11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
E. Oñate, J. Oliver and A. Huerta (Eds)

COMPARED COMPUTATIONAL PERFORMANCES OF GALERKIN APPROXIMATIONS FOR PERTURBED VARIABLE-COEFFICIENT DIFFERENTIAL EQUATIONS, ONE-DIMENSIONAL ANALYSIS

DIEGO GARIJO*, FRANCISCO J. GÓMEZ-ESCALONILLA† AND ÓSCAR F. VALENCIA‡

* Safran Engineering Services (Safran Group)
Calle Arnaldo de Vilanova, 5, 28906, Getafe, Madrid (Spain)
e-mail: diego.garijo@safran-engineering.com

†Airbus Defence & Space
Avda John Lennon s/n, 28906, Getafe, Madrid (Spain)
e-mail: Javier.Gomez-Escalonilla@military.airbus.com

‡FACC AG
Fischerstraße, 9, 4910, Ried im Innkreis (Austria)
e-mail: O.Valencia@facc.com

Key words: Galerkin, Perturbation, Moving Least Squares, Finite Element, Lagrange, Bernstein

Abstract. The computational performance of meshfree Moving Least Squares technique when solving the Galerkin weak form of one-dimensional perturbed variable-coefficient differential equations is tested against mesh-based Finite Element Method and spectral-type Lagrange and Bernstein Partitions of Unity. These approximations are subjected to numerical experimentation in classic perturbation benchmarks of literature. The behavior of these schemes when dealing with the reproduction of sharp gradients of the field variable in boundary layer regions is discussed. The accuracy and computational cost of the numerical approximations are compared, including convergence analysis under $h-$ and $p-$refinements using patterns of equi-spaced nodes in the whole domain.

1 INTRODUCTION

In recent decades, the Moving Least Squares (MLS) approximation [1] has become an efficient, well-established and widely studied technique for solving partial differential equations numerically. The MLS approximation was employed by Nayroles *et al.* [2] as the base of the Diffuse Element Method (DEM) and was later incorporated by Belytschko

et al. to the Galerkin weak form of differential equations yielding the meshless Element-Free Galerkin Method (EFGM) [3]. A complete survey of MLS applications, including parametric studies, consistency and convergence analysis with illustrative computational tests can be found in the works by Liu and Gu [4, 5].

The Finite Element Method (FEM) [6] is based on a topological map called mesh to discretize the equations in the domain of analysis. The $p-$refinement is generally limited to Lagrange polynomials of first or second order in most of commercial software packages. In this work, FEM and global high-order Lagrange polynomials are experimented.

In recent works, Bernstein polynomials [7] have been tested with success as trial functions for boundary value problems. Bernstein polynomials constitute a Partition of Unity (PU) by themselves and convergence is achieved by means of $p-$refinement from patterns of equi-spaced evaluation points, instead of $h-$refinement, which is the classic strategy of mesh-based approximations to reproduce local high-order solutions, e.g. confined boundary layers in convection-diffusion problems.

In this context, this work discusses and compares the computational performances of the four mentioned approximations when dealing with one-dimensional variable-coefficient differential equations with small perturbing parameters, whose solutions may exhibit boundary layer structure. The Galerkin weak form of these equations is developed and discretized. Numerical experiments test and compare the accuracy and convergence of the numerical solutions.

2 NUMERICAL APPROXIMATIONS AND DERIVATIVES

The formulation for the generation of the shape functions for the four approximation techniques (MLS, FEM, Lagrange PU and Bernstein PU) compared in this work is explicitly included in this section. These techniques will be used to approximate the field variable in the Galerkin variational form developed in section 3. Since the generic perturbation problem considered in section 3 is governed by the second-order differential equation (10), only the first derivatives of shape functions are computed.

2.1 Moving Least Squares

The MLS approximation $u^h(x)$ to a function $u(x)$ defined in a one-dimensional domain Ω was given by Lancaster and Salkauskas [1]:

$$u^h(x) = \sum_{i=1}^{n} \phi_i(x)\, u_i = \sum_{i=1}^{n} \sum_{j=1}^{m} p_j(x) \left(\mathbf{A}^{-1}(x)\, \mathbf{C}(x)\right)_{ji} u_i \tag{1}$$

where $\phi_i(x)$ are the MLS shape functions, $p_j(x)$ are the functions of the intrinsic basis, typically monomials —for instance, $\mathbf{p}^T = \{1\ x\}$ (linear basis), $\mathbf{p}^T = \{1\ x\ x^2\}$ (quadratic basis)—, m is the number of terms of such basis and the matrices $\mathbf{A}(x)$ and $\mathbf{C}(x)$ are

expressed as:

$$\mathbf{A}(x) = \sum_{i=1}^{n} w(x - x_i)\,\mathbf{p}(x_i)\,\mathbf{p}^T(x_i) \tag{2}$$

$$\mathbf{C}(x) = [w(x - x_1)\,\mathbf{p}(x_1) \quad w(x - x_2)\,\mathbf{p}(x_2) \quad \dots \quad w(x - x_n)\,\mathbf{p}(x_n)] \tag{3}$$

