$

The use of (28) instead of (24) with $\sinh\xi \simeq \xi$ yields a worse estimation of the remainder.

Type B: we approximate the argument of the logarithm $\sinh\xi \simeq e^\xi/2$ in (27):

$$\widetilde{J}_{2a}^{(B)} = 2^\lambda \int_0^{+\infty} e^{-(2n+1)\xi}\xi(\sinh(\xi/2))^{2\lambda}\sinh(\xi)d\xi \tag{32}$$

$$\widetilde{J}_{2a}^{(B)} = 2^\lambda \lambda\,\Gamma(2\lambda)[\Upsilon_1 + \Upsilon_2] \tag{33}$$

where

$$\begin{cases} \Upsilon_1 = \frac{\Gamma(2+2n-\lambda)}{\Gamma(3+2n+\lambda)}\left[\frac{1}{2n-\lambda} + \frac{1}{1+2n-\lambda} - \frac{1}{1+2n+\lambda} - \frac{1}{2+2n+\lambda} + \Psi(2n-\lambda) - \Psi(1+2n+\lambda)\right] \\ \Upsilon_2 = \frac{\Gamma(2n-\lambda)}{\Gamma(1+2n+\lambda)}\left[-\Psi(2n-\lambda) + \Psi(1+2n+\lambda)\right] \end{cases} \tag{34}$$

Using (28), (33) and (18) we obtain the the following approximation of the remainder (type B):

$$E_n^{(B)}(x^\lambda \log(x)) = 2^{-(1+\lambda)}\{-2\pi\cos(\pi\lambda)J_{1a} - 2\sin(\pi\lambda)\widetilde{J}_{2a}^{(B)}\} \tag{35}$$

The use of (24) instead of (28) with $\sinh\xi \simeq e^\xi/2$ yields a worse estimation of the remainder.

Figure 3 shows the relative error $R_n(f)$ derived from type A estimate (31), the actual relative error using tabulated Gauss-Legendre formulas and the relative error derived from (24) using the approximations presented in [16] (labelled J-E). In order to show the convergence of the estimates it has been used absolute values, 31-digits precision and decimal logarithmic scale.

In particular the new estimate (type A) is valid for $\lambda \lesssim n$ and it shows some spurious ripples for $\lambda > 2n-1$ which is the Gaussian limit for *exact* integration fo polynomials. Note that (26) does not converge for $\lambda \geq 2n$. Figure 3b shows, in detail, that the envelope of the type A estimate is good, however the zeros (negative peaks) are shifted in positions. The estimate (J-E) shows errors both in the envelope and in the position of the zeros of the relative error, in particular when λ is with significant values.

Figure 4 shows the relative error $R_n(f)$ derived from type B estimate (35), and the actual relative error using tabulated Gauss-Legendre formulas.

In particular the new estimate (type B) is valid for $\lambda \gtrsim n$ and it does not show any spurious ripples for $\lambda > 2n-1$ in comparison with type A estimate.

The new estimate (35) defined with (33) is valid for $\lambda > -1$ and moreover for $\lambda > 2n-1$. Note that the integral (27) does not converge for $\lambda > 2n$, therefore we extend the domain of validity of $\widetilde{J}_{2a}^{(B)}$ using closed-form explicit expression as (33).

Figure 4b shows good agreement between type B estimate and the actual relative error ($\lambda \gtrsim n$), even for significant values of λ. On the contrary Figure 4c shows, for small values of λ, that the envelope of the type B estimate is not good as the type A, and the zeros (negative peaks) are shifted in positions.

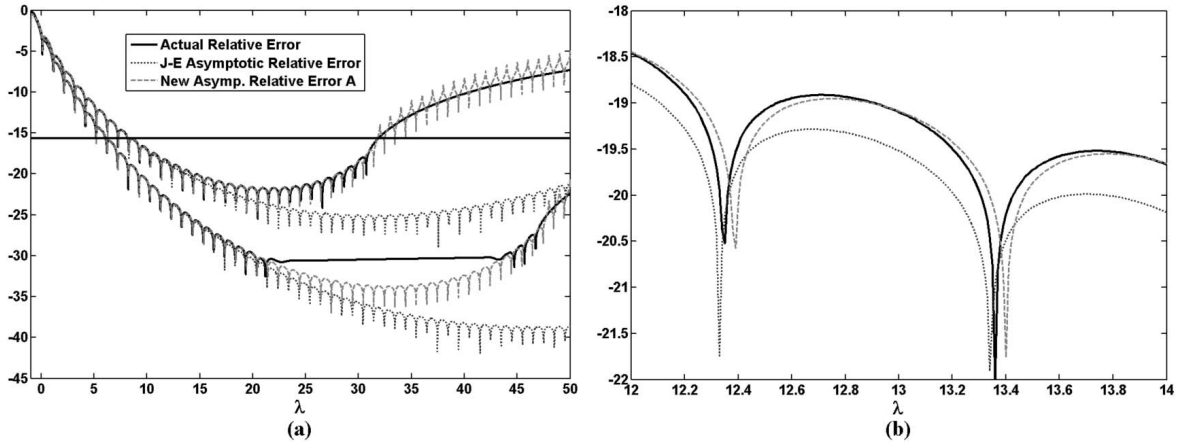

Figure 3. Müntz-logarithmic polynomials: a) Plots of the relative error and its estimates are reported in log_{10} scale for $n = 16, 24$. We have reported the actual value, the estimated value derived using the approximations of [16, 18] (with J-E label) and the value obtained from type A estimate (31). We observe that the actual value and the estimate (31) have approximately coincident envelopes for $\lambda < 2n-1$ value, as clearly shown from the detailed view reported in b). In a) the constant black solid line is the double precision (IEEE Format) relative accuracy level $log_{10}(eps) = -15.6536$, $eps = 2^{-52}$. The actual value is limited to 31-digits precision.

We observe that for an appropriate range of n the relative error is lower than the IEEE Double-Precision Format floating point relative accuracy for non-integer values of $\lambda \in [0, 2n-1]$ and even for non-integer values of $\lambda > 2n-1$. The estimate of the remainder can be used to find efficiently the range of λ as a function of n to obtain a fixed target precision of relative error.

This property is used to design *exact* quadrature rule for Müntz-logarithmic polynomials using the monomial transformation, see section 4. The exactness is establish for a fixed working precision (for example, we consider IEEE double precision).

3.3. Extended Müntz-logarithmic polynomials

In this section we illustrate the procedure to find new optimal estimates to extended Müntz-logarithmic polynomials, *i.e.* polynomials with monomials of $x^\lambda[\log(x)]^\mu$ kind with $\lambda > -1$ and $\mu \in \mathbb{N}$. The exact integral of a monomial is:

$$I(x^\lambda[\log(x)]^\mu) = \int_0^1 x^\lambda[\log(x)]^\mu dx = (-1)^\mu \frac{\mu!}{(1+\lambda)^{\mu+1}} \tag{36}$$

From (9) we obtain the following expression of the remainder:

$$E_n(x^\lambda[\log(x)]^\mu) = \frac{1}{2\pi\imath} \oint_{Cs} \frac{\Pi_n(s)}{P_n(s)} s^\lambda[\log(s)]^\mu ds \tag{37}$$

