Taylor-Series Expansion Based Numerical Methods: A Primer, Performance Benchmarking and New Approaches for Problems with Non-smooth Solutions

Thibault Jacquemin¹ · Satyendra Tomar¹ · Konstantinos Agathos¹ · Shoya Mohseni-Mofidi² · Stéphane P. A. Bordas¹,³,⁴

Received: 27 April 2019 / Accepted: 29 July 2019 / Published online: 30 August 2019
© CIMNE, Barcelona, Spain 2019

Abstract
We provide a primer to numerical methods based on Taylor series expansions such as generalized finite difference methods and collocation methods. We provide a detailed benchmarking strategy for these methods as well as all data files including input files, boundary conditions, point distribution and solution fields, so as to facilitate future benchmarking of new methods. We review traditional methods and recent ones which appeared in the last decade. We aim to help newcomers to the field understand the main characteristics of these methods and to provide sufficient information to both simplify implementation and benchmarking of new methods. Some of the examples are chosen within a subset of problems where collocation is traditionally known to perform sub-par, namely when the solution sought is non-smooth, i.e. contains discontinuities, singularities or sharp gradients. For such problems and other simpler ones with smooth solutions, we study in depth the influence of the weight function, correction function, and the number of nodes in a given support. We also propose new stabilization approaches to improve the accuracy of the numerical methods. In particular, we experiment with the use of a Voronoi diagram for weight computation, collocation method stabilization approaches, and support node selection for problems with singular solutions. With an appropriate selection of the above-mentioned parameters, the resulting collocation methods are compared to the moving least-squares method (and variations thereof), the radial basis function finite difference method and the finite element method. Extensive tests involving two and three dimensional problems indicate that the methods perform well in terms of efficiency (accuracy versus computational time), even for non-smooth solutions.

Keywords Collocation method · Non-smooth problems · Singularities · Discontinuities · Generalized finite difference · Discretization-corrected particle strength exchange · Linear elasticity · Voronoi diagrams · Stabilization · Visibility criterion · Diffraction criterion · L-shape · Fichera’s corner · Comparison and performance study · Verification · Benchmarking

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s11831-019-09357-5) contains supplementary material, which is available to authorized users.

¹ Institute of Computational Engineering, University of Luxembourg, Maison du Nombre, 6 Avenue de la Fonte, 4364 Esch-sur-Alzette, Luxembourg
² Fraunhofer-Institute for Mechanics of Materials IWM, Freiburg, Germany
³ Department of Medical Research, China Medical University Hospital, China Medical University, Taichung, Taiwan
⁴ Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan

1 Introduction
We focus in this paper on Taylor-series expansion based collocation approaches for partial differential equations (PDEs). In these methods, instead of writing the problem in an average sense, as in Galerkin methods, the strong form is written explicitly at a set of computational points, distributed over the domain. Derivative operators are computed through the use of stencils of points, which can be built in different ways. A lot of work has been done in this field since the early 1900s, and collocation methods are regaining interest, due to the advent of massively parallel computing, which lends itself very naturally to these methods. Our goal in this paper is to facilitate the understanding of newcomers to the field,
help choose optimal parameters, to benchmark the methods and, finally, to propose novel approaches to deal with non-smooth solutions. Specifically, we aim to:

- briefly review approaches to alleviate the mesh burden in computational mechanics;
- provide a gradual, clear and detailed introduction to Taylor-series expansion based collocation approaches;
- investigate the sensitivity of the above approaches to the parameters involved;
- provide recommendations on the methods optimal parameters;
- propose and experiment on a computational approach to handle sharp corners and singularities;
- provide a comprehensive investigation of the relative performance of some of the most popular approaches;
  - for smooth problems;
  - for rough and singular problems with low solution regularity;
- facilitate future benchmarking by providing all data files, including geometries, point distributions, loading and boundary conditions, solution fields, to help benchmarking existing and new methods.

Numerical methods have been under development for approximately 80 years. The first methods which were developed were finite difference methods, which focus on the approximation of the differential operator. The first known reference is the invention of finite difference methods for partial differential equations, in the work of Runge [1]. The idea was to use stencils of points in order to approximate differential operators using finite differences. In their initial form, finite difference methods were largely limited to Cartesian domains in space or to time approximations.

This limitation of finite difference methods to the union of Cartesian domains may have been the motivation behind the development of alternative methods including the Ritz method [2] and the Galerkin finite element method [3]. Contrary to finite difference schemes, finite element methods were able to handle arbitrarily complex geometries, at the cost of the generation of a mesh, i.e. a cover of the volume with simple shapes including tetrahedral, hexahedral and prismatic elements.

Shortly after the introduction of the concept of mesh, in 1977 with the creation of the smoothed particle hydrodynamics method [4], the notion of methods which would later become known as mesh-free methods came about. SPH enabled the solution of problems which caused difficulties to finite elements, in particular those involving fluid flow, fragmentation and very large deformations.

The finite element concept of mesh, closely related to that of interpolation and approximation comes with at least five associated challenges:

- the mesh should conform to the potentially complex geometry of the domain and hence be regenerated, at least partially, for each change in the geometry of the component under consideration;
- for moving boundaries, the mesh must be regenerated at each geometrical change in the boundary;
- the aspect ratios of the elements should be controlled to ensure accuracy, in large deformations, this includes ensuring that the elements do not become too deformed or inverted during deformation;
- locking problems have to be accounted for when small parameters appear within the PDE, e.g. for thin plates and shells or incompressible materials, warranting the development of new locking-free formulations;
- stability of approximation schemes for coupled multi-field problems must be ensured, leading to the requirement of hybrid methods.

Some of these challenges may well have motivated the inception of alternative methods known at the time as meshless or meshfree methods [5–10]. The original idea behind such methods was to decrease the burden posed by the generation and regeneration of a mesh. In particular, the Bubnov–Galerkin or Petrov–Galerkin methods relax some of the constraints associated with locating the points used to construct the approximation and thus simplify local refinement. Nonetheless, these methods rely on non-polynomial approximations which are usually non-interpolating, thus posing additional difficulties associated with enforcing boundary conditions and numerical integration. For many of these methods, numerical integration requires a background mesh or local integration rules on complex domains such as lenses. In their initial formulation, meshfree methods are computationally expensive, which somewhat limits their application to industrial problems. The 2008 review on implementation and recent advances in meshfree methods is a possible reference [11].

