High Order Approximation for Müntz and Müntz-Logarithmic Polynomials Using Emprical Interpolation Method

Tengteng Cui

with

Mejdi Azaiez and Chuanju Xu

厦門大學 XIAMEN UNIVERSITY

BORDEAUX INP

Outline

Motivation

- The need for a quadrature formula adapted to PDEs with Nonlocal and Singular Operators
- Hight order numerical integration when Müntz and Müntz-Logarithmic Polynomials are used
- Solve PDE in complex domains using monodomaine approaches
- etc....

Müntz legendre polynomial interpolation

We adopt the following definition for x^λ given by:

$$x^\lambda = e^{\lambda \log x}, \quad x \in (0, \infty), \lambda \in \mathbb{C}, \tag{1}$$

Given a complex sequence $\Lambda = \{\lambda_0, \lambda_1, \lambda_2, \cdots\}$, a linear combination system $\{x^{\lambda_0}, x^{\lambda_1}, \cdots, x^{\lambda_n}\}$ is called a Müntz polynomial, or a $\Lambda-$polynomials.
In the sequel, we consider

$$\Lambda = \{\lambda_0, \lambda_1, \lambda_2, \cdots\}, \quad R(\lambda_k) > -\frac{1}{2}, \tag{2}$$

where $R(\lambda)$ is the real part of λ. This ensures that every Λ-polynomial is dense in $L^2[0, 1]$.

Then, we give the definition of Müntz Legendre polynomial (see [**Taslakyan**])

Definition

Let $\Lambda = \{\lambda_0, \lambda_1, \lambda_2, \cdots\}$ be a complex sequence, $R(\lambda_k) > -\frac{1}{2}$. We define the nth Müntz-Legendre polynomial on $(0, 1]$ to be

$$L_n(\lambda_0, \cdots, \lambda_n; x) = \frac{1}{2\pi i} \int_\Gamma \prod_{k=0}^{n-1} \frac{t + \bar{\lambda}_k + 1}{t - \lambda_k} \frac{x^t}{t - \lambda_n} \, dt, \quad n = 0, 1, \cdots, \tag{3}$$

where the simple contour Γ surrounds all the zeros of the denominator in the integrand, and $\bar{\lambda}$ denotes the conjugate of λ.

[1] AK Taslakyan. Some properties of legendre quasi-polynomials with respect to a Müntz system. *Mathematics*, 2:179–189, 1984.

Now we consider the important special case where

$$\lambda_{2k} = \lambda_{2k+1} = k, \quad (k = 0, 1, \cdots). \tag{4}$$

Cauchy residue theorem applied to the integral in (3), conduct to the representation for the corresponding Müntz polynomials:

$$L_n(x) = R_n(x) + S_n(x) \log x \quad (n = 0, 1, \cdots), \tag{5}$$

where $R_n(x)$ and $S_n(x)$ are algebraic polynomials of degree $[n/2]$ and $[(n-1)/2]$, respectively, i.e.,

$$R_n(x) = \sum_{\nu=0}^{[n/2]} a_\nu^{(n)} x^\nu, \quad S_n(x) = \sum_{\nu=0}^{[(n-1)/2]} b_\nu^{(n)} x^\nu. \tag{6}$$

Notice that $L_n(1) = R_n(1) = 1$.

The first few Müntz polynomials (3) are:

$$\begin{aligned} L_0(x) &= 1, \\ L_1(x) &= 1 + \log x, \\ L_2(x) &= -3 + 4x - \log x, \\ L_3(x) &= 9 - 8x + 2(1+6x)\log x, \\ L_4(x) &= -11 - 24x + 36x^2 - 2(1+18x)\log x. \end{aligned}$$

We plot the first four Müntz Legendre polynomials :

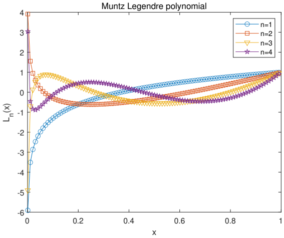


Figure 1: Müntz Legendre polynomials with $n = 1, 2, 3, 4$

Empirical Integration Method (EIM)

Let $\mathcal{G}(\cdot;\mu)$ be a parametrized function that generate(span) the full function space by choosing parameters μ from the parameter domain $\mathcal{D}$