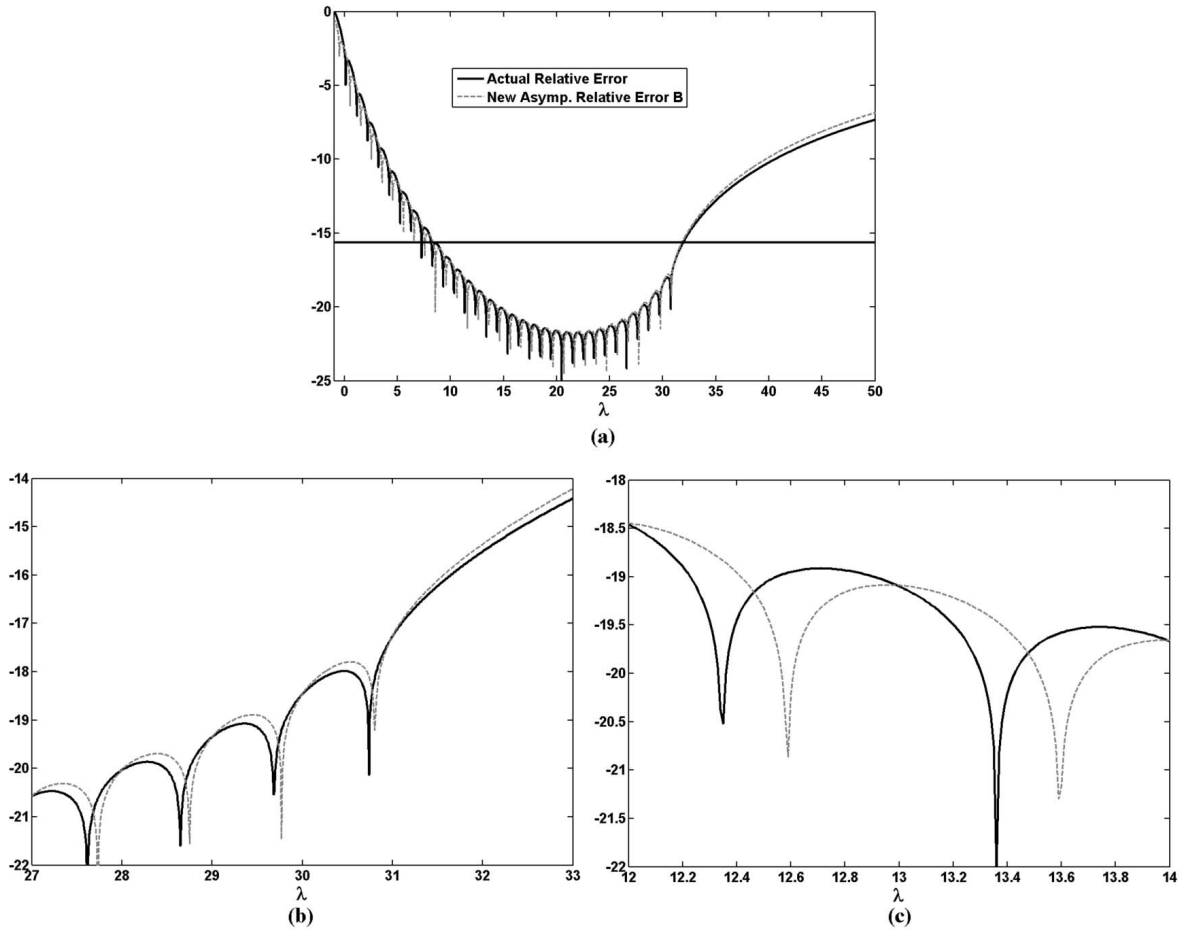

Figure 4. Müntz-logarithmic polynomials: a) Plots of the relative error and its estimates are reported in log_{10} scale for $n = 16$. We have reported the actual value, and the value obtained from type B estimate (35). We observe that the actual value and the estimate (35) have almost coincident values for $\lambda \gtrsim n$ value, as clearly shown from the detailed view reported in b). c) The position of zeros (negative peaks) are rather shifted for $\lambda \lesssim n$ in comparison with the ones identified by type A estimate.

We define the complex function $f(s)$ as the analytical continuation of the integrand $x^\lambda[\log(x)]^\mu$:

$$f(s) = s^\lambda[\log(s)]^\mu = |s|^\lambda e^{\imath\theta\lambda}\,[\log(|s|) + \imath\theta]^\mu \tag{38}$$

where $\theta = arg(s)$. Since $f(s)$ has a branch point singularity at the origin and $\Pi_n(s)/P_n(s)$ is analytic in $\mathbb{C}\backslash[0,1]$, we evaluate (37) along the plane cut constituted by the real axis from 0 to $-\infty$ (see Figure 1):

$$E_n(x^\lambda[\log(x)]^\mu) = \int_{-\infty}^{0} \frac{\Pi_n(x)}{P_n(x)}(-x)^\lambda M_{\lambda,\mu}(x)dx \tag{39}$$

where

$$M_{\lambda,\mu}(x) = \frac{1}{2\pi\imath}\left\{e^{\imath\pi\lambda}\left[log(-x)+\imath\pi\right]^{\mu} - e^{-\imath\pi\lambda}\left[log(-x)-\imath\pi\right]^{\mu}\right\} \tag{40}$$

and using the binomial expansion:

$$M_{\lambda,\mu}(x) = \frac{1}{\pi}\sum_{k=0}^{\lfloor\mu/2\rfloor}\binom{\mu}{2k}(\imath\pi)^{\mu-2k}\log(-x)^{2k}T_{\mu}(\lambda) + \binom{\mu}{2k+1}(\imath\pi)^{\mu-2k-1}\log(-x)^{2k+1}T_{\mu+1}(\lambda) \tag{41}$$

where

$$T_{\mu}(\lambda) = \begin{cases} \sin(\pi\lambda) & even\ \mu \\ \cos(\pi\lambda)/\imath & odd\ \mu \end{cases} \tag{42}$$

We define:

$$\widehat{J}_{\mu} = -\frac{2^{\lambda+1}}{2\pi}\int_{-\infty}^{0}\frac{\Pi_n(x)}{P_n(x)}(-x)^{\lambda}[\log(-x)]^{\mu}dx \tag{43}$$

Note that the previously defined J_1 and $\widetilde{J}_2$ are $\widehat{J}_{\mu}$ respectively with $\mu = 0, 1$. From (39), using (40)-(43), we obtain:

$$E_n(x^{\lambda}[\log(x)]^{\mu}) = \frac{1}{\pi}\sum_{k=0}^{\lfloor\mu/2\rfloor}\binom{\mu}{2k}(\imath\pi)^{\mu-2k}T_{\mu}(\lambda)\widehat{J}_{2k} + \binom{\mu}{2k+1}(\imath\pi)^{\mu-2k-1}T_{\mu+1}(\lambda)\widehat{J}_{2k+1} \tag{44}$$

In order to give explicit estimates of (44) we propose two approximations of $\widehat{J}_{\mu}$ with different range of validity: 1) type A is valid for "small" λ ($\lambda \lesssim n$), 2) type B is valid for "non-small" λ ($\lambda \gtrsim n$). First of all we apply to (43) the change of variable $x = -(\sinh(\xi/2))^2$ and we use the approximation (16):

$$\widehat{J}_{\mu} \simeq \widehat{J}_{\mu a} = 2^{\lambda+1}\int_{0}^{+\infty} e^{-(2n+1)\xi}(\sinh(\xi/2))^{2\lambda}[\log(\sinh(\xi/2)^2)]^{\mu}\frac{\sinh(\xi)}{2}d\xi \tag{45}$$

We observe that $\widehat{J}_{\mu a}$ converges if $\lambda < 2n$.

Type A: we approximate the argument of the logarithm $\sinh\xi \simeq \xi$ in (45)

$$\widehat{J}^{(A)}_{\mu a} = 2^{-\lambda+\mu}\int_{0}^{+\infty} e^{-(2n+1)\xi}\xi^{2\lambda}[\log(\xi/2)]^{\mu}d\xi \tag{46}$$

Explicit expression of (46) are numerical stable and easy to be computed for $\mu \in \mathbb{N}$ in terms of the gamma function $\Gamma(z)$, digamma function $\Psi(z)$ and its n^{th} derivatives $\Psi(n,z)$ with $n = 0..\mu-1$. For example for $\mu = 2$:

$$\begin{aligned}\widehat{J}^{(A)}_{2a} &= \tfrac{2^{2-\lambda}\Gamma(2+2\lambda)}{(1+2n)^{-2(1+\lambda)}}\cdot \\ &\cdot\left\{\log(2)^2 + \log(1+2n)\,\log(4+8n) - 2\,\log(2+4n)\,\Psi(2+2\lambda) + \Psi(2+2\lambda)^2 + \Psi(1,2+2\lambda)\right\}\end{aligned} \tag{47}$$

From (44) and (46) we obtain the first type (type A) of the estimate of the remainder, labelled $E^{(A)}_n(x^{\lambda}[\log(x)]^{\mu})$.

Type B: we apply the approximation $\sinh\xi \simeq e^{\xi}/2$ to (45) only to the logarithmic argument:

$$\widehat{J}_{\mu a}^{(B)} = 2^{\lambda+\mu} \int_0^{+\infty} e^{-(2n+1)\xi} \xi (\sinh(\xi/2))^{2\lambda} [\xi/2 - \log(2)]^{\mu} \sinh(\xi) d\xi \tag{48}$$

and using the binomial expansion it yields:

$$\widehat{J}_{\mu a}^{(B)} = 2^{\lambda+\mu} \sum_{k=0}^{\mu} \binom{\mu}{k} (-\log(2))^{\mu-k} \int_0^{+\infty} e^{-(2n+1)\xi} \xi (\sinh(\xi/2))^{2\lambda} [\xi/2]^{k} \sinh(\xi) d\xi \tag{49}$$

Explicit expression of (49) are computed for $\mu \in \mathbb{N}$ in terms of the gamma function $\Gamma(z)$, digamma function $\Psi(z)$ and its n^{th} derivatives $\Psi(n, z)$ with $n = 0..\mu - 1$.

From (44) and (49) we obtain the second type (type B) of the estimate of the remainder, labelled $E_n^{(B)}(x^{\lambda}[\log(x)]^{\mu})$.

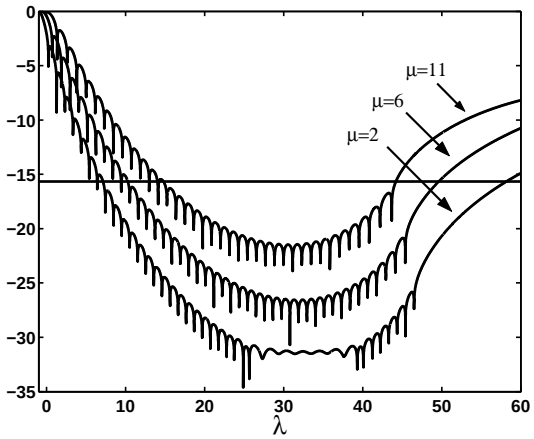

Figure 5. Plots of the actual relative error in log_{10} scale for extended Müntz-logarithmic monomials with $\mu = 2, 6, 11$ using 24 samples Gauss-Legendre quadrature.