Contemporarily to the birth of meshfree methods, partition of unity approaches see the light of day [12, 13]. In their original form, they enable the introduction of known features about the solution within the finite element approximation. Either this known feature is computed numerically (as in the generalised finite element method [14]) or they are extracted from analytical knowledge about the solution, as in the extended finite element method (XFEM). These methods, born in parallel to meshfree methods create an intermediate world between finite element methods and meshfree methods, and have similarities with both. For instance, methods such as XFEM enable the simulation of
propagating discontinuities in the field variable or its derivative with minimal or no remeshing, whilst some versions of partition of unity methods require special treatment of boundary conditions. Some of the most exciting applications of partition of unity methods include fracture mechanics either as enriched finite elements \([15–22]\) or as enriched meshfree methods \([23–29]\). Note that such partition of unity methods were also used to permit the implicit treatment of (evolving) discontinuities using level set methods, including an implicit description of the boundary of the computational domain \([30, 31]\). Several recent reviews can be consulted for an overview on partition of unity methods \([32]\).

A decade after the appearance of Galerkin meshfree methods, isogeometric analysis (IGA) approaches saw the light \([33]\). Their primary goal is to facilitate the connection between computer aided design (CAD) and computer aided engineering (CAE) with numerical analysis by using the same functions used to describe the geometry of the object to also approximate the unknown field variables. In this way, the method is able to represent complex geometries exactly. During early stage design iterations, any change in the geometry is automatically inherited by the approximation scheme for the field variables, thereby simplifying the iterative design process. Isogeometric analysis boundary element methods (IGABEM) \([34–43]\) transcend the intrinsic limitations of IGA within a finite element context, in particular the requirement of 3D volume parameterisation, akin to hexahedral meshing \([44–46]\). IGA shares many common points with meshfree methods, in particular its natural ability to deal with high order approximations, which makes it suitable to handle Kirchhoff–Love plates and shells and high-order PDEs. Various approaches combining enrichment with IGA were introduced \([47]\). The reader can refer to the recent overview and computer implementation aspects of IGA presented in \([48]\).

To overcome the most negative aspects of IGA, i.e. the need for structured Cartesian parameterisation associated with the tensor product nature of the method as well as the consequential difficulties associated with local mesh refinement, the geometry-independent field approximation method (GIFT) was proposed by Atroshchenko and colleagues in a series of papers \([49]\), which relaxes the strict requirement of using the same basis functions to represent the geometry and the field variables, and, hence enables the local refinement of the field approximation independently of the non-uniform rational B-splines (NURBS) representation of the boundary. This therefore maintains the tight coupling between the CAD and the analysis of a given component, without requiring the use of NURBS for field approximations, which has been shown to be suboptimal in certain situations, for example, for problems with corner singularities, or weakly regular solutions.

Contemporarily with IGA, methods based on implicit treatments of boundaries have continued to develop, thanks to the combined efforts of engineers and applied mathematicians \([50–59]\).

In light of the above summary, the finite element methods and the meshfree methods seem to have been abandoned by the computational mechanics community. Collocation methods, however, have been continuously studied from the mid-1950s to date. Collocation methods have been reintroduced into the literature thanks to the recrudescence of advanced computing hardware such as graphical processing units and Xeon Phis, among others. Such computing architectures have memory architectures which are well suited to handling similar data shapes such as the row of a stiffness matrix provided by collocation approaches.

Now that we have painted an impressionist picture of the path towards mesh-burden reduction, subsequent to the birth of finite difference methods and finite element methods, we proceed to introducing collocation methods, which we classify broadly into two groups. The first group includes all methods which use an approximation of the differential operator to solve the Partial Differential Equation (PDE). In this paper, two methods of the first group, which use a Taylor’s series expansion to approximate the field derivatives, are considered. These methods are the Generalized Finite Difference (GFD) method and the discretization-corrected particle strength exchange (DC PSE) method, see the discussion below. The second group includes methods which are based on an approximation of the unknown field. The most prominent method in this second group is the Moving Least Squares (MLS) method \([60, 61]\) that is used in the Element Free Galerkin (EFG) method \([5]\).

The idea of generalizing the Finite Difference Method (FDM) began in 1953 with MacNeal \([62]\), and in 1960 with Forsyth and Wasow \([63]\). They proposed a method to transform an irregular node distribution over the domain into a regular sub-domain on which the FDM can be applied. In 1962, Jensen \([64]\) introduced the basis of the Generalized Finite Difference Method. The method, described for two-dimensional problems, uses a six-node star and a second order Taylor’s series expansion to approximate the spatial derivatives up to the second order. In 1980, Liszka and Orkisz \([65]\) presented a method based on an eight-node star which allows obtaining a more stable approximation of the derivatives. The method is based on some selected weights and a mean least square approximation of the derivatives. In 1998, Orkisz \([66]\) presented a more complete version of the GFD method covering various subjects, such as the application of the method to the Galerkin framework, and the use of a posteriori error estimators for model adaptivity.

The Particle Strength Exchange (PSE) method was introduced by Degond and Mas-Gallic in 1989 \([67]\). Initially developed to approximate the diffusion operator...
of the convection-diffusion equations, the method has been generalized by Eldredge et al. in 2002 [68] in order to approximate any derivative order. The Discretization-Corrected Particle Strength Exchange (DC PSE) method has been introduced by Schrader et al. [69] in 2010 in order to account for the discretization of the domain in the operator calculation. This allows removing the discretization error, which led to the name being “Discretization-Corrected”.

The GFD and the DC PSE methods show many similarities, which are analyzed in this paper. Both methods are based on a set of parameters. Some methods, aiming at improving the accuracy of the solution, are presented and analyzed in the paper. The two considered methods are compared to other well known collocation methods. These methods can be classified into two categories. The methods based on an approximation of the differential operator such as the GFD and the DC PSE methods form the first group, and the methods based on an approximation of the field form the second group. For each category, the following methods are considered:

- Differential operator approximation;
  - Generalized Finite Difference method (GFD);
  - Discretization-Corrected Particle Strength Exchange method (DC PSE);
  - Radial Basis Function Finite Difference method (RBF-FD).
- Field approximation;
  - Moving Least Square method (MLS);
  - Interpolating Moving Least Square method (I-MLS).