- $\mathcal{U} = span\{\mathcal{G}(\cdot;\mu) : \mu \in \mathcal{D}\}$. It is called the training set
- $W_N = span\{\mathcal{G}(\cdot;\mu) : \mu \in \Xi\}$. $\Xi \subseteq \mathcal{D}$ is of dimension $\mathcal{N}$. It is called the basis set
- $W_N \subseteq \mathcal{U}$

Algorithm 1 Greedy EIM

$\mu_1 = \arg\max_{\mu \in \mathcal{D}} \|\mathcal{G}(\cdot;\mu)\|_{L^\infty(\Omega)}$
$x_1 = \arg\max_{x \in \Omega} |\mathcal{G}(x;\mu_1)|$
$q_1 = \mathcal{G}(\cdot;\mu_1)/\mathcal{G}(x_1;\mu_1)$
for $m = 2 : N$ do
$\mu_m = \arg\max_{\mu \in \mathcal{D}} \|\mathcal{G}(\cdot;\mu) - \mathcal{I}_{m-1}[\mathcal{G}(\cdot;\mu)]\|_{L^\infty(\Omega)}$
$x_m = \arg\max_{x \in \Omega} |\mathcal{G}(x;\mu_m) - \mathcal{I}_{m-1}[\mathcal{G}(\cdot;\mu_m)](x)|$
$q_m = \frac{\mathcal{G}(\cdot;\mu_m) - \mathcal{I}_{m-1}[\mathcal{G}(\cdot;\mu_m)]}{\mathcal{G}(x_m;\mu_m) - \mathcal{I}_{m-1}[\mathcal{G}(x_m;\mu_m)](x_m)}$
end for

Objective: approximate a function f over a domain Ω by a linear combination of N pre-defined basis functions

$$f(x) \approx \mathcal{I}_N[f](x) = \sum_{i=1}^{N} \beta_i q_i(x). \tag{7}$$

Müntz Legendre polynomials interpolation

$\mathcal{U} = span\{1, L_1(x), \cdots, L_N(x)\}$, where $\lambda_n = n\lambda + q$.
Distribution of the magic points $\{x(j)\}_{j=0}^{N}$ with various N using EIM based on Müntz Legendre polynomial:

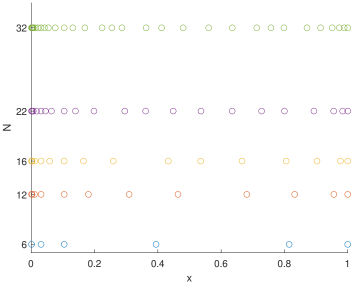


Figure 2: Interpolation nodes distribution of Müntz-legendre polynomial with different N

We plot the error curves for the Müntz-Legendre polynomials approximation to $f(x) = x^{1/3}$ by using EIM interpolation and classical GLL interpolation.

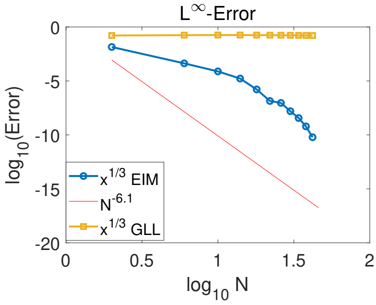


Figure 3: $f(x) = x^{1/3}$

Müntz polynomials interpolation

$\mathcal{U} = span\{1, x^{\lambda_1}, \cdots, x^{\lambda_N}\}$, where $\lambda_n = n\lambda + q$. Fig.4 exhibits the influence of the parameters λ, q on the nodes distribution.

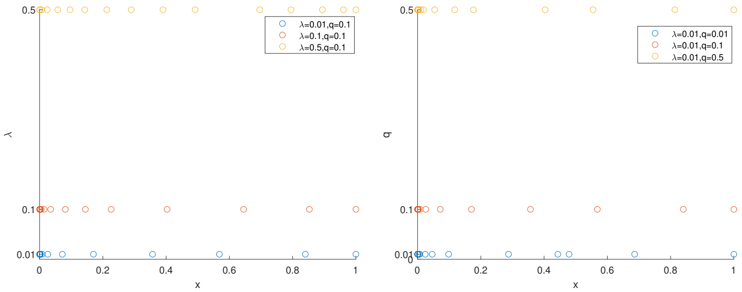


Figure 4: Interpolation nodes distribution of Müntz polynomial $N = 16$ with different λ, q

We plot the error curves for the fractional polynomial approximation to $f(x) = x^{1/9}$ with various λ, q in Fig.5.