Figure 5 shows the actual relative error for Müntz-logarithmic monomials with $\mu = 2, 6, 11$ obtained by using Gauss-Legendre quadrature with 24 samples. As expected, the error increases by increasing μ.

Figures 6a and 6b show the relative error derived from type A estimate and the actual relative error using tabulated Gauss-Legendre formulas (24 samples) and with $\mu = 6$. In order to show the convergence of the proposed estimates it has been used absolute values, 31-digits precision and decimal logarithmic scale.

In particular the new estimate (type A) is valid for "small" λ ($\lambda \lesssim n$) and it shows spurious ripples for $\lambda > 2n - 1$. Note that the type A estimate does not converge for $\lambda > 2n$.

Figures 6c and 6d show the relative error derived from type B estimate and the actual relative error using tabulated Gauss-Legendre formulas (24 samples) and with $\mu = 6$.

In particular, unlike type A estimate, the type B estimate is valid for $\lambda \gtrsim n$.

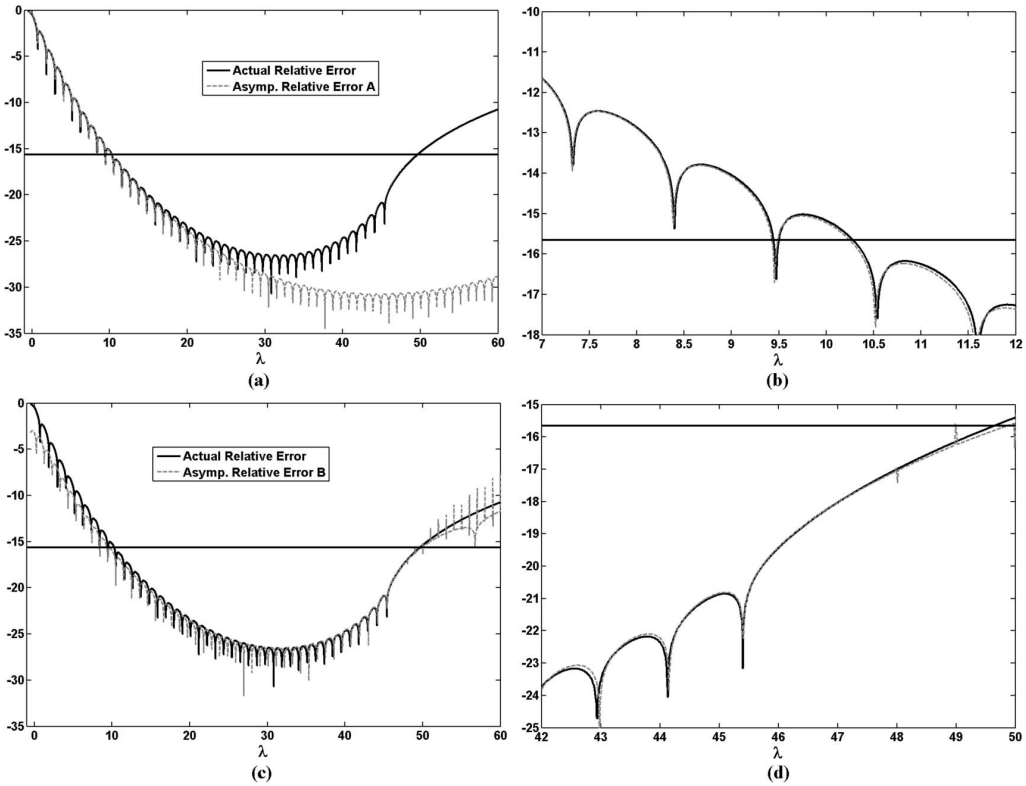

Figure 6. Extended Müntz-logarithmic polynomials: a-b) Plots of the relative error and its type A estimate reported in log_{10} scale for $\mu = 6$ and using Gauss-Legendre quadrature with 24 samples. c-d) Plots of the relative error and its type B estimate reported in log_{10} scale for $\mu = 6$ and using Gauss-Legendre quadrature with 24 samples.

The new estimate is valid for $\lambda > -1$ and in particular for $\lambda > 2n-1$. Note that the integral (45) does not converge for $\lambda > 2n$, therefore we extend the domain of $\widehat{J}_{\mu a}$ by using closed-form explicit expression of (49) that are available through symbolic computational software and/or table of integrals [31].

Figure 6d shows good agreement between type B estimate and the actual relative error, even for significant values of λ: some convergence problem is experienced only for significant integer values of λ (see spurious spikes for $\lambda = 48, 49, 50...$).

We observe that for an appropriate range of n the relative error is lower than the double-precision floating point relative accuracy for non-integer values of $\lambda \in [0, 2n-1]$ and even for non-integer values of $\lambda > 2n-1$. The estimate of the remainder can be used to find efficiently the range of λ as a function of n to obtain a fixed target precision of relative error.

As for the other kinds of polynomials, this property will be used to design *exact* quadrature rule for extended Müntz-logarithmic polynomials using the monomial transformation.

4. Design of Monomial transformations

We recall the well-known in literature monomial transformation:

$$\gamma_r(t) = t^r \tag{50}$$

where the order of the transformation r is a real positive number. This transformation maps the interval $(0,1)$ onto itself. This transformation is related to the $(1/m)th$ sigmoidal transformation as $m \to +\infty$, see [19]. The monomial transformation is also a particular kind of polynomial transformation that have been used in quadrature schemes for evaluating integrands with endpoint singularity since two decades [20], see for details [18]-[28] and references therein.
Our work examines the capability of the monomial transformation using Gauss-Legendre quadrature and it provides an algorithm to anticipate the machine precision for the numerical integration of extended Müntz-logarithmic polynomials.
By applying the monomial transformation (50) to the monomial $x^\lambda[\log(x)]^\mu$ (extended Müntz-logarithmic polynomial) we obtain:

$$\int_0^1 x^\lambda[\log(x)]^\mu dx = \int_0^1 r^{\mu+1}\, t^{r\lambda+r-1}\, [\log(t)]^\mu dt \tag{51}$$

Note that the integrand in t is again an extended Müntz-logarithmic monomial with μ logarithmic order and $(r\lambda + r - 1)$ Müntz order.
The design of the quadrature is based on the choice of r in order to shift the monomial orders in the region of convergence for a fixed target precision. Given an extended Müntz-logarithmic polynomial and a target precision for the numerical integration we need to:

- define λ_{min}, λ_{max}, μ_{min}, μ_{max} in relation to the polynomial, see section 2;
- define $\beta_{min}^{(n)}$, $\beta_{max}^{(n)}$ as the lower and upper bound of β to obtain a relative error (8) in the evaluation of $\int_0^1 x^\beta[\log(x)]^\mu dx$ below the target precision as functions of n (number of samples), *i.e.* the range of $\beta^{(n)}$ that yields the relative error $R_n(x^\beta[\log(x)]^\mu) < eps$ where *eps* is the relative accuracy level;
- therefore the constraints are:

$$\begin{cases} r\lambda_{min} + r - 1 > \beta_{min}^{(n)} \\ r\lambda_{max} + r - 1 < \beta_{max}^{(n)} \end{cases} \tag{52}$$

The quadrature is defined by the pairs of number (n, r) enforcing the previous constraints.

From (52) we obtain that the design of transformation order r must satisfy:

$$\frac{1+\beta_{\min}^{(n)}}{1+\lambda_{\min}} < r < \frac{1+\beta_{\max}^{(n)}}{1+\lambda_{\max}} \tag{53}$$

Once we have selected (n, r) we apply first the monomial transformation and then the classical Gauss-Legendre quadrature to the integral $I(f)$:

$$I(f) = \int_0^1 f(x)dx = \int_0^1 f(t)\, r\, t^{r-1} dt \simeq I_n^r(f) = \sum_{k=1}^{n} w_k^{(x)} f(x_k) = \sum_{k=1}^{n} w_k^{(t)} f(t_k) r\, t_k^{r-1} \tag{54}$$

We obtain the quadrature $\{x_k, w_k^{(x)}\}$ $(k = 1..n)$ in the original domain x using:

$$\begin{cases} x_k = t_k^r \\ w_k^{(x)} = r\, t_k^{r-1} w_k^{(t)} \end{cases} \tag{55}$$

where $\{t_k, w_k^{(y)}\}$ are respectively points and weights of the classical Gauss-Legendre quadrature in $(0, 1)$.

4.1. Lower and upper bound of β

The algorithm presented in the previous section is effective when the computation of lower and upper bound of β as functions of n is efficient (Note that we use β as quadrature design parameters while λ are the exponents of the polynomial to be integrated). These values may be computed and tabulated by using the actual relative error for different values of n but, for a general procedure, they can be derived from the estimates of section 3.