A brief outline of the remainder of the paper is as follows. In Sect. 2, we briefly describe each of the methods considered in this paper. In Sect. 3, three linear elastic problems, for which an analytical solution is known (i.e. a cylinder under internal pressure, a sphere under internal pressure, and an L-shape domain in mode I loading), are presented. Moreover, the error norms are also introduced in Sect. 3. The methods are compared for the $L_2$ norm and the $L_{\infty}$ norm of the error for the calculated stress components. In Sect. 4, we present a parametric sensitivity study of the methods. This includes a study of the weight function, of the correction function (for DC PSE), and of the number of support nodes. In Sect. 5, we present some improvement methods for the GFD and the DC PSE methods, such as Voronoi diagram, stabilization, and criteria for support node selection for singular problems. In Sect. 6, we present some benchmarking results from the comparison of the various methods listed above. We also present some results on convergence rates and computational expenses of these methods. In Sect. 7, we present the results of the GFD method for 3D problems. Moreover, we compare our results with finite elements results obtained using the commercial package ABAQUS [70]. Some conclusions are drawn in Sect. 8. Finally, a detailed comparison of the GFD and DC PSE methods for 1D problems is provided in the “Appendix”.

As a novelty of our work, we list two main components, namely (1) a detailed comparison of the GFD and DC PSE methods for 2D and 3D linear elastic problems (such as the pressurized cylinder and the L-shape domain in mode I loading), and (2) the assessments of the improvement methods as well as the identification and comparison of variations on DC PSE methods. To the best of the authors’ knowledge, these studies are not found in the literature.

### 2 Collocation Methods

#### 2.1 Introduction

Solving a problem by collocation methods consists in solving the set of PDEs only at collocation centers. A number of nodes spread over the domain are used to estimate the derivatives at the collocation centers. In most collocation methods, the equations are solved at the nodes. The problem being solved locally, the strong form of the PDEs is considered. In this paper, we primarily consider the GFD and DC PSE methods. These methods are compared to the MLS approximation method and to the RBF-FD method, which are among the most popular methods of approximation in the framework of collocation methods. In the remainder of the present section, we present the principles of each of these methods. In order to facilitate the comprehension of the methods, the case of a two dimensional problem in a Cartesian coordinate system is considered. The GFD and DC PSE methods are also presented and compared for the case of a 1D problem in “Appendix”.

In the sections below, the spatial coordinates are denoted by $x$ and $y$. The coordinates of a node $X$ are then $X = [x, y]^T$. The subscripts $c$ and $p$ are used to identify, respectively, the collocation node and a particle “$p$”. The first and second derivatives in the two spatial directions are denoted by $\partial_1, \partial_2, \partial_2^2, \partial_1\partial_2, \partial_1^2, \partial_2^2$. In the general case, these derivatives are written as $D^{n_1,n_2}(X_c)$, where $n_1$ and $n_2$ are, respectively, the derivation orders in the directions $x$ and $y$. 

© Springer
In 1972, Jensen [64] presented a method to approximate derivatives using the Taylor’s series expansion on an irregular grid. This method is known as the Generalized Finite Difference (GFD) method.

For the GFD method, the derivatives are calculated at collocation nodes $X_c = [x_c, y_c]^T$ using a Taylor’s series expansion of the unknown field. The field derivatives at $X_c$ are computed in order to reproduce the known field values $f(X_{pi})$ for a number of points $X_{pi} = [x_{pi}, y_{pi}]^T$. The number of selected points depends on the approximated derivative order.

### 2.2.2 Differential Operator Approximation

Considering a function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, the Taylor’s series expansion of this function at $X_{pi}$ in the vicinity of a collocation node $X_c$ is written:

$$f(X_{pi}) = \sum_{i=0}^{+\infty} \sum_{j=0}^{+\infty} \frac{\partial^i j f(X_c) (x_{pi} - x_c)^i (y_{pi} - y_c)^j}{i! j!}.$$

For ease of notations, we write the second order approximation of the function $f$ at the point $X_{pi}$ near $X_c$ as $f_h(X_{pi})$. For $f_h(X_{pi})$, Eq. (1) becomes:

$$f_h(X_{pi}) = f(X_c) + (x_{pi} - x_c) \frac{\partial f(X_c)}{\partial x} + (y_{pi} - y_c) \frac{\partial f(X_c)}{\partial y} + \frac{(x_{pi} - x_c)^2}{2!} \frac{\partial^2 f(X_c)}{\partial x^2} + \frac{(x_{pi} - x_c)(y_{pi} - y_c)}{2!} \frac{\partial^2 f(X_c)}{\partial x \partial y} + \frac{(y_{pi} - y_c)^2}{2!} \frac{\partial^2 f(X_c)}{\partial y^2}.$$

Equation (2) can be cast in a matrix form:

$$\begin{bmatrix} \frac{\partial f(X_c)}{\partial x} \\ \frac{\partial f(X_c)}{\partial y} \\ \frac{\partial^2 f(X_c)}{\partial x^2} \\ \frac{\partial^2 f(X_c)}{\partial x \partial y} \\ \frac{\partial^2 f(X_c)}{\partial y^2} \end{bmatrix} + \begin{bmatrix} x_{pi} - x_c \\ y_{pi} - y_c \end{bmatrix} = f_h(X_{pi}) - f(X_c).$$

In order to determine an approximation of the field derivatives $Df(X) = \left[ \frac{\partial f(X)}{\partial x} \frac{\partial f(X)}{\partial y} \frac{\partial^2 f(X)}{\partial x^2} \frac{\partial^2 f(X)}{\partial x \partial y} \frac{\partial^2 f(X)}{\partial y^2} \right]^T$ (five unknowns), Eq. (2) will be written for five nodes $X_{pi}$ in the vicinity of $X_c$ (see Fig. 2). Thereby, a linear system is obtained.

![Fig. 1 Collocation node support](image1.png)

![Fig. 2 Five nodes support of a collocation node $X_c$](image2.png)
If more than five points are chosen for solving Eq. (3), the system is overdetermined. In that case, the derivatives of the approximation at the collocation node \(X_c\) can be approximated as a function of \(f_0(X_c)\) and \(f_k(X_{pi})\) by solving the above system. If more than five points \(X_{pi}\) are chosen for solving Eq. (3), the system is overdetermined. In that case, the derivatives at \(X_c\) leading to the minimum error can be determined using the least square method.