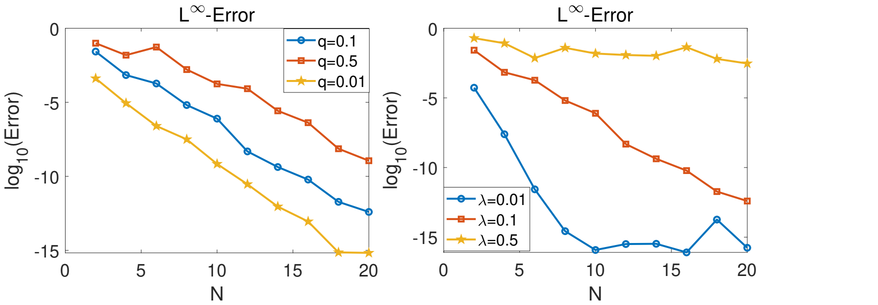


Figure 5: $\lambda = 0.1$(left) and $q = 0.1$(right)

Müntz-logarithmic polynomials interpolation

$$\mathcal{U} = span\{1, x^{\lambda_1}\log(x), \cdots, x^{\lambda_N}\log(x)^N\}, \text{ where } \lambda_n = n\lambda + q.$$

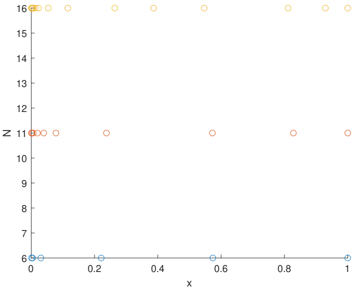


Figure 6: Interpolation nodes distribution of Müntz -logarithmic polynomial with different N

We plot the error curves for Müntz-logarithmic polynomial approximation to singular function $f(x) = x^{1/8}$ and $f(x) = x^{1/8} log(x)$ in the left of Fig.7. Then we plot more singular functions $f(x) = x^{-1/6}$ and $f(x) = x^{-1/10}$ in the right of Fig.7.

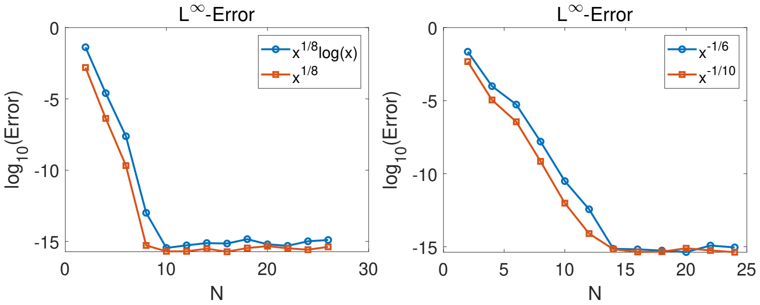


Figure 7: $\lambda = 0.01, q = 0$

Influence of sample points : Left we plot the error curves to approximate the smooth function $f(x) = exp(x)$. Right for the singular function $f(x) = x^{1/10}$

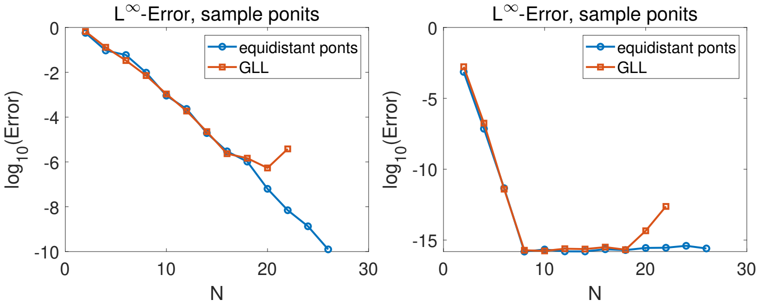


Figure 8: $\lambda = 0.01, q = 0.1$

The approximate results based on EIM on 2D domain

First, we plot the distribution of the first 25 spatial magic points on a triangular domain in the left of Fig.9, and the 25 magic points on a square domain in the right of Fig.9 about the Müntz polynomial.