In particular we need to find the β values so that the envelope of the estimates of the relative error is equal (or lower) than the target precision (for example IEEE double precision). In order to satisfy this requirement it is possible to apply standard algorithms to find roots to the envelope of the relative error estimates (Newton's method, secant method ...). First of all we need to find the envelopes of the estimates for the remainders reported in section 3, *i.e.* remove the spikes and the modulation when infinite precision occurs, see Figures 2-6. This is not a trivial task for the general case of extended Müntz-logarithmic polynomials.

From section 3.1 we obtain the envelope of the estimate for classical Müntz polynomials modifying (13):

$$E_n^{env}(x^\beta) = \begin{cases} -2^{-(1+\beta)}\, 2J_{1a} & -1 < \beta < 2n - 1/2 \\ -2^{-(1+\beta)}\, 2\sin(\pi\beta) J_{1a} & \beta \geq 2n - 1/2 \end{cases} \tag{56}$$

Note that the envelope and the estimate for $\beta \geq 2n - 1/2$ coincide.

From section 3.2 we obtain the envelope of the estimate for Müntz-logarithmic polynomials modifying (35):

$$E_n^{env}(x^\beta \log(x)) = \begin{cases} 2^{-(1+\beta)}\{-2\pi J_{1a} - 2\widetilde{J}_{2a}^{(B)}\} & -1 < \beta < 2n - 1/2 \\ 2^{-(1+\beta)}\{-2\pi\cos(\pi\beta) J_{1a} - 2\sin(\pi\beta)\widetilde{J}_{2a}^{(B)}\} & \beta \geq 2n - 1/2 \end{cases} \tag{57}$$

Note that the envelope coincides with the type B estimate for $\beta \geq 2n - 1/2$.

In general for the extended Müntz-logarithmic polynomials we define the envelope of the estimates for the relative error modifying (44) and using type A and B definitions proposed in section 3.3:

$$E_n^{env}(x^\beta [\log(x)]^\mu) = \frac{1}{\pi} \sum_{k=0}^{\lfloor \mu/2 \rfloor} \binom{\mu}{2k} (\imath\pi)^{\mu-2k} \widetilde{T}_\mu(\beta) \widehat{J}_{2k} + \binom{\mu}{2k+1} (\imath\pi)^{\mu-2k-1} \widetilde{T}_{\mu+1}(\beta) \widehat{J}_{2k+1} \tag{58}$$

Table I. Table of the lower and upper bound of β for double precision case.

n	x^β		$x^\beta log(x)$		$x^\beta [log(x)]^3$	
	$\lceil \beta_{min}^{(n)} \rceil$	$\lfloor \beta_{max}^{(n)} \rfloor$	$\lceil \beta_{min}^{(n)} \rceil$	$\lfloor \beta_{max}^{(n)} \rfloor$	$\lceil \beta_{min}^{(n)} \rceil$	$\lfloor \beta_{max}^{(n)} \rfloor$
12	10.55	21.42	-	-	-	-
16	7.48	33.36	8.99	31.59	16.01	29.12
20	6.30	47.87	7.36	44.29	12.75	40.77
24	5.61	65.79	6.47	60.11	10.70	55.22
28	5.14	86.98	5.89	78.82	9.30	74.01
32	4.80	111.42	5.46	100.36	8.05	95.68
36	4.53	139.09	5.14	124.68	7.91	121.06
40	4.31	169.99	4.88	151.77	7.62	143.61
44	4.13	204.10	4.66	181.62	7.34	171.10
48	3.98	241.43	4.48	214.22	7.08	195.16
52	3.85	281.98	4.33	249.56	6.86	222.50
56	3.74	325.74	4.19	287.65	6.66	252.28
60	3.63	372.71	4.08	328.46	6.48	284.24
64	3.54	422.90	3.97	372.01	6.32	318.29
68	3.46	476.31	3.88	418.28	6.18	354.37
72	3.39	532.92	3.79	467.26	6.05	392.43
76	3.32	592.75	3.71	518.97	5.93	432.47
80	3.26	655.79	3.64	573.39	5.82	474.45
84	3.20	722.05	3.58	630.52	5.72	518.36
88	3.15	791.51	3.52	690.36	5.63	564.19
92	3.10	864.19	3.46	752.91	5.54	611.93
96	3.06	940.09	3.41	818.15	5.46	661.57

with

$$\widetilde{T}_\mu(\beta) = \begin{cases} 1 & even\ \mu\,\&\,\beta < 2n-1 \\ -\imath & odd\ \ \mu\,\&\,\beta < 2n-1 \\ \sin(\pi\beta) & even\ \mu\,\&\,\beta \geq 2n-1 \\ \cos(\pi\beta)/\imath & odd\ \ \mu\,\&\,\beta \geq 2n-1 \end{cases} \tag{59}$$

The definition of (44) in terms of $\widetilde{T}_\mu(\beta)$ instead of $T_\mu(\beta)$ remove the spikes giving the envelope of the estimates.

Figures 7 report the envelopes of the relative error estimates and the actual relative errors respectively for (a) Müntz polynomials, (b) Müntz-logarithmic polynomials and (c) extended Müntz-logarithmic polynomials with $\mu = 3$, when $n = 24$ Gauss-Legendre quadrature rule is applied.

In Table I we report an estimation of the lower and upper bound of β versus n for the three type of polynomials under investigation when the target precision is set to be double precision. These values are derived applying the Newton's method to the proposed envelopes.

From Table I we note that, for double precision case, it is possible to derive useful regression curve to fit $\beta_{min}^{(n)}$ and $\beta_{max}^{(n)}$. In particular the following fitting schemes hold for the three kind of polynomials under investigation, see Figure 8.

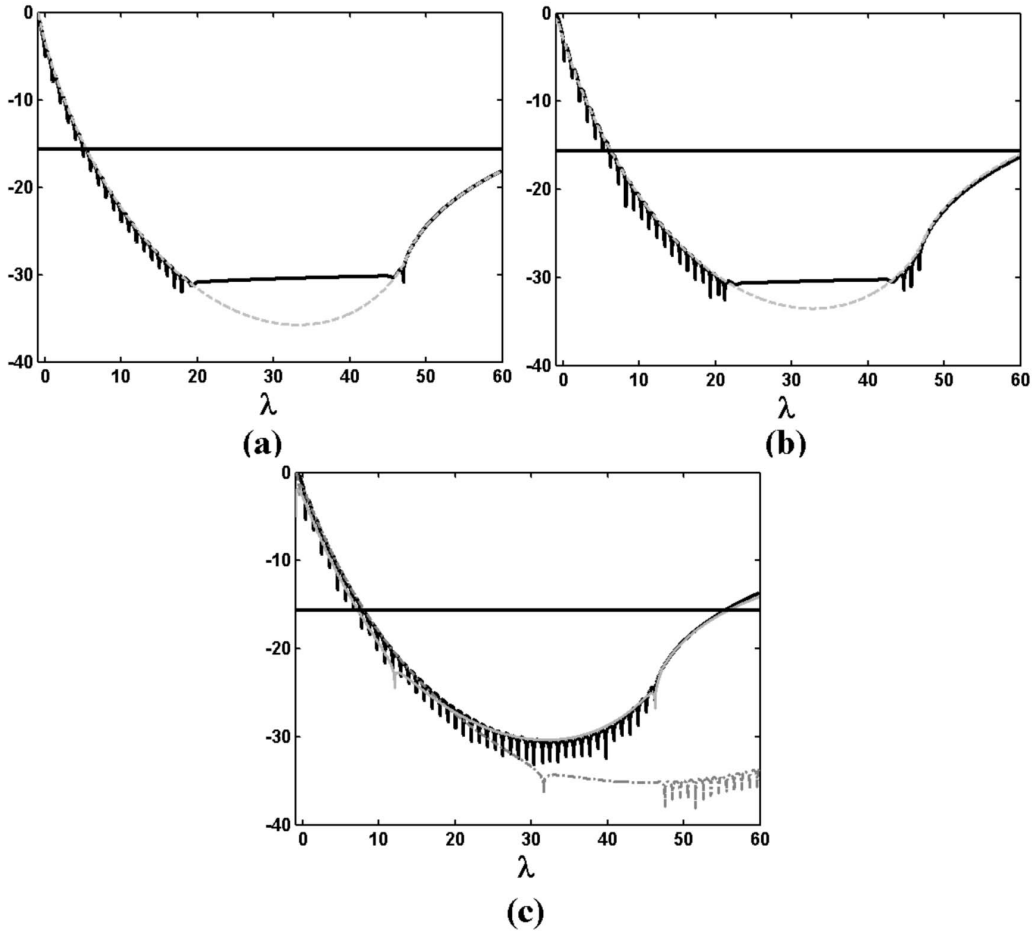

Figure 7. Plots of the envelopes of the relative error estimates (gray lines) and the actual relative errors (black solid line) respectively for (a) Müntz polynomials, (b) Müntz-logarithmic polynomials and (c) extended Müntz-logarithmic polynomials with $\mu = 3$; when $n = 24$ Gauss-Legendre quadrature rule is applied. Note that in Figure (a) and (b) the envelope approximately overlaps the actual relative error except for the ripple. In Figure (c) two envelopes are reported: dashed-dot line is referred to type A estimate which approximately overlaps the actual relative error for $\beta \lesssim n$ except for the ripple, and the gray solid line is referred to type B estimate which approximately overlaps the actual relative error for $\beta \gtrsim n$ except for the ripple.

$$\begin{cases} \beta_{min}^{(n)} & = \sqrt[3]{\frac{1}{c_1 n + c_0}} \\ \beta_{mx}^{(n)} & = d_2 n^2 + d_0 \end{cases} \tag{60}$$

Table II reports the fitting parameters for $\beta_{min}^{(n)}$ and $\beta_{max}^{(n)}$ (60) used in Figure 8.