### 2.2.3 Overdetermined Approximation

If an arbitrary number of nodes \(m\) is selected, the derivatives are determined using the mean least square method. A mean least square functional \(B\) is presented below for the two dimensional case for both, the general form (5) and the second order approximation (6). A weight function \(w\) is typically used to balance the contribution of each node in the approximation. While a wide range of functions can be used as weight, 3\(^{rd}\) and 4\(^{th}\) order splines are usually preferred.

\[
B_n(X_c) = \sum_{i=1}^{m} w(X_{pi} - X_c) \left[ \frac{\sum_{j=0}^{\infty} \sum_{k=0}^{\infty} \frac{\partial^j f(X_{pi}) (y_{pi} - y_c)^j}{\partial x^j \partial y^k}}{j!} \right]^2.
\]

The derivatives \(\textbf{D}f(X_c)\) that best approximate the known field values using the Taylor’s series expansion, minimize \(B_n(X)\) when:

\[
\left. \frac{\partial B_n(X)}{\partial \textbf{D}f(X)} \right|_{x=X_c} = 0.
\]

Equation (7) can be written as a linear system of the form:

\[
\textbf{A}(X_c) \textbf{D}f(X_c) = \textbf{E}(X_c) \textbf{F}(X_c).
\]

For the two dimensional second order case, the matrices \(\textbf{A}(X_c), \textbf{E}(X_c)\) and \(\textbf{F}(X_c)\) are:

\[
\textbf{A}(X_c) = \begin{bmatrix} m_{11} & m_{12} & \ldots & m_{15} \\ m_{21} & m_{22} & \ldots & m_{25} \\ \vdots & \vdots & \ddots & \vdots \\ m_{51} & m_{52} & \ldots & m_{55} \end{bmatrix} \in \mathbb{R}^{5 \times 5},
\]

\[
\textbf{E}(X_c) = \begin{bmatrix} -m_{01} & m_{01,1} & \ldots & m_{01,m} \\ -m_{02} & m_{02,1} & \ldots & m_{02,m} \\ \vdots & \vdots & \ddots & \vdots \\ -m_{05} & m_{05,1} & \ldots & m_{05,m} \end{bmatrix} \in \mathbb{R}^{5 \times (m+1)},
\]

\[
\textbf{F}(X_c) = \begin{bmatrix} f(X_c) f(X_{p1}) f(X_{p2}) \ldots f(X_{pm}) \end{bmatrix}^T,
\]

where the moments \(m_{ij,k}\) and \(m_{ij}\) correspond to:

\[
m_{ij,k} = w(X_{pk} - X_c) P_{(i+1,k)}(X_c) P_{j+1,k}(X_c),
\]

\[
m_{ij} = \sum_{k=1}^{m} m_{ij,k}.
\]

The matrix \(\textbf{P}(X_c) \in \mathbb{R}^{5 \times m}\) is written as follows:

\[
\textbf{P}(X_c) = \begin{bmatrix} 1 & (x_{p1} - x_c) & (y_{p1} - y_c) & \ldots & (x_{pm} - x_c) & (y_{pm} - y_c) \\ (x_{p1} - x_c)^2 & (y_{p1} - y_c)^2 & \ldots & (x_{pm} - x_c)^2 & (y_{pm} - y_c)^2 \\ (x_{p1} - x_c)(y_{p1} - y_c) & (x_{p2} - x_c)(y_{p2} - y_c) & \ldots & (x_{pm} - x_c)(y_{pm} - y_c) \\ \end{bmatrix}.
\]
The derivative vector $\mathbf{Df}(\mathbf{X}_c)$ can then be determined as a function of $\mathbf{F}(\mathbf{X}_c)$:

$$\mathbf{Df}(\mathbf{X}_c) = \mathbf{A}(\mathbf{X}_c)^{-1}\mathbf{E}(\mathbf{X}_c)\mathbf{F}(\mathbf{X}_c). \quad (14)$$

The approximated derivatives are determined by solving the linear system (14). These derivatives are, by definition, consistent with each other as they participate in reproducing the unknown field values based on a Taylor’s series expansion.

### 2.3 Discretization-Corrected Particle Strength Exchange Method (DC PSE)

#### 2.3.1 General DC PSE Operator

The DC PSE method is based on a Taylor’s series expansion of the unknown field. A convolution function is used to select the approximated derivative term. All the other unknown terms of the expansion are canceled out by the convolution function. The Taylor’s series expansions presented in Eqs. (1) and (2) are convoluted by a function $\eta$ over a domain $\Omega$:

$$\int_{\Omega} f(\mathbf{X}_p)\eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p$$

$$= \sum_{i=0}^{+\infty} \sum_{j=0}^{+\infty} \int_{\Omega} \frac{\partial^{i+j}f(\mathbf{X}_c)}{\partial x^i\partial y^j} \frac{(x_p - x_c)^i}{i!}\frac{(y_p - y_c)^j}{j!} \eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p. \quad (15)$$

The second order approximation of Eq. (15) is written as follows:

$$\int_{\Omega} f_h(\mathbf{X}_p)\eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p$$

$$= \int_{\Omega} f(\mathbf{X}_c)\eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p$$

$$+ \int_{\Omega} \frac{\partial f(\mathbf{X}_c)}{\partial x}(x_p - x_c)\eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p$$

$$+ \int_{\Omega} \frac{\partial f(\mathbf{X}_c)}{\partial y}(y_p - y_c)\eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p$$

$$+ \int_{\Omega} \frac{\partial^2 f(\mathbf{X}_c)}{\partial x^2}(x_p - x_c)^2\eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p$$

$$+ \int_{\Omega} \frac{\partial^2 f(\mathbf{X}_c)}{\partial x\partial y}(x_p - x_c)(y_p - y_c)\eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p$$

$$+ \int_{\Omega} \frac{\partial^2 f(\mathbf{X}_c)}{\partial y^2}(y_p - y_c)^2\eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p. \quad (16)$$

Equations (15) and (16) can be simplified by introducing the moments $M_{ij}(\mathbf{X}_c)$ which are defined as follows:

$$M_{ij}(\mathbf{X}_c) = \int_{\Omega} \frac{(x_p - x_c)^i}{i!}\frac{(y_p - y_c)^j}{j!} \eta(\mathbf{X}_p - \mathbf{X}_c)d\mathbf{X}_p. \quad (17)$$