being $w(x - x_i)$ a weight function similar to the SPH kernels. In order to avoid the singularity of matrix $\mathbf{A}(x)$, the selectable parameters n and m of the method, which indeed drive, respectively, the $h-$ and $p-$refinements of the approximation, should satisfy the condition $n >> m$. The size of the support domain is tuned by parameter d_{max}, as exposed in Belytschko *et al.* [8]. Since the MLS is a non-interpolating technique, the shape functions do not verify the Kronecker delta property, therefore $u_i \neq u^h(x_i)$. As stated by Babuška and Melenk (see Belytschko *et al.* [8]), the MLS shape functions constitute a Partition of Unity.

An important feature of the MLS shape functions is their numerical, not analytical, nature. The computation of the spatial derivatives of the MLS shape functions is therefore non-straightforward. The introduction of an intermediate vector $\boldsymbol{\gamma}(x)$ is often used for the calculations (Dolbow and Belytschko [9]): if the array of MLS shape functios $\boldsymbol{\Phi}(x)$ is $\boldsymbol{\Phi}(x) = \boldsymbol{\gamma}^T(x)\,\mathbf{C}(x)$, where $\boldsymbol{\gamma}(x)$ is obtained from $\mathbf{A}(x)\,\boldsymbol{\gamma}(x) = \mathbf{p}(x)$, then the spatial derivatives can be computed using $\mathbf{A}\boldsymbol{\gamma}_{,x} = \mathbf{p}_{,x} - \mathbf{A}_{,x}\boldsymbol{\gamma}$, yielding $\boldsymbol{\Phi}_{,x} = \boldsymbol{\gamma}_{,x}^T\mathbf{C} + \boldsymbol{\gamma}^T\mathbf{C}_{,x}$.

2.2 Lagrange Polynomials and the Finite Element Method

Given a set of non-coincident points $x_0, x_1, \dots, x_s$, the family of global polynomial functions of order s $(i = 0, 1, \dots, s)$:

$$l_s^i(x) = \prod_{\substack{0 \le k \le s \\ k \neq i}} \frac{(x - x_k)}{(x_i - x_k)} = \frac{(x - x_0)}{(x_i - x_0)} \cdot \dots \cdot \frac{(x - x_{i-1})}{(x_i - x_{i-1})} \cdot \frac{(x - x_{i+1})}{(x_i - x_{i+1})} \cdot \dots \cdot \frac{(x - x_s)}{(x_i - x_s)} \tag{4}$$

is the Lagrange polynomial basis. The Lagrange interpolating functions (4) possess the Kronecker delta property for the nodes x_i, $i = 0, 1, \dots, s$ and, moreover, they constitute a basis of the space Π_s of polynomials of order $g \le s$ and satisfy the Partition of Unity property for any nodal distribution. The spatial derivatives of the Lagrange functions can be obtained anaytically by simply polynomial derivation in equation (4). The Lagrange shape function associated to node k can be defined directly as:

$$\phi_k^L(x) = l_{n-1}^{k-1}(x) \quad , \quad k = 1, \dots, n \tag{5}$$

where n is the number of nodes $x_1, x_2, \dots, x_n$. Shape functions (5) extended globally to the domain of analysis Ω imply a pure $p-$refinement strategy to solve the discretized differential equations of a Galerkin scheme. This may produce loss of numerical efficiency for high orders of approximation, detrimental saturation due to Runge phenomenon for equi-spaced nodes and a remarkable increase of the computational cost due to the resulting full matrices instead of the classic band matrices of FEM and MLS approximations.

In the FEM, the numerical approximation $u^h(x)$ is defined in terms of Lagrange interpolants $N_j^e(x)$:

$$u^h(x) = \sum_{j=1}^{n_e} N_j^e(x)\, u_j^e \tag{6}$$

where the sum is extended to the n_e nodes defined by the connectivity of the element e where the point x lays. The shape functions $N_j^e(x)$ are typically linear or quadratic Lagrange basis functions, therefore the numerical approximation is driven by the mesh refinement. The Finite Element approximation satisfies the Kronecker delta property, being u_j^e in (6) the actual nodal values of the field variable $u(x)$. Besides, Lagrange interpolants $N_j^e(x)$ conform a PU. For the particular case of linear interpolats, as it is implemented in the numerical experiments presented in section 4, the derivative of FEM shape function associated to node k is:

$$\frac{dN_k}{dx} = \begin{cases} 1/(x_k - x_{k-1}), & x < x_k \\ -1/(x_{k+1} - x_k), & x > x_k \end{cases} \quad , \quad k = 1, 2, \ldots, n \tag{7}$$

with the classic lack of compatibility between adjacent elements. In the one-dimensional case, if the number of elements in the domain is set to 1 and high-order Lagrange interpolants are selected as shape functions, the FEM approximation is equivalent to a Galerkin scheme with the shape functions (4) to aproximate the field variable.