Note that (60) are particular useful to quickly design the *monomial quadrature rule* for

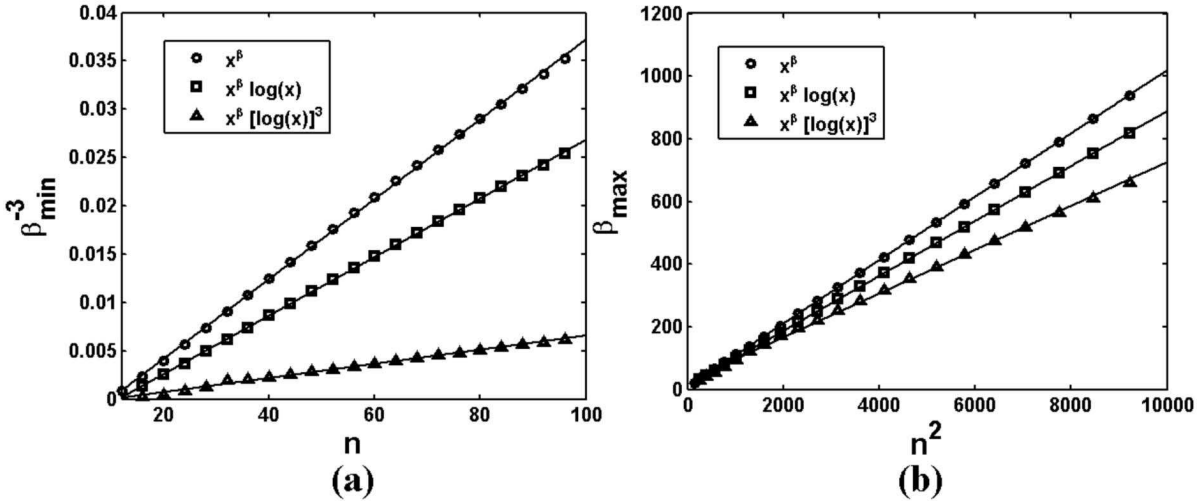

Figure 8. Regression curve (a) β_{min} (b) β_{max}. The scale are chosen properly in order to show the validity of the curve fitting reported in (60).

Table II. Table of fitting parameters for β_{min} and β_{max}.

	c_1	c_0	d_2	d_0
x^β	$4.1296 * 10^{-4}$	$-4.0693 * 10^{-3}$	$1.0123 * 10^{-1}$	7.8147
$x^\beta log(x)$	$3.0285 * 10^{-4}$	$-3.4647 * 10^{-3}$	$8.7825 * 10^{-2}$	$1.0918 * 10^{1}$
$x^\beta [log(x)]^3$	$7.3104 * 10^{-5}$	$-7.4999 * 10^{-4}$	$7.0035 * 10^{-2}$	$2.5611 * 10^{1}$

Müntz, Müntz-logarithmic and extended Müntz-logarithmic polynomials, see next section.

4.2. Design Algorithm for IEEE Double Precision Quadrature

The double precision *monomial quadrature rule* of Müntz, Müntz-logarithmic and extended Müntz-logarithmic polynomials with minimum sampling points can be obtained by following the algorithm described below. The algorithm also gives the monomial transformation order r. Note that the algorithm is valid for polynomials with a single value of μ but this limitation can be overcome as reported in example 5 of section 5.

Step 1 Collect/Select λ_{min} and λ_{max} for the polynomial under investigation.

Step 2 To obtain the minimum number of sampling points, from (53), we enforce the equation

$$\frac{1+\beta_{\min}^{(n)}}{1+\lambda_{\min}} = \frac{1+\beta_{\max}^{(n)}}{1+\lambda_{\max}} \tag{61}$$

and substitute the curve fitting (60) for $\beta_{min}^{(n)}$ and $\beta_{max}^{(n)}$, *i.e.* we solve the following polynomial equation of degree 7:

$$(c_0 + c_1 n)\left[(1 + d_0 + d_2 n^2)(1 + \lambda_{\min}) - 1 - \lambda_{\max}\right]^3 - (1 + \lambda_{\max})^3 = 0 \tag{62}$$

Step 3 The zeros of the polynomial equation are constituted by 3 couple of complex conjugate solutions and 1 real solution (n_r). The smallest integer greater than or equal to the real solution is the minimum number of sampling point for double precision quadrature rule ($n_{\min} = \lceil n_r \rceil$). $n_{\min}$ is always greater than 10 for double precision as $\beta_{min}^{(n)} \leq \beta_{max}^{(n)}$.

Step 4 Substituting n_r (real value) in (61) we obtain the monomial transformation order

$$r = \frac{1 + \beta_{\min}^{(n)}}{1 + \lambda_{\min}} = \frac{1 + \beta_{\max}^{(n)}}{1 + \lambda_{\max}} \tag{63}$$

Step 5 Use (55) and the readily available tables of Gauss-Legendre quadratures to obtain the quadrature.

Step 2 is the basic design constraint: from (52)-(53) we enforce that the transformed minimum/maximim λ order becomes the minimum/maximum $\beta^{(n)}$ values for a fixed target quadrature precision as functions of n.

The algorithm can be generalized for target precision different from double precision and for different values of μ (extended Müntz-logarithmic polynomials) by evaluating specific curve fitting parameters for $\beta_{min}^{(n)}$ and $\beta_{max}^{(n)}$. Besides, if no fitting is available the estimates reported in section 4.1 can be used to find $\beta_{min}^{(n)}$ and $\beta_{max}^{(n)}$.
In the Appendix we report some special sets of Müntz polynomials with their properties.

5. Numerical results

This section demonstrates the efficacy and the efficiency of the proposed algorithm. We reports several numerical examples, some of them are reported in the classical literature, see example 2, 3, and 4. Note that we report only examples where the singularity is located at the origin to avoid cancellation error in quadrature sample points which tend to be clustered near the singular points for high transformation order r. Therefore if we want to integrate a singular function with singularity out of the origin we suggest to reframe the problem in the interval $(0, 1)$. The performance of the proposed quadrature scheme is established in terms of actual relative error (64) where $I_n^r(f)$ is defined in (54):

$$R_n^r(f) = \frac{|I(f) - I_n^r(f)|}{|I(f)|} \tag{64}$$

Without loss of generality, in the numerical examples we often use the most common in practice and readily available in literature Gauss-Legendre rules (for example see pp. 916-917 [32] which reports the quadrature rules for $n > 10$: $n = 12, 16, 20, 24, 32, 40, 48, 64, 80, 96$). In the following examples the target precision is set to be double (see footnote at the beginning of section 2.2 for definition) and *d.p.* denotes that double precision relative accuracy is obtained.

5.1. Example 1

The first example is a simple application of the algorithm described in section 4.2: consider the integral

$$f(x) = 5x^{-\frac{\pi}{4}} - x^{-\frac{1}{2}} + 1 + 10x^2 + e\,x^{e+\frac{1}{4}} \tag{65}$$

$$I(f) = \int_0^1 f(x)dx = \frac{7}{3} + \frac{4e}{4e+5} + \frac{20}{4-\pi} \simeq 26.31729737648832 \tag{66}$$

The integrand is a Müntz polynomial with

$$\Lambda_5 = \{-\frac{\pi}{4}, -\frac{1}{2}, 0, 2, e+\frac{1}{4}\} \tag{67}$$

that contains monomials with singular irrational and rational exponents, classical polynomial terms and monomials with non singular irrational exponent. Following the algorithm of section 4.2, we note that $\lambda_{min} = -\pi/4$ and $\lambda_{max} = e + 1/4$ [Step 1]. Using (62) and Table II [Step 2] we obtain that the required number of sampling point for double precision quadrature is [step 3]:

$$n_{\min} = \lceil n_r \rceil = \lceil 31.284201303977138 \rceil = 32 \tag{68}$$

Using (63) we obtain

$$r = 27.187743291832103 \tag{69}$$

(only the first digits can be considered) [step 4].

We define the quadrature by selecting $n = n_{\min} = 32$ Gauss-Legendre quadrature points to satisfy the requirement (68) and by applying the transformation rule (55) [step5] with (69).

The numerical integration, as expected, gives double precion relative error ($R^r_{32}(f) < 2^{-52} = 2.22 * 10^{-16}$). Besides if we try to apply the algorithm to an $n < n_{\min}$ Gauss-Legendre rule, as expected, we obtain lower precision: selecting 24 Gauss-Legendre points and (69) we obtain the following actual relative error $R^r_{24}(f) = 4.51710^{-11}$. Besides we note that if the the classical Gauss-Legendre rule with 32 samples is applied, we obtain relative error higher than 10^{-1}.