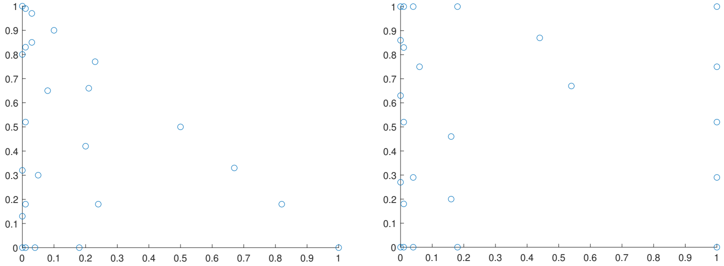


Figure 9: distribution of magic points about Müntz polynomial

Similarly, we can also get the distribution of 25 magic points of Müntz-logarithmic polynomial on the triangular domain and square domain respectively in Fig 10.

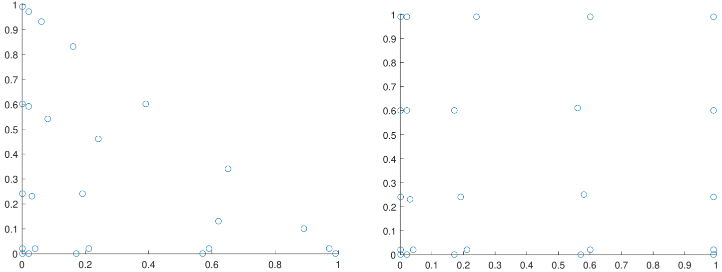


Figure 10: distribution of magic points about Müntz-logarithmic polynomial

Generalized gaussian quadrature rules

Now we aim to construct Lagrangian interpolation operator $\mathcal{I}_M$ in X_N over the set of points $T_N = \{x_i, 1 \le i \le N\}$.
Then using it as

$$\mathcal{I}_N[u(x)] = \sum_{i=1}^{N} u(x_i) h_i^N(x), \quad where \quad h_i^N(x_j) = \delta_{ij}$$

So we can construct Gaussian quadrature rule on the form

$$\int_0^1 h_k(x)dx = \sum_{i=1}^{N} h_k(x_i) w_i, \tag{8}$$

where x_i are the magic points, and $w_i = \int_0^1 h_i(x)dx$.

The starting point is that:$span\{h_i\} = span\{\mathcal{G}(\cdot;\mu_i)\} = span\{q_i\}$. We can therefore build a system of linear equations where the weights w_i are the unknowns and the basis functions h_i are used implicitly

$$\int_0^1 \mathcal{G}(\cdot;\mu_i)dx = \int_0^1 (\sum_{j=1}^{N} \mathcal{G}(x_j;\mu_i)h_j(x))dx \quad i = 1, \cdots N$$
$$= \sum_{j=1}^{N} \mathcal{G}(x_j;\mu_i) \int_0^1 h_j(x)dx, \quad j = 1, \cdots N$$

which written out becomes

$$\begin{bmatrix} \mathcal{G}(x_1;\mu_1) & \cdots & \mathcal{G}(x_N;\mu_1) \\ \vdots & \ddots & \vdots \\ \mathcal{G}(x_1;\mu_N) & \cdots & \mathcal{G}(x_N;\mu_N) \end{bmatrix} \begin{bmatrix} \int_0^1 h_1(x)dx \\ \vdots \\ \int_0^1 h_N(x)dx \end{bmatrix} = \begin{bmatrix} \int_0^1 \mathcal{G}(\cdot;\mu_1)dx \\ \vdots \\ \int_0^1 \mathcal{G}(\cdot;\mu_N)dx \end{bmatrix}$$

Example of Gaussian quadrature points and weights with respect to the system of functions $\{x^{\lambda_k}, x^{\lambda_k}\log(x)\}$ on $(0, 1)$ are given in Table 1 when $\lambda_k = \lambda k + q$,$\lambda = 0.1$, $q = 0$, and tested on selected functions in Figure.11.