2.3 Bernstein Polynomials

Bernstein basis [7] can also be used to generate a set of shape functions usable in a Galerkin implementation [10, 11]:

$$u^h(x) = \sum_{i=1}^{n} B_{i-1}^{n-1}(x)\, a_i = \sum_{i=1}^{n} \binom{n-1}{i-1} x^{i-1}(1-x)^{n-i} a_i \tag{8}$$

where $B_{i-1}^{n-1}(x) \equiv B_i$ are the polynomial functions of Bernstein basis, n is the number of evaluation points in the domain and a_i is the i-th evaluation point parameter, which is not the actual field variable value at evaluation point i, since Bernstein expansion does not possess the Kronecker delta property, $a_i \neq u^h(x_i)$, as it is not an interpolation scheme. Kronecker delta property is satisfied only at the extremes of the interval, $B_j(x_1) = \delta_{j1}$, $B_j(x_n) = \delta_{jn}$, being $x_1 = 0$ and $x_n = 1$. Bernstein basis constitutes a Partition of Unity of order 1 for patterns of equi-spaced evaluation points [10] and is able to reproduce any continuous and bounded function $f(x)$ in the interval [0,1] by uniform convergence [7].

The derivatives of Bernstein shape functions can be expressed as [11]:

$$\frac{dB_i}{dx} = \frac{i - 1 - (n-1)\, x}{x\,(1-x)} B_i \quad , \quad x \neq 0, 1 \tag{9}$$

$$\frac{dB_i}{dx} = \begin{cases} \mp (n-1) & \text{if } i-1 = 0, 1; \\ 0 & \text{if } i-1 \neq 0, 1. \end{cases} \quad , \quad x = 0$$

$$\frac{dB_i}{dx} = \begin{cases} \mp (n-1) & \text{if } i-1 = n-2, n-1; \\ 0 & \text{if } i-1 \neq n-2, n-1. \end{cases} \quad , \quad x = 1$$

Again, the increase of computational cost due to the resulting full matrices (instead of the band matrices of FEM and MLS) is measured in numerical experiments presented in section 4.

3 GALERKIN WEAK FORM FOR 1D PERTURBATION PROBLEMS

The boundary value problem considered for analysis is:

$$\mathcal{L}_\varepsilon u \equiv a(x;\varepsilon)\frac{d^2u}{dx^2} + b(x;\varepsilon)\frac{du}{dx} + c(x;\varepsilon)\,u = f(x) \quad , \quad 0 < \varepsilon << 1 \tag{10}$$

$$u(0) = \bar{u}_0 \quad , \quad u(1) = \bar{u}_1 \tag{11}$$

The governing differential equation (10) is the general form of a non-homogeneous variable-coefficient second-order differential equation with small perturbing parameter ε. Typically, when the highest derivative is multiplied by the perturbing parameter, $a(x;\varepsilon) = \varepsilon$, the solution to (10) presents narrow regions of very sharp gradients to adapt to the imposed boundary conditions (11), this is, exhibits boundary layer structure [12]. The problem (10)-(11) is a classic mathematical model of one-dimensional Perturbation Theory, which has been widely studied with numerical approximations, including Finite Difference, Finite Element and meshless techniques, see Morton and Mayers [13] or Miller *et al.* [14]. The global weak form of equation (10) is posed by introducing a virtual field δU and enforcing the integral of $\delta U \cdot (\mathcal{L}_\varepsilon u - f(x))$ to be equal to zero in the whole domain $[0, 1]$.

Essential boundary conditions (11) are enforced by means of Lagrange Multipliers, which is a classic strategy for approximations that do not verify the Kronecker delta property [4]. The additional set of unknowns to be introduced into the weak form are $\boldsymbol{\lambda}^T = \{\lambda_0 \;\; \lambda_1\}$ and $\boldsymbol{\delta\lambda}^T = \{\delta\lambda_0 \;\; \delta\lambda_1\}$.

The Galerkin scheme employs the same shape functions to approximate both the trial and the test fields:

$$u = \sum_{i=1}^{n} \psi_i(x)\,\alpha_i \quad , \quad \delta U = \sum_{j=1}^{n} \psi_j(x)\,\delta\alpha_j \tag{12}$$

where ψ_k and α_k are, respectively, generic shape functions and generic nodal parameters (actual nodal values in the case of verification of Kronecker delta property) of the selected approximation. Inserting now the expressions (12) into the weak form of equation (10) and with the definition of the arrays $\boldsymbol{\Psi}^T = \{\psi_1 \;\; \psi_2 \;\; \dots \;\; \psi_n\}$, $\boldsymbol{\alpha}^T = \{\alpha_1 \;\; \alpha_2 \;\; \dots \;\; \alpha_n\}$ and $\boldsymbol{\delta\alpha}^T = \{\delta\alpha_1 \;\; \delta\alpha_2 \;\; \dots \;\; \delta\alpha_n\}$, the weak form in compact form becomes:

$$\boldsymbol{\delta\alpha}^T a\left(1;\varepsilon\right)\boldsymbol{\Psi}_j\left(x=1\right)\lambda_1+\delta\lambda_1 a\left(1;\varepsilon\right)\boldsymbol{\Psi}_i^T\left(x=1\right)\boldsymbol{\alpha}-\delta\lambda_1 a\left(1;\varepsilon\right)\bar{u}_1- \tag{13}$$
$$-\boldsymbol{\delta\alpha}^T a\left(0;\varepsilon\right)\boldsymbol{\Psi}_j\left(x=0\right)\lambda_0-\delta\lambda_0 a\left(0;\varepsilon\right)\boldsymbol{\Psi}_i^T\left(x=0\right)\boldsymbol{\alpha}+\delta\lambda_0 a\left(0;\varepsilon\right)\bar{u}_0-$$
$$-\boldsymbol{\delta\alpha}^T\int_0^1 a\left(x;\varepsilon\right)\frac{d\boldsymbol{\Psi}_j}{dx}\frac{d\boldsymbol{\Psi}_i^T}{dx}dx\cdot\boldsymbol{\alpha}+\boldsymbol{\delta\alpha}^T\int_0^1\left(b\left(x;\varepsilon\right)-\frac{da\left(x;\varepsilon\right)}{dx}\right)\boldsymbol{\Psi}_j\frac{d\boldsymbol{\Psi}_i^T}{dx}dx\cdot\boldsymbol{\alpha}+$$
$$+\boldsymbol{\delta\alpha}^T\int_0^1 c\left(x;\varepsilon\right)\boldsymbol{\Psi}_j\boldsymbol{\Psi}_i^T dx\cdot\boldsymbol{\alpha}-\boldsymbol{\delta\alpha}^T\int_0^1 f\left(x\right)\boldsymbol{\Psi}_j dx=0$$

Solving (13) for all $\boldsymbol{\delta\alpha}$ and $\boldsymbol{\delta\lambda}$ fields, the following system of equations is derived:

$$\left[\begin{array}{cc}\mathbf{K}^\varepsilon-\mathbf{D}^\varepsilon-\mathbf{M}^\varepsilon & \mathbf{G}^\varepsilon\\ \mathbf{G}^{\varepsilon T} & \mathbf{0}\end{array}\right]\left\{\begin{array}{c}\boldsymbol{\alpha}\\ \boldsymbol{\lambda}\end{array}\right\}=\left\{\begin{array}{c}-\mathbf{f}\\ -\bar{\mathbf{u}}^\varepsilon\end{array}\right\} \tag{14}$$

where the terms in the matrices are:

$$\mathbf{K}_{ij}^\varepsilon=\int_0^1 a\left(x;\varepsilon\right)\frac{d\psi_i}{dx}\frac{d\psi_j}{dx}dx\qquad\mathbf{D}_{ij}^\varepsilon=\int_0^1 b\left(x;\varepsilon\right)\frac{d\psi_i}{dx}\psi_j dx\qquad\mathbf{f}_i=\int_0^1\psi_i f\left(x\right)dx \tag{15}$$
$$\mathbf{M}_{ij}^\varepsilon=\int_0^1 c\left(x;\varepsilon\right)\psi_i\psi_j dx\qquad\mathbf{G}_{ik}^\varepsilon=-a\left(x=x_k;\varepsilon\right)\psi_i\left(x=x_k\right)\qquad\bar{\mathbf{u}}_k^\varepsilon=a\left(x=x_k;\varepsilon\right)\bar{u}_k$$

As a general rule, the computation of $\mathbf{M}_{ij}$, $\mathbf{D}_{ij}$ and $\mathbf{K}_{ij}$ in (15) requires a numerical integration quadrature, mainly due to shape functions may not be analytical —this is the case of MLS—, and also depending on the form of the terms $a\left(x;\varepsilon\right)$, $b\left(x;\varepsilon\right)$ and $c\left(x;\varepsilon\right)$.

4 NUMERICAL EXPERIMENTS

In this section, the exposed Galerkin approximations are tested in reference cases of bibliography. The experiments deal with the response of the approximations in terms of numerical error, convergence and computational cost when the perturbing parameter ε is modified. The accuracy of the approximations is evaluated with the global norm of error E, defined as:

$$E=log\left\{\sqrt{\frac{1}{2}\int_0^1\left(u^{exact}\left(x\right)-u^{num}\left(x\right)\right)^2 dx}\right\} \tag{16}$$

where u^{exact} and u^{num} are the exact and the numerical solutions, respectively. The computational cost is measured by the elapsed time from the start of the assembly of system (14)-(15) until the retrieval of the nodal parameters $\boldsymbol{\alpha}$.

The four approximations tested share the following features in their implemantations: patterns of n equi-spaced nodes (which means $n-1$ elements in FEM and order of approximation $n-1$ for Lagrange and Bernstein PUs), numerical integration of terms (15) performed via Gauss quadrature of 11 points in each interval between nodes and enforcement of essential boundary conditions by means of Lagrange Multipliers. For MLS, support domain size parameter is swept in the range $d_{max}=2.0$ to 4.0, linear intrinsic basis is used and weight function selected is a cubic spline as per [9].