Considering that the field is relatively smooth in $\Omega$, the integration can be transformed into a discrete summation over the nodes of the domain. Constant values $V_p$ are associated to each of the nodes of the domain. The moments then become:

$$M_{ij}(\mathbf{X}_c) = \sum_{p \in \Omega} V_p \frac{(x_p - x_c)^i}{i!}\frac{(y_p - y_c)^j}{j!} \eta(\mathbf{X}_p - \mathbf{X}_c). \quad (18)$$

The values $V_p$ associated to the particles $p$ are hard to determine in the general case. Assuming a uniform distribution of the particles over the domain, these values are typically set to unity. Equation (18) then becomes:

$$M_{ij}(\mathbf{X}_c) = \sum_{p \in \Omega} \frac{(x_p - x_c)^i}{i!}\frac{(y_p - y_c)^j}{j!} \eta(\mathbf{X}_p - \mathbf{X}_c). \quad (19)$$

Using these moments, Eqs. (15) and (16), respectively, become:

$$\sum_{p \in \Omega} f_h(\mathbf{X}_p)\eta(\mathbf{X}_p - \mathbf{X}_c)$$

$$= \sum_{p \in \Omega} \sum_{i=0}^{+\infty} \sum_{j=0}^{+\infty} \frac{\partial^{i+j}f(\mathbf{X}_c)}{\partial x^i\partial y^j} M_{ij}(\mathbf{X}_c), \quad (20)$$

The selection of an appropriate function $\eta$ allows approximating the desired derivative $D^{i+j}f(\mathbf{X}_c) \approx \frac{\partial^{i+j}f(\mathbf{X}_c)}{\partial x^i\partial y^j}$ by setting all the moments to zero except the one multiplying $D^{i+j}f(\mathbf{X}_c)$, which is set to unity. Equation (20) can then be written:

$$D^{i+j}f(\mathbf{X}_c) = \sum_{p \in \Omega} f_h(\mathbf{X}_p)\eta(\mathbf{X}_p - \mathbf{X}_c)$$

$$\text{with } M_{ij}(\mathbf{X}_c) = 1$$

$$M_{ij}(\mathbf{X}_c) = 0 \quad \text{if } (i,j) \neq (k,l). \quad (22)$$

#### 2.3.2 The Convolution Function

In order to satisfy at each node of the domain the moment condition (22), the convolution function $\eta$ needs to be chosen.
The shape of the weight function has a significant impact on the solution as it balances the contribution of each node of the support in the field derivative approximation. Three types of weight functions can be considered in particular. These are:

The exponential weight functions:

\[ w(s) = \begin{cases} 
\alpha^s e^{-s} & \text{if } s \leq 1 \\
0 & \text{if } s > 1,
\end{cases} \tag{25} \]

where \( \alpha \) is an exponent and \( e \) is a shape parameter.

3rd order spline weight functions:

\[ w(s) = \begin{cases} 
\frac{2}{3} - 4s^2 + 4s^3 & \text{if } s \leq 0.5 \\
\frac{4}{3} - 4s + 4s^2 - \frac{4}{3}s^3 & \text{if } 0.5 < s \leq 1 \\
0 & \text{if } s > 1,
\end{cases} \tag{26} \]

4th order spline weight functions:

\[ w(s) = \begin{cases} 
1 - 6s^2 + 8s^3 - 3s^4 & \text{if } s \leq 1 \\
0 & \text{if } s > 1.
\end{cases} \tag{27} \]

A typical convolution function, composed of a polynomial correction function \( P \) and a vector of coefficients \( a \), is written as:

\[ \eta(X_p - X_c) = P(X_p - X_c)^T a \ w(s_p). \tag{28} \]

In order for this convolution function to satisfy the moment condition, the polynomial order shall be of at least the derivation order.

2.3.3 Correction Function Calculation

The coefficient vector \( a \) is the solution of a linear system \( A_M(X_c)a = B_M \), where the left side of the equation corresponds to the moments calculation for the unknown convolution function \( \eta \). The vector \( B_M \) corresponds to the moment condition which needs to be satisfied to obtain the desired derivative approximation. For instance, in order to approximate the derivative \( D^{2,0}\eta(X_c) \), the system is:

\[
\begin{align*}
M_{0,0}(X_c) &\iff \sum_{p \in \Omega_c} P(X_p - X_c)^T a w(s_p) = 0, \\
M_{1,0}(X_c) &\iff \sum_{p \in \Omega_c} (x_p - x_c)P(X_p - X_c)^T a w(s_p) = 0, \\
M_{0,1}(X_c) &\iff \sum_{p \in \Omega_c} (y_p - y_c)P(X_p - X_c)^T a w(s_p) = 0, \\
M_{2,0}(X_c) &\iff \sum_{p \in \Omega_c} \frac{(x_p - x_c)^2}{2!}P(X_p - X_c)^T a w(s_p) = 1, \\
M_{1,1}(X_c) &\iff \sum_{p \in \Omega_c} (x_p - x_c)(y_p - y_c)P(X_p - X_c)^T a w(s_p) = 0, \\
M_{0,2}(X_c) &\iff \sum_{p \in \Omega_c} \frac{(y_p - y_c)^2}{2!}P(X_p - X_c)^T a w(s_p) = 0.
\end{align*}
\tag{29} \]

Considering the vector \( Q(X_c, X_p) = \begin{bmatrix} 1 & (x_p - x_c) & (y_p - y_c) & \frac{(x_p - x_c)^2}{2!} & \frac{(y_p - y_c)^2}{2!} \end{bmatrix}^T \), the correction function basis \( P \) and the weight function \( w \), the coefficients of the matrix \( A_M \in \mathbb{R}^{6 \times 6} \) can be written:

\[ A_{M,(i,j)}(X_c) = \sum_{p \in \Omega_c} Q_i(X_c, X_p) P_j(X_p - X_c) w(s_p). \tag{30} \]

Having solved the system of equations \( a = A_M^{-1}(X_c)B_M \), the derivative \( D^{2,0}\eta(X_c) \) can be approximated with Eq. (22) as a function of \( f_0(X_p), \rho \in \Omega_c \). From a computational point of view, it shall be noted that the inversion of the matrix \( A_M(X_c) \) only needs to be performed once per collocation node \( X_c \). If the approximation of another derivative is required for the solution of the partial differential equation, only the moment condition set by the vector \( B_M \) is updated.