Table 1: Gaussian quadrature points and weights

N	Nodes x_i	Weights w_i
5	0.0010	0.0011746602382842875258656930694295
	0.0747	0.1347439896388434977092799796200 2
	1.0000	0.2484389746777320817524943172422 6
	0.0070	0.0229476873418622993001993126188 4
	0.4100	0.5926946881032778337121606974494 5

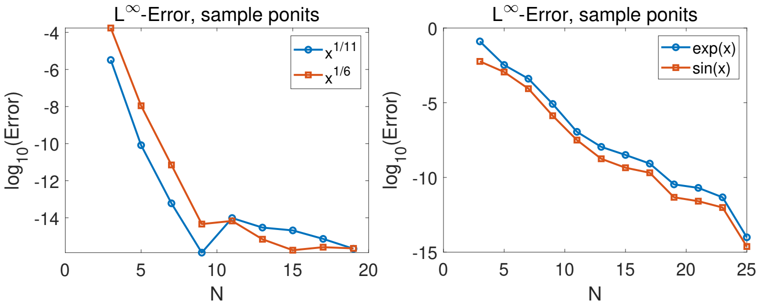


Figure 11: approximate results of numerical quadrature

Example of Gaussian quadrature points and weights with respect to the system of functions $\{1, x^{\lambda_1}, \cdots, x^{\lambda_N}\}$ on $(0, 1)$ are given in Table 2 when $\lambda_k = \lambda k + q$,$\lambda = 0.1$, $q = 0.1$.

Table 2: Gaussian quadrature points and weights

N	Nodes x_i	Weights w_i
6	1	0.24732489779607142863735567346317
	0	0.0015809301585368928180284784028382
	0.6042	0.11657754219045053408602708239755
	0.0007	0.04713139171177925752564795832732
	0.3994	0.61526451692235644479684401455218
	0.0001	-0.027879278779194557863903207143061

In the left of the figure we give the approximate results about the integral of singular functions when $\lambda = 0.1, q = 0.01$. Then we plot the error curves about the integral of smooth functions in the right of Fig.12 when $\lambda = 0.5, q = 0.5$.

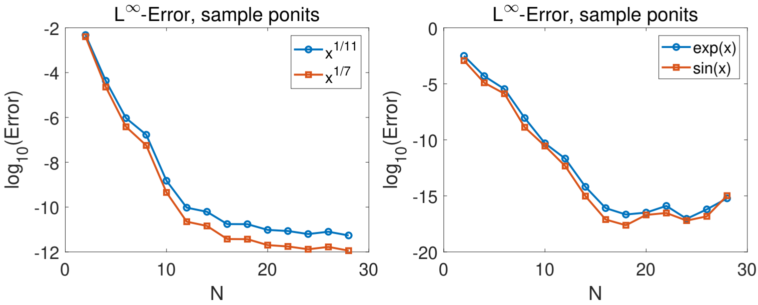


Figure 12: approximate results of numerical quadrature

Gaussian quadrature points and weights with respect to the system of functions $\{x^{\lambda_k}\log(x)^k\}$ on $(0,1)$ are given in Table 3 when $\lambda_k = \lambda k + q$,$\lambda = 0.01$, $q = 0.1$.

Table 3: Gaussian quadrature points and weights

N	Nodes x_i	Weights w_i
5	0.0010	0.0013744210088252069431219091042599
	0.0080	0.02517745597339138253722281518308
	0.0816	0.14519450449192957813194983190346
	1.0000	0.23997786024373369479144027575075
	0.4238	0.58827412192226618895774325859437

We compare the approximate results of present Gaussian quadrature rules with the generalized Gaussian quadrature in Table 3 of [**Rokhlin**] to the singular function.

Table 4: the compared absolute error

absolute error of $\int_0^1 x^{1/6}dx$			
N	5	10	15
their method	5e-5	2e-6	4e-7
our method	8e-11	3e-17	4e-13

The result show that our method is more efficient.

[1] J Ma, V Rokhlin, and Stephen Wandzura. Generalized gaussian quadrature rules for systems of arbitrary functions. SIAM Journal on Numerical Analysis, 33(3):971–996, 1996.