4.1 Boundary-layer analysis of a variable-coefficient differential equation

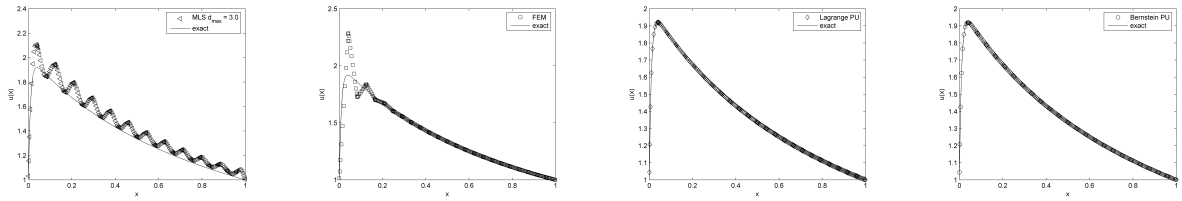
The first analysis (case A) is performed for the perturbed differential equation:

$$\varepsilon u^{''} + (1+x)\, u^{'} + u = 0 \quad , \quad 0 < \varepsilon << 1 \tag{17}$$

with boundary conditions $u\,(0) = 1, u\,(1) = 1$. The exact solution up to order ε^3 can be found in Bender and Orszag [12]:

$$\begin{aligned} u_{unif,3}\,(x) = \left(2\bar{X} - e^{-X}\right) + \varepsilon\left[2\bar{X}^3 - (1/2)\bar{X} + \left((1/2)X^2 - (3/2)\right)e^{-X}\right] + \\ +\varepsilon^2\left[6\bar{X}^5 - (1/2)\bar{X}^3 - (1/4)\bar{X} - \left((1/8)X^4 - (3/4)X^2 + (21/4)\right)e^{-X}\right] + \\ +\varepsilon^3\left[30\bar{X}^7 - (3/2)\bar{X}^5 - (1/4)\bar{X}^3 - (5/16)\bar{X} + \right. \\ \left. + \left((1/48)X^6 - (3/16)X^4 + (21/8)X^2 - (1949/72)\right)e^{-X}\right] + O\left(\varepsilon^4\right) \end{aligned} \tag{18}$$

with $\varepsilon X = x$ and $\bar{X} = (1+x)^{-1}$. Expression (18) is a uniform approximation to $u\,(x)$ over the domain $0 \leq x \leq 1$, $|u_{unif,3}\,(x) - u\,(x)| = O\,(\varepsilon^4)$. The solution presents a boundary layer in the left extreme $x = 0$ of the interval. The exact and Galerkin solutions are plotted in Figure 1 for $\varepsilon = 10^{-2}$ and $n = 25$. In MLS, parameter d_{max} is taken 2.0, 3.0 and 4.0. From Figure 1, it is highlighted the capability of Lagrange and Bernstein PUs to reproduce the sharp gradient in the boundary layer region. FEM error is mainly confined to the left extreme, while MLS approximation resembles a waveform extended globally.



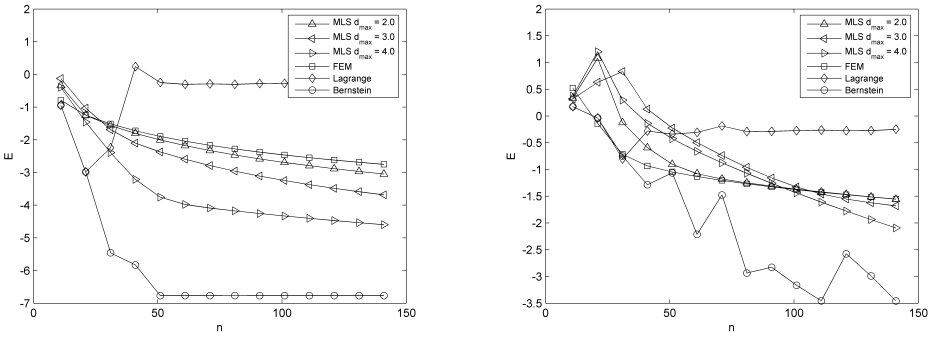
(a) (b) (c) (d)

Figure 1: Case A, $\varepsilon = 10^{-2}$, $n = 25$, (a) MLS, $d_{max} = 3.0$; (b) FEM; (c) Lagrange; (d) Bernstein

The convergence in norm of error E with n is presented in Figure 2. A diagram of error versus computational cost is provided in Figure 3. In both figures, results for perturbing parameter $\varepsilon = 10^{-2}$ and $\varepsilon = 10^{-3}$ are shown. From Figure 2a, the fastest rate of convergence, as well as the lowest error, is obtained with Bernstein PU. Above $n \sim 50$, the error becomes constant, with excellent accuracy $E \sim -7$, suggesting that the minimum numerical error has been achieved. MLS approximation reaches intermediate results between FEM and Bernstein PU, with better performances as d_{max} is increased.