2.3.4 Identified Variations of DC PSE Method

It can be observed from the DC PSE method presented above that setting the moment \( M_{0,0}(X_c) \) to zero is not necessary. The value \( f(X_c) \) is determined in the global problem, and does not need to be canceled by a null moment \( M_{0,0}(X_c) \). Based on this remark, three approaches can be considered. These approaches have respectively been labeled DCPSE0, DCPSE1 and DCPSE2.

- **DCPSE0**: \( M_{0,0}(X_c) \) is set to zero (case presented above);
2.4 Radial Basis Function Finite Difference Method

2.4.1 Principle

In [72, 73], Kansa introduced the idea of using Radial Basis Functions for solving differential equations over a domain. Contributions to the RBF-FD method were made later by Driscoll and Fornberg [74], Shu [75], Fornberg [76, 77] and Davydov [78, 79]. The principle of the RBF-FD method is to determine an approximation of the differential operator at a collocation node \( \mathbf{x}_c \) based on a linear combination of the field values at the nodes \( \mathbf{x}_p \) nearby. Considering \( m \) nodes in the support of \( \mathbf{x}_c \), the aim is to determine a set of weights \( \lambda \), written in a vector form as \( \mathbf{W}(\mathbf{x}_c) = [\lambda_{p1} \ldots \lambda_{pm}] \), so that:

\[
D^{\nu,n} f(\mathbf{x}_c) = \mathbf{W}(\mathbf{x}_c) \begin{bmatrix} f(\mathbf{x}_{p1}) \\ \vdots \\ f(\mathbf{x}_{pm}) \end{bmatrix}.
\]  

(31)

The RBF-FD method assumes that Eq. (31) is exact for all radial basis functions \( \varphi \) centered at each node of the support.

2.4.2 Radial Basis Functions

Various classes of RBFs can be used for the purpose of the RBF-FD method. Depending on the type of RBF, one or two shape parameters need to be selected. The selection of the shape parameter(s) is critical to the accuracy of the solution as it balances the contribution of the neighboring nodes in the derivative approximation. The main classes of RBFs are presented in Table 1.

In order for the overall system matrix to be sparse, the radial basis functions are chosen with a compact support.

2.4.3 Obtaining the Differentiation Matrix

Replacing the unknown field \( f \) by the selected radial basis function \( \varphi_{x_c} \) centered in \( \mathbf{x}_c \), Eq. (31) becomes:

\[
D^{\nu,n} \varphi_{x_c}(\mathbf{x}_c) = \mathbf{W}(\mathbf{x}_c) \begin{bmatrix} \varphi_{x_c}(\mathbf{x}_{p1}) \\ \vdots \\ \varphi_{x_c}(\mathbf{x}_{pm}) \end{bmatrix}.
\]  

(32)

The weights are determined so that the approximation is exact for every radial basis function centered at each support node of the collocation node support. A linear system of equations is thus obtained:

\[
\begin{bmatrix} \varphi_{X_{p1}}(\mathbf{x}_{p1}) & \varphi_{X_{p1}}(\mathbf{x}_{p2}) & \ldots & \varphi_{X_{p1}}(\mathbf{x}_{pm}) \\ \varphi_{X_{p2}}(\mathbf{x}_{p1}) & \varphi_{X_{p2}}(\mathbf{x}_{p2}) & \ldots & \varphi_{X_{p2}}(\mathbf{x}_{pm}) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{X_{pm}}(\mathbf{x}_{p1}) & \varphi_{X_{pm}}(\mathbf{x}_{p2}) & \ldots & \varphi_{X_{pm}}(\mathbf{x}_{pm}) \end{bmatrix} \begin{bmatrix} \lambda_{p1} \\ \lambda_{p2} \\ \vdots \\ \lambda_{pm} \end{bmatrix} = \begin{bmatrix} D\varphi_{X_{p1}}(\mathbf{x}_c) \\ D\varphi_{X_{p2}}(\mathbf{x}_c) \\ \vdots \\ D\varphi_{X_{pm}}(\mathbf{x}_c) \end{bmatrix}.
\]  

(33)

Additional constraints can be added to the system in order for the radial basis functions to reproduce exactly polynomials of at least the derivative order. This also ensures a certain regularity of the solution. For instance, for the case of a 2D problem and for a first order derivation in the \( y \) direction, a first order polynomial basis can be added to the set of RBFs. Thereby, the system presented in Eq. (33) becomes:

\[
\begin{bmatrix} \varphi_{X_{p1}}(\mathbf{x}_{p1}) & \ldots & \varphi_{X_{p1}}(\mathbf{x}_{pm}) & 1 & x_{p1} & y_{p1} \\ \varphi_{X_{p2}}(\mathbf{x}_{p1}) & \ldots & \varphi_{X_{p2}}(\mathbf{x}_{pm}) & 1 & x_{p2} & y_{p2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \varphi_{X_{pm}}(\mathbf{x}_{p1}) & \ldots & \varphi_{X_{pm}}(\mathbf{x}_{pm}) & 1 & x_{pm} & y_{pm} \end{bmatrix} \begin{bmatrix} \lambda_{p1} \\ \lambda_{p2} \\ \vdots \\ \lambda_{pm} \\ \lambda_{m+1} \\ \lambda_{m+2} \\ \lambda_{m+3} \end{bmatrix} = \begin{bmatrix} D^{0,1}\varphi_{X_{p1}}(\mathbf{x}_c) \\ D^{0,1}\varphi_{X_{p1}}(\mathbf{x}_c) \\ \vdots \\ D^{0,1}\varphi_{X_{pm}}(\mathbf{x}_c) \end{bmatrix}.
\]  

(34)
Once the weights $W(X_c)$ determined by the solution of Eq. (33) or Eq. (34), the derivative $D^\alpha \frac{\partial}{\partial x} f(X_c)$ can be approximated.

### 2.5 Moving Least Square Approximation

#### 2.5.1 Field Approximation

The MLS method has been introduced by Lancaster and Salkauskas in 1981 [60]. Interpolation for the lowest order has been introduced by Shepard in 1968 [61]. The method has been widely used in the context of the Element-Free Galerkin (EFG) method [5] and in the framework of collocation for the FEM [81]. The MLS method consists in approximating the unknown field using a function basis. Differentiation of the approximated field and solution of a partial differential equation then becomes possible. The unknown field can be approximated with various types of functions depending on the considered application. Polynomial functions are typically used for linear elasticity problems.