We fix $\lambda = 0.01, q = 0.$

Left of figure gives approximate results about the integral of singular functions. Right of the figure concerns the integral of smooth functions

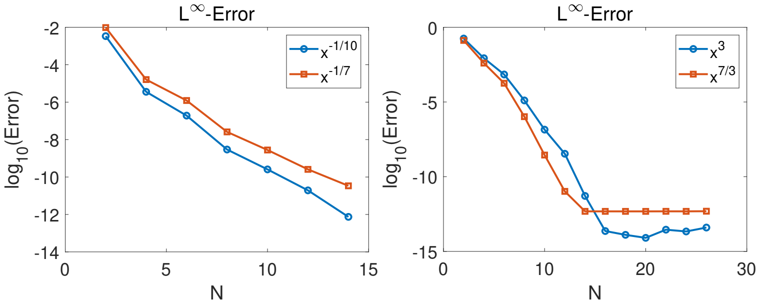


Figure 13: approximate results of numerical quadrature

Application to fractional differential equations

Given $f \in L^2(0,1)$, we consider the following Caputo fractional differential diffusion equation of order $s \in (0,1)$.

$$\begin{cases} {}_0^C D_t^s u(t) + q(t)u(t) = f(t), \quad t \in (0,1) \\ u(0) = u_0. \end{cases} \tag{9}$$

Setting $u = v + u_0$ into the above equation, we solve

$$\begin{cases} {}_0^C D_t^s v(t) + q(t)v(t) = f(t) - u_0 q(t), \quad t \in (0,1) \\ v(0) = 0. \end{cases} \tag{10}$$

Let us denote W_M based on Müntz polynomial and Müntz-logarithmic polynomial respectively.
The Müntz Galerkin method for (10) is: find $u_N \in W_N$ such that

$$({}_0^C D_t^s v_N, w_N) + (q v_N, w_N) = (\tilde{f}, w_N), \quad \forall w_N \in W_N. \tag{11}$$

We consider :

$$ {}_0^C D_t^s u(t) = f(t), \quad u(0) = 0. \tag{12} $$

We choose the exact solution to be $u(t) = t^{13/4}$.

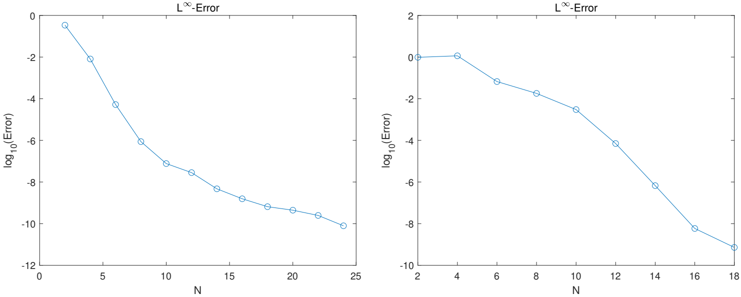
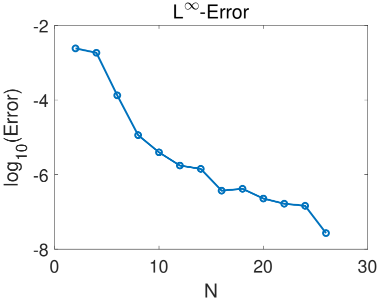


Figure 14: the convergence rate based on Müntz polynomial with $s = 0.7, \lambda = 0.7, q = 0.1$(left) and Müntz-logarithmic polynomial with $s = 0.6, \lambda = 0.1, q = 0.1$(right)

Next we consider

$$ {}_0^C D_t^s u(t) + (1 + \sin t) u(t) = \cos t, \quad u(0) = 1, \tag{13} $$

- Reference solution is the one computed with $M = 30$.
- The convergence rate based on Müntz polynomial is shown on when $s = 0.6, \lambda = 0.6, q = 0$.



Elliptic equation with geometric singularities

We consider now the problem

$$-\Delta u = f \tag{14}$$

The domain Ω is the sector shown in figure with $\Theta = \frac{3\pi}{2}$.

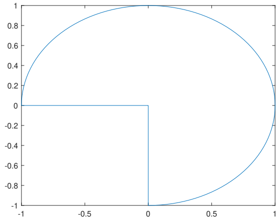


Figure 16: Dirichlet boundary condition

Using the lemma about the regularity of the solution in [**Li**]. Consider first the homogeneous boundary conditions:

$$u|_{\theta=0} = 0, \quad u|_{\theta=\frac{3\pi}{2}} = 0. \tag{15}$$

Therefore, the solution can be obtained as:

$$u = \sum_{k=0}^{\infty} a_k r^{\frac{2k}{3}} \sin(\tfrac{2k}{3}\theta). \tag{16}$$

[1] Z. C. Li and T. T. Lu. Singularities and Treatments of Elliptic Boundary Value. Mathematical and Computer Modeling 31 (2000) 97-145

- In polar coordinates, this problem reads

$$-\frac{1}{r}(ru_r)_r - \frac{1}{r^2}u_{\theta\theta} = f, \quad (r,\theta) \in (0,1) \times (0, \frac{3\pi}{2}), \qquad (17)$$
$$u|_{\partial\Omega} = 0.$$

We apply the following transform, $x = r, y = \frac{2\theta}{3\pi}$. Denote $\tilde{u}(x,y) = u(r,\theta)$, $\tilde{f}(x,y) = f(r,\theta)$.