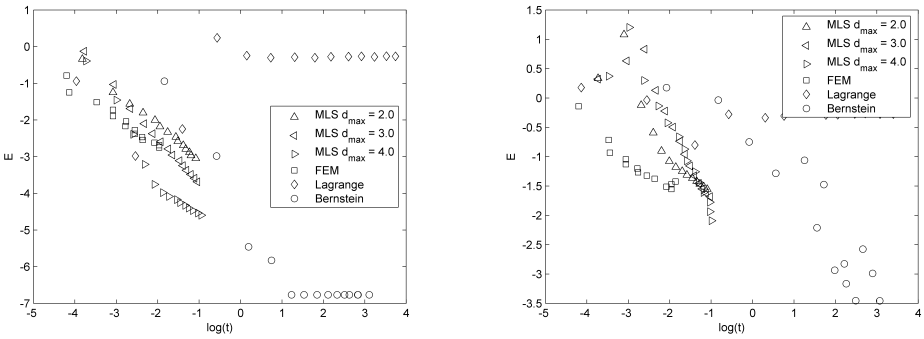
Lagrange PU crashes soon above $n \sim 30$ due to Runge phenomenon for patterns of equi-spaced nodes. Similar results are observed in Figure 2b, although for $\varepsilon = 10^{-3}$

Bernstein PU becomes slightly unstable and MLS convergence is slowed, with remarkable errors for low values of n.



(a) (b)

Figure 2: Norm of error E for case A, (a) $\varepsilon = 10^{-2}$; (b) $\varepsilon = 10^{-3}$



(a) (b)

Figure 3: Norm of error E vs computational cost for case A, (a) $\varepsilon = 10^{-2}$; (b) $\varepsilon = 10^{-3}$

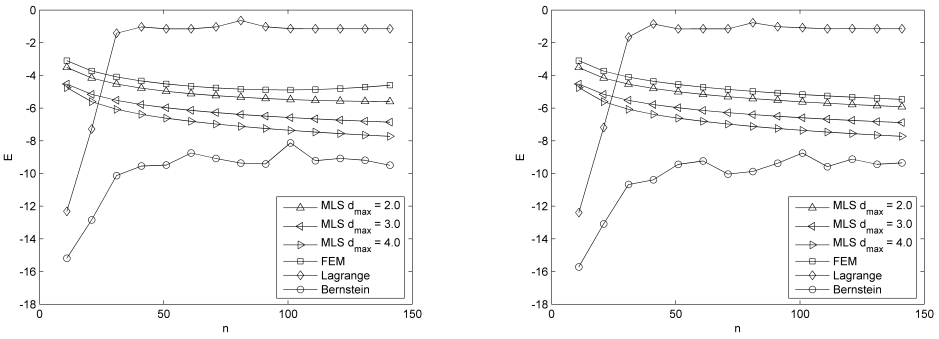
The diagrams $E - log(t)$ in Figure 3 show that the excellent accuracy of Bernstein PU is spoiled by the large computational cost associated to the generation of full matrices. FEM is, of course, the fastest approximation, while MLS performance can be tuned by means of parameter d_{max}, which determines the band width of the matrices involved.

4.2 Exactly soluble differential equation with singular coefficients

The second boundary value problem (case B) considered is:

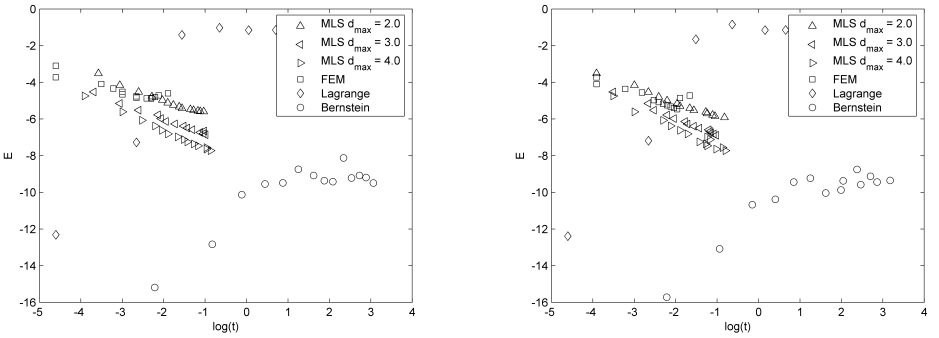
$$\varepsilon u'' + (1/x)u' + (1/x^2)u = 2/x - 2\varepsilon - 3 \quad , \quad 0 < \varepsilon << 1 \tag{19}$$

with boundary conditions $u(0) = 0, u(1) = 0$. The exact solution in this case is $u^{exact} = x - x^2$. The numerical results are shown in Figures 4 and 5:



(a) (b)

Figure 4: Norm of error E for case B, (a) $\varepsilon = 10^{-4}$; (b) $\varepsilon = 10^{-8}$



(a) (b)

Figure 5: Norm of error E vs computational cost for case B, (a) $\varepsilon = 10^{-4}$; (b) $\varepsilon = 10^{-8}$

This case has the closed polynomial solution $u^{exact} = x - x^2$, which Lagrange and Bernstein PUs are able to reproduce exactly with low orders of approximation. Afterwards, numerical dissipation (attributed in [11] to truncation errors and evaluation of powers of high order for Bernstein PU) arises when increasing the refinement, leading to a lack of convergence. Bernstein error, however, becomes stable at very accurate levels. MLS performance achieves very good error-time efficiency and parameter d_{max} can be adjusted to retrieve the desired accuracy.