Considering a polynomial basis $P(X)$ and a coefficient vector $a(X_c)$, an approximation of the field $f$ around a collocation node $X_c$ can be written as follows:

$$ f(x, X_c) = P(X)^T a(X_c). $$

The coefficients $a(X_c)$ are determined by minimizing the error of the approximated field over a set of $m$ nodes around the collocation node. The error is weighted by a function $w$ centered in $X_c$. The minimization problem can be expressed by a functional $B(X_c)$:

$$ B(X_c) = \sum_{i=1}^{m} w(x) \left[ P(x)^T a(x) - f(x) \right]^2. $$

The resulting error represented by the functional $B(X_c)$ is minimal when:

$$ \frac{\partial B(X_c)}{\partial a(X_c)} = 0. $$

This problem is a linear system of the form:

$$ A(X_c) a(X_c) = E(X_c) F(X_c). $$

For a polynomial vector of size $n$, the matrices $A(X_c)$, $E(X_c)$ and $F(X_c)$ correspond to:

$$ A(X_c) = \begin{bmatrix} m_{01} & m_{02} & \cdots & m_{0n} \\ m_{11} & m_{12} & \cdots & m_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{bmatrix} \in \mathbb{R}^{n \times n}, $$

$$ E(X_c) = \begin{bmatrix} m_{01,1} & m_{01,2} & \cdots & m_{01,m} \\ m_{02,1} & m_{02,2} & \cdots & m_{02,m} \\ \vdots & \vdots & \ddots & \vdots \\ m_{0n,1} & m_{0n,2} & \cdots & m_{0n,m} \end{bmatrix} \in \mathbb{R}^{n \times m}, $$

$$ F(X_c) = \begin{bmatrix} f(X_{p1}) \\ f(X_{p2}) \\ \vdots \\ f(X_{pm}) \end{bmatrix}^T, $$

where

$$ m_{ij} = w(X_c - X_{pk}) P(X)^T (X_{pk}) P(X_{pk}), $$

$$ m_i = \sum_{k=1}^{m} m_{ik}. $$

For a two dimensional second order case with a polynomial basis, $P$ is chosen as:

$$ P^X_{yk}(X_{pk}) = \begin{bmatrix} 1 \\ (x_{pk} - x_c) \\ (y_{pk} - y_c) \\ (x_{pk} - x_c)^2 \\ (y_{pk} - y_c)^2 \end{bmatrix}. $$

As for the RBF-FD method, the dimension of the function basis $P$ can be augmented in order to improve the regularity of the solution.

#### 2.5.2 Boundary Condition Enforcement

The MLS method does not interpolate the field values. This impacts the Dirichlet boundary condition enforcement. Unlike the methods presented in the previous sections, the Dirichlet boundary condition is applied to the approximated field rather than directly to the degree of freedom solved in the linear system. The shape functions of the approximated field need to be calculated at the Dirichlet boundary nodes and are used to set the boundary condition. This leads to a slightly denser linear system as the rows of the matrix corresponding to the Dirichlet degree of freedoms are filled with coefficients allowing the approximation of the field at the boundary condition location.

### 2.5.3 Interpolating Moving Least Square Method

The Interpolating Moving Least Square (IMLS) method is a variation of the Moving Least Square method that allows the approximated field to interpolate the solution. The method
has been presented in [82], and analyzed in a number of papers [83, 84]. Interpolation of the approximated field can be achieved by various means. One of which consists in choosing a near singular weight function. The weight function assigns to the reference node a very large weight compared to the other nodes of the support. This makes the system (38) nearly singular but allows interpolation of the field. In this paper, the following weight function is considered:

\[
    w(s) = \begin{cases} 
        e^{-s^2(s^n - \epsilon)^{-1}} & \text{if } s \leq 1 \\
        0 & \text{if } s > 1. 
    \end{cases} 
\]  

(45) 

Here the parameters \( n \) and \( \epsilon \) control the singularity of the function. Based on the analysis of Maisuradze et al. [84], the following parameters have been selected: \( n = 4 \) and \( \epsilon = 10^{-15} \).

3 Problems and Error Norms

3.1 Problems Considered

For our comparisons, we have selected two two-dimensional and one three-dimensional linear elastic problems. These are:

- A cylinder under internal pressure (2D—plane stress model);
- An L-shape domain in Mode I loading (2D—plane stress model);
- A sphere under internal pressure (3D).

An analytical solution is known for each of these problems. The 2D and 3D problems are, respectively, presented in Figs. 3 and 4. A Cartesian coordinate system has been used. Due to the symmetries of the problems in the Cartesian coordinate system, we have only considered 1/4th of the cylinder and 1/8th of the sphere.

For all the problems presented in this section, a regular node discretizations has been selected.

The stress solution in terms of \( \sigma_{11}, \sigma_{12}, \sigma_{22} \) for each problem is presented in Figs. 5, 6 and 7, respectively, for the pressurized cylinder, the L-shape and the pressurized sphere. The equivalent von Mises stress (noted \( \sigma_{VM} \)) is also presented.

It should be noted that the stress solution tends to infinity at the interior corner of the L-shape domain. In order to represent the solution in Fig. 6, we have truncated the stress results around the singularity and represented truncated values in the same color as the selected threshold values.

3.2 Error Estimation

To assess the influence of the considered parameters on the solution, and to compare the accuracy of the methods, a set of errors on the stress results has been calculated. For this
Fig. 5  Pressurized cylinder—stress solution

Fig. 6  L-shape plate in mode I loading—stress solution
purpose, a properly scaled $L_2$ error norm and a $L_\infty$ error norm have been selected. The $L_2$ error norm averages out the error over all the collocation nodes.

Considering a domain $\Omega$ discretized with $n$ collocation nodes, where $\sigma_{ij}(X_k)$ and $\sigma_{ij}^h(X_k)$, respectively, represent the exact and the approximated stress values at a node $X_k$, the $L_2$ error norm is calculated as follows:

$$L_2(\sigma_{ij}) = \sqrt{\frac{\sum_{k=1}^n (\sigma_{ij}^e(X_k) - \sigma_{ij}^h(X_k))^2}{n}}. \quad (46)$$

For the L-shape problem, the singular point has not been included in the $L_2$ error norm as the analytical solution diverges at this point. The $L_\infty$ error norm corresponds to the maximum absolute error observed over the considered domain.

$$L_\infty(\sigma_{ij}) = \max_{k \in \Omega} \left( |\sigma_{ij}^e(X_k) - \sigma_{ij}^h(X_k)| \right). \quad (47)$$

### 4 Parametric Study

#### 4.1 General

In order to better understand the GFD and the DC PSE methods, and to assess the impact of the various parameters on the solution, we performed a parametric study. The purpose of this study is to select a single set of parameters that can be applied to any problem without knowing a priori the solution type. The following items have been considered:

- The selected weight function;
- The selected correction function basis for the DC PSE method;
- The number of support nodes.