5.2. Example 2

In boundary element application quadratic basis functions times logarithmic kernel need to be integrated. Without loss of generality we reframe the example reported in [18] in the interval $(0, 1)$. The quadratic basis functions are

$$\begin{cases} \phi_0(x) = (-1+x)(-1+2x) \\ \phi_{\frac{1}{2}}(x) = -4x(-1+x) \\ \phi_1(x) = x(-1+2x) \end{cases} \tag{70}$$

and the boundary source integral is:

$$I(\phi_k) = \int_0^1 \log(x)\phi_k(x)dx, \quad k = \{0, \frac{1}{2}, 1\} \tag{71}$$

The integrands are Müntz-logarithmic polynomials, see definition in section 2.1, with

$$\Lambda_{3,1} = \{(0,1), (1,1), (2,1)\} \tag{72}$$

Table III. Table of results for Example 2.

f	$I(f)$	$R^r_{16}(f)$	$R^r_{12}(f)$
$\phi_0(x)\log(x)$	$-17/36$	*d.p.*	$6.43984 * 10^{-8}$
$\phi_{\frac{1}{2}}(x)\log(x)$	$-5/9$	*d.p.*	$1.09477 * 10^{-7}$
$\phi_1(x)\log(x)$	$1/36$	$1.9984 * 10^{-15}$	$1.09477 * 10^{-6}$
$\log(x)$	-1	*d.p.*	$1.08802 * 10^{-14}$
$x\log(x)$	$-1/4$	*d.p.*	$1.88738 * 10^{-15}$
$x^2\log(x)$	$-1/9$	*d.p.*	$1.36847 * 10^{-7}$

Note that $\lambda_{min} = 0$ and $\lambda_{max} = 2$ [Step 1]. The required number of sampling point for double precision quadrature is [step 2 and 3]:

$$n_{\min} = \lceil n_r \rceil = \lceil 15.108671538771373 \rceil = 16 \tag{73}$$

Using (63) we obtain

$$r = 10.655328168802873 \tag{74}$$

(only the first digits can be considered) [step 4].

Therefore we select $n = 16$ Gauss-Legendre quadrature points to satisfy the requirement (73) and we apply the transformation rule (55) [step5] with (74).

In Table III we report the exact value of the integral, the relative error for approximate result using the proposed quadrature and the relative error for approximate result using the proposed quadrature but reducing the number of samples. It is interesting to compare this table of results with Table 7 of [18] where similar transformations with uncontrolled (not designed) transformation orders have been applied (note that $\phi_0(x), \phi_{\frac{1}{2}}(x), \phi_1(x)$ correspond to the J_1, J_4, J_3 of [18]). The rules proposed in Table 7 of [18] do not allow to anticipate the precision. We observe that the designed quadrature performs better than all the quadrature proposed in Table 7 of [18] as the double precision (*d.p.*) is anticipated by the proposed algorithm.

Note that in column 3-row 3 we would expect *d.p.* instead of $1.9984 * 10^{-15}$: the small error is due to cancellation error in sampling $\phi_1(x)$.

5.3. Example 3

This example is reported in [9], where a generalized Gaussian type quadrature rule for Müntz system has been applied. Once the table of nodes and weights for a specific Müntz system is evaluated the algorithm of [9] is more efficient than the one proposed in section 4.2; however it is practically impossible to list all the tables of nodes and weights for all kind of Müntz systems and the computational cost of run-time generalized Gaussian quadrature is high. On the counterpart the algorithm of section 4.2 needs only the classical Gauss-Legendre quadrature readily available in literature.

Table IV. Table of results for Example 3.

$R^r_{32}(f)$	$R^r_{24}(f)$	$R^r_{20}(f)$	rel.err. ref. [9] GG $n = 10$	rel.err. ref. [9] GG $n = 15$	rel.err. ref. [9] GL $n = 100$	rel.err. ref. [9] GL $n = 600$
d.p.	$7.5258 * 10^{-14}$	$7.31675 * 10^{-13}$	$1.3 * 10^{-15}$	*d.p.*	$1.2 * 10^{-3}$	$3.3 * 10^{-5}$

The test integral is:

$$I(f) = \int_0^1 B_0(x)(1+\log(x))dx = {}_1F_2\left(\{\frac{1}{2}\},\{1,\frac{3}{2}\},-\frac{1}{4}\right) - {}_2F_3\left(\{\frac{1}{2},\frac{1}{2}\},\{1,\frac{3}{2},\frac{3}{2}\},-\frac{1}{4}\right) \quad (75)$$

where $B_0(x)$ is the Bessel function and where ${}_pF_q(\mathbf{a},\mathbf{b},z)$ is the generalized hypergeometric function. The integrand can be approximated with a polynomial function of degree 14 times $(1+\log(x))$, therefore we can define $\lambda_{min} = 0$ and $\lambda_{max} = 14$ [Step 1]. The required number of sampling point for double precision quadrature is [step 2 and 3]:

$$n_{\min} = \lceil n_r \rceil = \lceil 31.23058891461433 \rceil = 32 \quad (76)$$

Using (63) we obtain

$$r = 6.505205732881846 \quad (77)$$

(only the first digits can be considered) [step 4].

Therefore we select $n = 32$ Gauss-Legendre quadrature points to satisfy the requirement (76) and we apply the transformation rule (55) [step5].

In Table IV we report the relative error for approximate result using the proposed quadrature and the relative error for approximate result using the proposed quadrature but reducing the number of samples with respect to $n_{\min}$. Besides we report the numerical results of [9] where GG and GL are referred to respectively the generalized Gaussian quadrature rule for the Müntz system under test and the Gauss-Legendre quadrature rule with n samples.

5.4. Example 4

Taking inspiration from Table 4 and 8 of [7] we present the following test case. The integrand is $f(x) = x^{25}x^{-\frac{2}{3}}$. First of all we have to distinguish if we want to integrate in double precision only the monomial $x^{25}x^{-\frac{2}{3}}$ or the Müntz system $M(\Lambda_{26})$ with

$$\lambda_k = k - 2/3, \quad k = 0..25 \quad (78)$$

In the first case (the integrand is just $f(x)$) the proposed algorithm provides:

$$n_{\min} = \lceil n_r \rceil = \lceil 10.240745585502083 \rceil = 11 \quad (79)$$

$$r = 0.7670123938807416 \quad (80)$$

Table V. Table of relative errors for $f(x)$ in Example 4.

n	$R_n^r(f)$	GL	rel.err. ref. [7]
10	$2.20435 * 10^{-13}$	$3.78334 * 10^{-8}$	$1.6142 * 10^{-3}$
12	*d.p.*	$3.56844 * 10^{-13}$	–
15	*d.p.*	*d.p.*	$3.3992 * 10^{-8}$
16	*d.p.*	*d.p.*	–
20	*d.p.*	*d.p.*	$1.0371 * 10^{-14}$

where we have supposed $\lambda_{min} = \lambda_{max} = 73/3$ and therefore the quadrature is "exact" for λ near 73/3. In Table V the actual relative error is reported for $n = 12$ with r defined in (80). Besides we report also the direct application of classical Gauss-Legendre (GL) quadrature (without monomial transformation) to the integrand $f(x)$ with $n = 12, 16$. The last column of Table V reports the numerical results of [7] where a generalized Gaussian quadrature algorithm is used with $n = 10, 15, 20$.

In the second case, let us consider $g(x) \in M(\Lambda_{26})$, *i.e.* $g(x) = (1 + x^{25})x^{-\frac{2}{3}}$. The proposed algorithm provides:

$$n_{\min} = \lceil n_r \rceil = \lceil 58.5235362118078 \rceil = 59 \tag{81}$$

$$r = 14.033995434911114 \tag{82}$$

where we have supposed $\lambda_{min} = -2/3$ and $\lambda_{max} = 73/3$. The first column of Table VI reports the actual relative error using the monomial transformation quadrature with (82) and different number of Gauss-Legendre samples.

In the Appendix we report a simplified scheme to evaluate Müntz polynomials with rational exponent. This technique requires only the classical Gauss-Legendre quadrature tables and it is particularly suitable for Müntz system with rational exponents as $M(\Lambda_{26})$. The main difference from the algorithm proposed in section 4.2 is that the simplified scheme integrates only polynomials constituted by monomials with rational exponents while the the algorithm of section 4.2 integrates polynomials constituted by monomials with real exponents $\lambda_{\min} \leq \lambda \leq \lambda_{\max}$. Using the algorithm of the Appendix we find that $q = 3$, $m_{\max} = -2$ and $k_{\max} = 25$ therefore from (100):

$$n_{\min} = \frac{\gamma_{\max} + 1}{2} \tag{83}$$

and in our case $n_{\min} = 38$. The second column of Table VI reports the actual relative error of a monomial quadrature based on integer monomial transformation order $r = q$ and Gauss-Legendre points and weights. Due to the performance of Gauss-Legendre rules with high number of samples it is possible to reduce the quadrature points to integrate almost "exactly" the integral under test (we can mix the property that we have used to derive the algorithm of 4.2 and the properties of Müntz polynomials with rational exponent). For this purpose we also report in Table VI the actual relative error for $n < n_{\min}$.