4.3 Singular coefficient problem with uniform approximation to order ε

The last benchmark analysis (case C) is the perturbation problem:

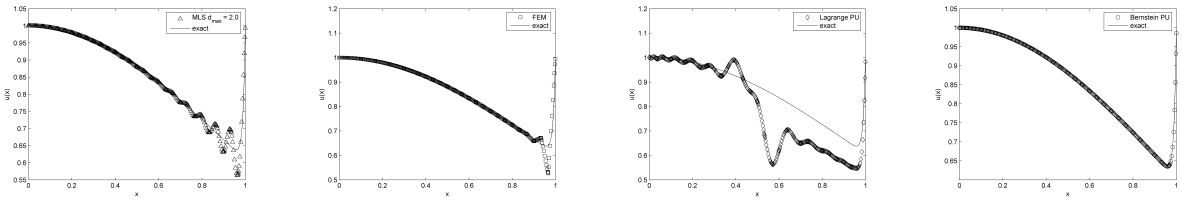
$$\varepsilon u'' - (1/x)u' - u = 0 \quad , \quad 0 < \varepsilon << 1 \tag{20}$$

with boundary conditions $u(0) = 1, u(1) = 1$. The solution presents a boundary layer of thickness ε at the right extreme $x = 1$. Bender and Orszag [12] find a uniform

approximation to $u(x)$ in $0 \leq x \leq 1$ accurate to order ε:

$$u_{unif,1} = e^{-x^2/2}\left[1 + (1/4)\varepsilon\left(x^2 - 1\right)^2\right] + \left(1 - e^{-1/2}\right)\left[1 - (1/2)\varepsilon\left(X^2 - 4X\right)\right]e^{-X} \quad (21)$$

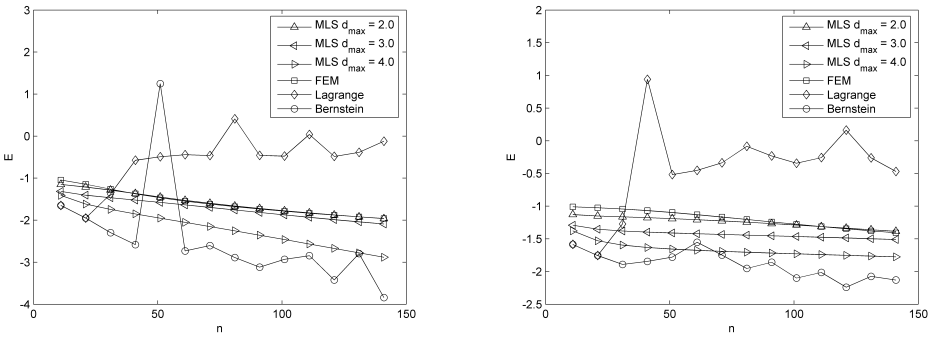
with $X = (1-x)/\varepsilon$. The exact and Galerkin solutions are plotted in Figure 6 for $\varepsilon = 10^{-2}$ and $n = 31$. The norm E and computational cost are presented in Figures 7 and 8.



(a) (b) (c) (d)

Figure 6: Case C, $\varepsilon = 10^{-2}$, $n = 31$, (a) MLS, $d_{max} = 2.0$; (b) FEM; (c) Lagrange; (d) Bernstein

The third perturbation problem selected is an example of an exponential decay of the solution in a narrow region, in this case at the right end of the interval. Figure 6 shows the numerical solutions of the four approximations, with behavior similar to previous cases. From Figures 7 and 8, it can be derived that MLS technique is more accurate than FEM for practically the same computational cost and, besides, MLS possesses greater robustness than Bernstein expansion, which presents some sudden peaks in its $E - n$ curves.