- The problem (17) becomes

$$-(x\tilde{u}_x)_x - (\frac{2}{3\pi})^2 \frac{1}{x}\tilde{u}_{yy} = x\tilde{f}, \quad (x,y) \in (0,1) \times (0,1),$$
$$\tilde{u}(x = 0, 1, 0 \le y \le 1) = 0. \qquad (18)$$

For EIM we consider a given finite-dimensional space

$$W_N = \{(1-x)x^{\lambda_k} \otimes (1-y)y^{\lambda_j}, \lambda_k = k\lambda + q, \lambda_j = j\lambda + q\}$$
$$k = 1, 2, \cdots m, j = 1, 2, \cdots, n, N = mn.$$

Example 1 : the exact solution is $u(x,y) = (x-1)^c x^c y^d (y-1)^d$.
Table 5 is for $\lambda = 1.1, q = 0.1$ and $c = 5/4, d = 8/3$.

Table 5: the Max error on 2D square domain

N	4	9	16
L^∞-error	8e-10	1e-11	5e-13

Example 2 : the exact solution is $u(x,y) = (1-x)^c x^d \sin(\pi y)$.
Table 6 is for $\lambda = 1.1, q = 0.1$ and $c = 8/3, d = 5/4$.

Table 6: the Max error on 2D square domain

N	4	9	16
L^∞-error	2e-11	1e-10	1e-12

singular two point boundary value problem

In the last example, we use Müntz-logarithmic polynomial as basis functions to solve the following singular two point boundary value problem

$$
\begin{aligned}
&(x^\alpha u')' = f, x \in (0,1), 0 \leq \alpha < 1, \\
&u(0) = u(1) = 0.
\end{aligned} \tag{20}
$$

Let us denote

$$
X_N := span\{x^{\lambda_k}(\log(x))^k, k = 1, 2, \cdots, N\}. \tag{21}
$$

Our Müntz-logarithmic Galerkin method is: find $u_N \in X_N$ such that

$$
((x^\alpha u_N')', w_N) = (I_N f, w_N), \quad \forall w_N \in X_N. \tag{22}
$$

We now present some numerical results. We first take the exact solution to be $u(x) = x^{11/3}(1-x)$ and $u(x) = x^{7/3}(1-x)$ The convergence rate is shown on the Fig.17. We observe that the error converges exponentially in both cases despite the fact that the solutions are weakly singular near $x = 0$.

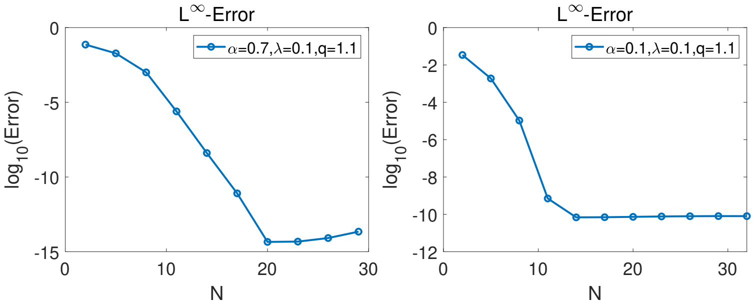


Figure 17: $u(x) = x^{11/3}(1-x)$(left) $u(x) = x^{7/3}(1-x)$(right)

Concluding Remarks

- We first presented EIM and some of its properties
- We applied EIM based on Müntz polynomial and Müntz-logarithmic polynomial to approximate singular function.
- We derived a generalized Gauss quadrature based on lagrange interpolation polynomial using the magic points.
- We gave some numerical evidences demonstrated the efficiency of our approaches

Further directions:

- Continue with collocation methods for the singular problem.
- Interpolation error estimates about Müntz polynomial and Müntz-logarithmic polynomial based on EIM.