Table V and VI also report the actual relative error for the generalized Gaussian quadrature proposed in [7] that must be computed for each Müntz system and the algorithm needs

Table VI. Table of relative errors for $g(x)$ in example 4.

n	$R_n^r(g)$	$R_n^q(g)$	rel.err. ref. [7]
10	$1.15239 * 10^{-2}$	$3.07588 * 10^{-4}$	$2.09641 * 10^{-5}$
15	$4.58915 * 10^{-3}$	$3.87017 * 10^{-7}$	$4.41448 * 10^{-10}$
20	$8.34589 * 10^{-4}$	$1.84626 * 10^{-11}$	*d.p.*
24	$1.32755 * 10^{-4}$	$4.38322 * 10^{-16}$	expected *d.p.*
32	$1.01898 * 10^{-6}$	$2.92215 * 10^{-16}$	expected *d.p.*
40	$1.64433 * 10^{-9}$	*d.p.*	expected *d.p.*
48	$5.57838 * 10^{-13}$	*d.p.*	expected *d.p.*
56	$2.92215 * 10^{-16}$	*d.p.*	expected *d.p.*
64	$4.38322 * 10^{-16}$	*d.p.*	expected *d.p.*

Table VII. Table of relative errors for $h(x)$ in Example 4.

n	$R_n^r(h)$	$R_n^q(h)$	rel.err. ref. [7]
3	0.895479	*d.p.*	–
20	*d.p.*	*d.p.*	$2.31536 * 10^{-5}$

quad-precision for double precision weights and points. We assess, as in Example 3, that it is practically impossible to list all the tables of nodes and weights for all kind of Müntz systems and the computational cost of run-time generalized Gaussian quadrature is high. On the contrary the algorithm of section 4.2 and of the Appendix needs only the classical Gauss-Legendre readily available in literature.

We observe that the simplified algorithm of the Appendix integrate exactly monomials $x^{\lambda_{k,m}}$ where $\lambda_{k,m} = m/q + k$ for each $|m| < q$ (see definition (96)) while [7] integrate exactly only non classical monomials for a fixed value of m together with classical monomials with integer exponents.

Let us consider $h(x) = x^{-\frac{2}{3}} + x^{-\frac{1}{3}} + 1 + x$. Using the design algorithm of section 4.2 we obtain that the double precision can be obtained for $(n_{\min}, r) = (19, 22.376195152034512)$ however in this case the simplified algorithm of the Appendix is more efficient because the quadrature require only $n = 3$ sample, see (83). As expected, [7] fails because the quadrature is designed only for terms with $m = -2$. The results are reported in Table VII.

5.5. Example 5

The last example is an application of the proposed algorithm to extended Müntz-logarithmic polynomials. Let us consider the integrand

$$f_\xi(x) = \left(\frac{1}{\sqrt{x}} + x^4\right)[log(x)]^3 + x^\xi \tag{84}$$

where we distinguish two case: $\xi = 24/5$ and $\xi = 8$. The exact value of the integrals are respectively:

$$I(f_{24/5}) = \int_0^1 f_{24/5}(x)dx = -\frac{1737049}{18125} \simeq -95.83718620689655 \tag{85}$$

$$I(f_8) = \int_0^1 f_8(x)dx = -\frac{539429}{5625} \simeq -95.89848888888889 \tag{86}$$

The integrands are constituted by extended Müntz-logarithmic monomials:

$$\Lambda_{3,2} = \{(-1/2, 3), (4, 3), (\xi, 0)\} \tag{87}$$

Therefore we start to design the quadrature from the monomials with $\mu = 3$ because the constraint is usually more "strict" for monomials with $\mu = \mu_{\max}$. Note that $\lambda_{min} = -1/2$ and $\lambda_{max} = 4$ for $\mu = 3$. Using (62) and Table II we obtain that the required number of sampling point for double precision quadrature is:

$$n_{\min} = \lceil n_r \rceil = \lceil 31.543942878679584 \rceil = 32 \tag{88}$$

Using (63) we obtain

$$r = 19.25944979499394 \tag{89}$$

Therefore we select $n = 32$ Gauss-Legendre quadrature points to satisfy the requirement (88). However, we need to check if the designed quadrature is able to double precision integrate the Müntz term with $\mu \neq \mu_{\max}$, *i.e.* $x^{24/5}$ and x^8. The constraint is defined in (52):

$$r * \lambda_{\max} + r - 1 < \beta_{\max}^{(32)} \tag{90}$$

where $\beta_{\max}^{(32)}$ is the one defined for Müntz polynomials (Table II). The test is satisfied for $x^{24/5}$ (double precision) while it fails for x^8.

The design algorithm (section 4.2) for $\xi = 8$ must be modified in step 2 and 4 using a mixed scheme: we need to consider the extended Müntz-logarithmic polynomial scheme with $\mu = 3$ for the definition of $\lambda_{\min}$ and $\beta_{\min}^{(n)}$, and the Müntz polynomial scheme for the definition of $\lambda_{\max}$ and $\beta_{\max}^{(n)}$. We obtain that the required number of sampling point for double precision quadrature is:

$$\overline{n}_{\min} = \lceil n_r \rceil = \lceil 38.552744936292875 \rceil = 39 \tag{91}$$

Using (63) we obtain

$$\overline{r} = 17.697142289677416 \tag{92}$$

Table VIII reports the relative error for approximate result using the proposed quadrature with $n = 32, 40$ and $r, \overline{r}$ as given in (90)-(92).

A sub-optimal, but all in one go, scheme can be obtained by selecting λ_{min} and λ_{max} independently from μ and design r with $\beta_{\min}^{(n)}$, $\beta_{\max}^{(n)}$ related to $\mu_{\max}$. In this case the double precision quadrature of (84) is guaranteed with $(n_{min}, r) = (35, 18.6213453377283)$ for $\xi = 24/5$ and with $(n_{min}, r) = (43, 17.025407835957978)$ for $\xi = 8$.

6. Conclusion

In this paper we propose a new algorithm to design monomial transformation for Gauss-Legendre quadrature of Müntz and Müntz-logarithmic polynomials. The algorithm does permit to anticipate the precision (machine precision) of the numerical integration of Müntz-logarithmic polynomials in terms of Gauss-Legendre quadrature samples and monomial transformation order. The method is numerically stable, efficient, easy to be implemented and it has low run-time computational cost. The algorithm requires the study in depth of the properties of classical Gauss-Legendre using new asymptotic estimates for the remainder. The proposed rules have been fully tested and several numerical examples are included.

Appendix: Special Müntz polynomials

In this Appendix we report some remarkable results for special classes of Müntz polynomials that are useful in physical-engineering applications. For these classes of polynomials the quadrature design algorithms can be simplified.

Finite energy Müntz polynomials

We define finite energy Müntz, Müntz-logarithmic and extended Müntz-logarithmic polynomials the ones that are L^2 integrable, *i.e.*

$$\|f\|_2 < \infty \tag{93}$$

thus $\lambda_{min} > -1/2$. From (53) the optimal design of the quadrature for this class of polynomials is obtained when we enforce the following value of transformation order ($-1/2$ is the "worst" minimum limit value of λ):

Table VIII. Table of results for Example 5.

f	$R_{32}^{r}(f)$	$R_{40}^{\overline{r}}(f)$
$f_{24/5}(x)$	*d.p.*	*d.p.*
$f_8(x)$	$3.0556 * 10^{-13}$	*d.p.*

$$r^{(n)} = \frac{1+\beta_{\min}^{(n)}}{1+\lambda_{\min}} = 2(1+\beta_{\min}^{(n)}) \tag{94}$$

Note that the previous quantities are defined in terms of the number of quadrature points n (see (60) for double precision).

From (53) the quadrature under design is able to integrate monomial up to the following order:

$$\lambda_{\max}^{(n)} = \frac{\beta_{\max}^{(n)} + 1 - r^{(n)}}{r^{(n)}} = \frac{\beta_{\max}^{(n)} + 1}{2(1+\beta_{\min}^{(n)})} - 1 \tag{95}$$

The design is reduced to the choice of the number of quadrature points (n) and the target precision for the quadrature in order to integrate accurately a finite energy extended Müntz-logarithmic polynomial with highest order $\lambda_{\max}^{(n)}$.
Note that by substituting (60), (95) becomes an explicit function of n and the target precision is set to be double. Figure 9 reports the behavior of $\lambda_{\max}^{(n)}$ for the three polynomials under investigation in section 4.1 (x^λ, $x^\lambda log(x)$, and $x^\lambda [log(x)]^3$). Note that the intersection of the $\lambda_{\max}^{(n)}$ curves with the bisectrix $f(n) = n$ or the line $f(n) = 2n-1$ provides a performance rate of the quadratures: in particular $n_g = 84, 99, 157$ are the intersection abscissas (rounded towards plus infinity) with the bisectrix respectively for x^λ, $x^\lambda log(x)$, and $x^\lambda [log(x)]^3$; and $n_G = 143, 171, 270$ are the intersection abscissas (rounded towards plus infinity) with the line $f(n) = 2n-1$ respectively for x^λ, $x^\lambda log(x)$, and $x^\lambda [log(x)]^3$. Therefore when $n > n_G$ the proposed quadrature evaluates, with double precision, integrals of Müntz polynomials from the order $-1/2$ up to an order greater than $2n-1$ which is the classical Gauss-Legendre limit.