(a) (b)

Figure 7: Norm of error E for case C, (a) $\varepsilon = 10^{-3}$; (b) $\varepsilon = 10^{-4}$

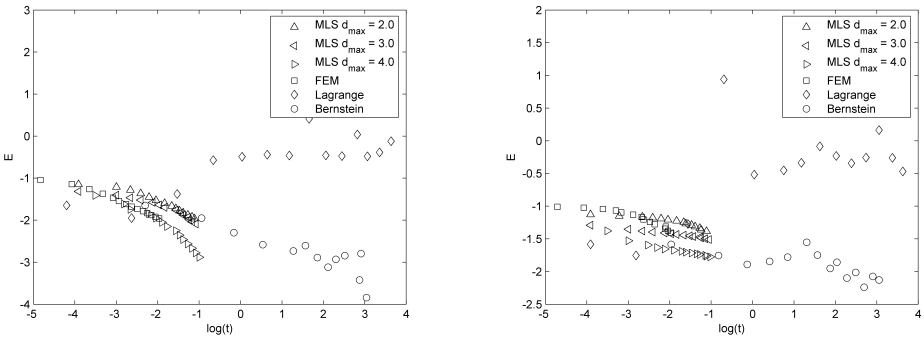


Figure 8: Norm of error E vs computational cost for case C, (a) $\varepsilon = 10^{-3}$; (b) $\varepsilon = 10^{-4}$

5 CONCLUSIONS

In this work, four numerical approximations, MLS, FEM and Lagrange and Bersntein PUs, are reviewed and the fundamentals of the shape functions generation process for each of them are exposed, highlighting the main features of each technique. In order to evaluate the relative performances of the four approximations, they are inserted into a Galerkin implementation of one-dimenisional perturbed variable-coefficient differential equations. These equations, together with the imposed boundary conditions at both ends of the domain, are classic mathematical models of Perturbation Theory. The analytical solutions usually exhibit boundary layer structure, which means that sharp gradients of the field variable take place in narrow regions.

The four numerical approximations are then subjected to experimentation in benchmark problems of the bibliography. The parameters swept in the analysis are the value of the small perturbing parameter ε, the number of nodes n in the discretization and the intrinsic selectable parameter d_{max} of MLS. The nodal distributions selected are patterns of equi-spaced nodes, therefore the $h-$ and $p-$refinements are extended to the whole interval, and the thickness of the boundary layers is not a required datum for the positioning of the nodes. Accuracy, convergence and computational cost are measured and compared in each analysis case. The main conclusion derived from the experiments is that MLS technique can fill the gap between FEM —which is the fastest method, but with a limited accuracy for the reproduction of the boundary layers— and the spectral-type methods —which present excellent accuracy but are much more computationally demanding—. Even using homogeneous nodal patterns, the flexibility of MLS allows to retrieve very accurate results without the computational efforts associated to the generation of full matrices, bridging therefore both extremes of error vs machine time diagrams. Parameter d_{max} can be adjusted to tune the desired accuracy; flexibility is expected to be improved by considering other selectable parameters of MLS, like the order or enrichment of the intrinsic basis and the adequate choice of the weight function.

REFERENCES

[1] Lancaster, P. and Salkauskas, K. Surfaces Generated by Moving Least Squares Methods. *Math. Comput.* (1981) **37**:141–158.

[2] Nayroles, B.; Touzot, G. and Villon, P. Generalizing the Finite Element Method: Diffuse Approximations and Diffuse Elements. *Comput. Mech.* (1992) **10**:307–318.

[3] Belytschko, T.; Lu, Y.Y. and Gu, L. Element-Free Galerkin Methods. *Int. J. Numer. Meth. Engng.* (1994) **37**:229–256.

[4] Liu, G.R. *Mesh Free Methods. Moving beyond the Finite element Method*. CRC Press (2003).

[5] Liu, G.R. and Gu, Y.T. *An introduction to Meshfree Methods and Their Programming*. Springer (2005).

[6] Zienkiewicz, O.C. and Taylor, R.L. *The Finite Element Method. Volume 1: The Basis*. Butterworth - Heinemann. Fifth Edition (2000).

[7] Lorenz, G.G. *Bernstein Polynomials*. Chelsea Publishing Company. Second Edition, New York (1986).

[8] Belytschko, T.; Krongauz, Y.; Organ, D.; Fleming, M. and Krysl, P. Meshless methods: An overview and recent developments. *Comput. Methods in Appl. Mech. Eng.* (1996) **139**(1-4):3–47.

[9] Dolbow, J. and Belytschko, T. An Introduction to Programming the Meshless Element Free Galerkin Method. *Arch. Comp. Methods Engrg.* (1998) **5**(3):207–241.

[10] Valencia, Ó.F.; Gómez-Escalonilla, F.J.; Garijo, D. and López, J. Bernstein polynomials in EFGM. *Proc. IMechE, Part C: J. Mechanical Engineering Science* (2011) **225**(8):1808–1815.

[11] Garijo, D.; Valencia, Ó.F.; Gómez-Escalonilla, F.J. and López, J. Bernstein-Galerkin approach in elastostatics. *Proc. IMechE, Part C: J. Mechanical Engineering Science* (2014) **228**(3):391–404.

[12] Bender, C.M. and Orszag, S.A. *Advanced Mathematical Methods for Scientists and Engineers*. McGraw-Hill Book Company (1978).

[13] Morton, K.W. and Mayers, D.F. *Numerical Solution of Partial Differential Equations*. Cambridge University Press, Second Edition (2005).

[14] Miller, J.J.H.; O'Riordan, E. and Shishkin, G.I. *Fitted numerical methods for singular perturbation problems*. World Scientific, Singapore (1996).