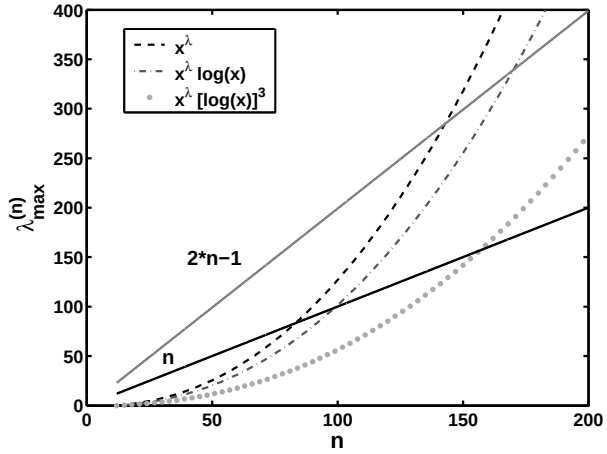

Figure 9. The behavior of $\lambda_{\max}^{(n)}$ for finite energy Müntz polynomials when the target precision is set to be double. The solid lines are the functions $f(n) = 2n-1$ and $f(n) = n$ useful to show the performance of the quadrature.

Müntz polynomials with rational exponents

These polynomials are constituted by monomials x^{λ} whose exponents λ are rational numbers. This kind of polynomials are well known in literature and Gauss-Legendre quadrature combined with monomial transformation of integer order r has been applied. For the sake of completeness we include a brief description on how to deal with this class of polynomials.

Without loss of generality, all the exponents of the monomials associated to Müntz polynomials with rational exponents are with the same denominator q:

$$\lambda_{k,m} = \frac{m}{q} + k, \quad q \in \mathbb{N}_0, \quad \forall m \in \mathbb{Z} \bigcap \{q > |m|\}, \quad \forall k \in \mathbb{N} \bigcap \left\{ \frac{m}{q} + k > -1 \right\} \tag{96}$$

By using the monomial transformation $x = s^q$ $(r = q)$ we obtain:

$$\int_0^1 x^{\frac{m}{q}+k} dx = \int_0^1 s^{m+q\,k} q\, s^{q-1} ds = \int_0^1 s^{\gamma} ds \tag{97}$$

where the exponents $\gamma = m + qk + q - 1$ are integer numbers.

Let us define

$$[m_{\min}, k_{\min}] \quad = \quad \{[m,k] \ : \ \gamma = \gamma_{\min}\} \tag{98}$$

$$[m_{\max}, k_{\max}] \quad = \quad \{[m,k] \ : \ \gamma = \gamma_{\max}\} \tag{99}$$

where typically in practical applications with singular integrands $k_{\min} = 0$ and $m_{\min} < 0$.

The constrain, that defines the optimal quadrature using Gauss-Legendre rules and the monomial transformation $x = s^q$, is:

$$\gamma_{\max} = m_{\max} + qk_{\max} + q - 1 \leq 2n - 1 \tag{100}$$

From (100) we need at least $n_{\min} = (m_{\max} + q + q\,k_{\max})/2$ points of Gauss-Legendre quadrature to exactly integrate a given Müntz polynomial with rational exponents $\lambda_{k,m}$ with $m_{\min}/q \leq \lambda_{k,m} \leq m_{\max}/q + k_{\max}$.

REFERENCES

1. Borwein P, Erdélyi T. *Polynomials and Polynomial Inequalities, Graduate Texts in Math. 161.* Springer-Verlag, 1995.
2. Erdélyi T, Johnson WB. The "Full Müntz Theorem" in $L_p[0,1]$ for $0 < p < \infty$. *Journal d'Analyse Math.* 2001; **84**:145–172.
3. Stieltjes TJ. Sur une généralisation de la théorie des quadratures mécaniques. *C.R. Acad. Sci. Paris* 1884; **99**:850-851.
4. Gautschi W. Algorithm 331: Gaussian quadrature formulas. *Communications of ACM* 1968; **11**(6):432–436.
5. Gautschi W. On generating orthogonal polynomials. *SIAM J. SCL. Stat. Comput.* 1982; **3**(3):289–317.
6. Gautschi W. Algorithm 726: ORTHPOL-A package of routines for generating orthogonal polynomials and Gauss-type quadrature rules. *ACM Transactions on Mathematical Software* 1994; **20**(1):21–62.
7. Ma J, Rokhlin V, Wandzura S. Generalized Gaussian quadrature rules for systems of arbitrary functions. *SIAM Journal on Numerical Analysis* 1996; **33**(3):971–996.
8. Gautschi W. Algorithm 793: GQRAT-Gauss Quadrature for Rational Functions. *ACM Transactions on Mathematical Software* 1999; **25**(2):213–239.
9. Milovanović GV, Cvetković AS. Gaussian-type quadrature rules for Müntz system. *SIAM Journal on Scientific Computing* 2005; **23**(3):893-913.

10. Barrett W. Convergence Properties of Gaussian Quadrature Formulae. *The Computer Journal* 1961; **3**(4):272–277.
11. Chawla MM, Jain MK. Error Estimates for Gauss Quadrature Formulas for Analytic Functions. *Mathematics of Computation* 1968; **22**(101): 82-90.
12. Donaldson JD, Elliott D. A Unified Approach to Quadrature Rules with Asymptotic Estimates of Their Remainders. *SIAM Journal on Numerical Analysis* 1972; **9**(4):573–602.
13. Lether FG. Error Bounds for Gaussian Quadrature of Analytic Functions. *Applied Mathematics and Computation* 1980; **7**(3):237–246 .
14. Gautschi W, Varga RS. Error Bounds for Gaussian Quadrature of Analytic Functions. *SIAM Journal on Numerical Analysis* 1983; **20**(6):1170–1186.
15. Hunter DB. Some error expansions for Gaussian quadrature. *BIT Numerical Mathematics* 1995; **35**(1):64–82.
16. Johnston PR, Elliott D. Error estimation of quadrature rules for evaluating singular integrals in boundary element problems. *International Journal for Numerical Methods in Engineering* 2000; **48**(7):949–962.
17. Elliott D. Uniform Asymptotic Expansions of the Jacobi Polynomials and an Associated Function. *Mathematics of Computation* 1971; **25**(114):309–315.
18. Johnston PR, Elliott D. Transformations for evaluating singular boundary element integrals. *Journal of Computational and Applied Mathematics* 2002; **146**(2):231–251.
19. Elliott D. Sigmoidal transformations and the trapezoidal rule. *Journal of the Australian Mathematical Society Series B(E)* 1998; **40**:E77–E137.
20. Telles JCF. A self-adaptive co-ordinate transformation for efficient numerical evaluation of general boundary element integrals. *International Journal for Numerical Methods in Engineering* 1987; **24**: 959–973.
21. Sato M, Yoshioka S, Tsukui K. Accurate numerical integration of singular kernels in the two-dimensional boundary element method. In *Boundary Elements X*, C.A.Brebbia (Ed.), Springer: Berlin, 1988; Vol.1: 279-298.
22. Alarćon E, Doblaré M, Sanz-Serna. An efficient nonlinear transformation for the numerical integration f the singular integrals apperaing in the 2-D boundary element method. In *Boundary Element Methods in Engineering* Annigeri BS, Tseng K (eds), 1989; 472–279.
23. Cerrolaza M, Alarćon E. A bi-cubic transformation for the numerical evaluation of the Cauchy principal value integrals in boundary methods. *International Journal for Numerical Methods in Engineering* 1989; **28**: 987–999.
24. Evans G. *Practical numerical integration.* Wiley: Chichester, 1993.
25. Monegato G, Sloan IH. Numerical solution of the generalised airfoil equation for an airfoil with a flap. *SIAM J. Numer. Anal.* 1997; **34**, 6: 2288-2305.
26. Monegato G, Scuderi L. High order methods for weakly singular integral equations with non smooth input functions. *Math. Comput.* 1998; **67**: 1493–1515.
27. Singh KM, Tanaka M. On non-linear transforations for accurate numerical evaluation of weakly singular boundary integrals. *International Journal for Numerical Methods in Engineering* 2001; **50**: 2007–2030.
28. Monegato G, Scuderi L. Numerical solution of functions with endpoint singularities and/or complex poles in 3D Galerkin boundary element methods. *Publ. RIMS*, Kyoto Univ. 2005; **41**: 869–895.
29. Graglia R, Lombardi G. Singular higher order complete vector bases for finite methods. *IEEE Transactions on Antennas and Propagation*, 2004; **52**, n. 7: 1672–1685 .
30. IEEE standard for binary floating-point arithmetic. *ANSI/IEEE Std 754-1985*, 12 Aug 1985.
31. Gradshteyn IS, Ryzhik IM. *Tables of Integrals, Series and Products (5th edn).* Academic Press: Boston, 1994.
32. Abramowitz M, Stegun IA. *Handbook of mathematical functions : with formulas, graph, and mathematical tables (10th edn).* Dover: New York, 1972.