The 2D cylinder, the 2D L-shape and the 3D sphere models have been considered for this study. A regular distribution of 5,372 nodes and of 13,735 nodes has been selected for the 2D problems, respectively. A regular distribution of 83,174 nodes has been selected for the 3D problem. The DCPSE1 variation of the DC PSE method has been selected for the purpose of this sensitivity study. The sensitivity analyses for the weight function and the DC PSE correction function have been performed for the 2D model only. The impact of the number of support nodes on the error has been assessed for both, the 2D and 3D problems. Due to the symmetries of the models, results are only presented in terms of error on the $\sigma_{11}$ and $\sigma_{12}$ components of the stress tensor.

#### 4.2 Weight Function Sensitivity

##### 4.2.1 GFD

Various functions can be considered for the weight function introduced in Eqs. (5) and (12). The 3rd and 4th order splines are the preferred types [see Eqs. (26) and (27)]. In order to vary the shape of the weight functions, we have composed the splines with the following power function

$$W(s) = (w(s))^\gamma, \quad (48)$$

where $w$ is the spline function and $W$ is the modified weight function. We have compared the results obtained for power parameters $\gamma$ between 0.4 and 1.2. We have also considered a linear weight function for comparison purposes. The equation of this function is given by

$$w(s) = \begin{cases} 1 - s & \text{if } s \leq 1 \\ 0 & \text{if } s > 1. \end{cases} \quad (49)$$

The shapes of the considered functions are presented in Fig. 8.

The $L_2$ and $L_\infty$ errors obtained with the considered weight functions are presented in Figs. 9 and 10 for the 2D cylinder and for the 2D L-shape, respectively. The error for the linear weight function is also presented in these figures for comparison purposes, even though a power parameter is not used in this function.

We can see from Fig. 9 that within the range [0.6; 0.9] the power parameter has little impact on the error. In this range, the type of spline used does not significantly impact the error either. From Fig. 10 we can see that the power parameter has little impact on the error in terms of $L_2$ norm. A linear weight function leads to similar results as the spline weight functions is the range of power parameter [0.3;1.1]. In terms of $L_\infty$ error norm, the linear function leads to a lower error than the spline functions in the range of power parameter [0.4;1.2].

A 4th order spline with a power parameter of 0.75 appears to be a reasonable choice as it leads to a minimum error for both considered problem. It leads to a low error for the 2D cylinder without being too close to the rapid error increase that is observed when the power parameter decreases below 0.6. It also leads to a reasonably low error for the singular problem both in terms of $L_2$ and $L_\infty$ error norms. A unique set of parameters has been selected for both problems in order to be applied to a wide variety of problems in the domain of linear elasticity.

##### 4.2.2 DC PSE

In this Section, we assess the influence of the selected weight function on the error for the DC PSE method. The exponential weight function presented in Eq. (25) is compared to the
3rd and 4th order splines, of Eqs. (26) and (27) respectively. We have considered various combinations of shape parameters ($e$) and exponents ($\alpha$). In Fig. 11, the profile of three exponential weight functions, along with the two splines functions, is presented.
In Figs. 12 and 13, the error in terms of $L_2$ and $L_\infty$ norms for the $\sigma_{11}$ and $\sigma_{12}$ stress components is presented for various combinations of exponents and shape parameters. The error obtained with the 3rd and 4th order splines is also presented on the graphs for comparison purposes, even though the shape parameter does not apply for these functions.

The analysis of Figs. 12 and 13 shows that the exponential weight functions lead to smaller errors than the spline functions for shape parameters between 0.25 and 0.45 and for both problems. The combination of an exponent of 1.0 and a shape parameter of 0.25 leads to the smallest error for the 2D cylinder. A shape parameter of 0.30, associated to an exponent of 1.0, leads to similar results for this problem without being too close to a rapid increase in the observed error.

The analysis of the results for the 2D L-shape shows that an exponent of 2.0 associated with a shape parameter of 0.33 leads relatively constantly to the lowest error.

In order to be applied to most of the problems where the type of solution is a priori unknown, a single set of parameters is selected. This set of parameters is: a shape parameter of 0.30 and an exponent of 1.0. This combination leads to a more significant error reduction than the set of parameters leading to the minimum error for the 2D L-shape problem.

### 4.3 DC PSE Correction Function

The basis functions used to build the correction function in the DC PSE method can be selected from various function types. The most commons bases are the polynomial basis and the exponential basis. For the case of a two-dimensional problem, the polynomial basis is $P = [1 \ x \ y \ x^2 \ xy \ y^2]^T$, and the exponential basis: $P = [1 \ e^x \ e^y \ e^{2x} \ e^{x+y} \ e^{2y}]^T$. In order for these bases to be independent from the node density, the functions have been scaled according to the support radius. For a node $X_i$ in the support of the collocation node $X_c$, the scaling parameters are $S_x = \frac{X_i - x_c}{r_c}$ and $S_y = \frac{Y_i - y_c}{r_c}$, where $r_c$ is the support radius of the collocation node $X_c$. The polynomial basis becomes:

$$P = \begin{bmatrix} 1 & S_x & S_y & S_x^2 & S_xS_y & S_y^2 \end{bmatrix}^T$$

(50)
The errors obtained for each correction function basis are presented in Fig. 14 for the 2D cylinder and in Fig. 15 for the 2D L-shape.

It can be observed from Fig. 14 that the polynomial function basis constantly leads to a much lower error than the exponential basis. Depending on the number of nodes, the error increases by a factor between 5 and 30 when the exponential basis is used.

In Fig. 15 we can see that both correction function bases lead to similar results. In terms of \( L_2 \) error norm the exponential basis leads to a reduction of around 5% compared to the results where a polynomial basis is used.

Based on the results presented in this section, the polynomial basis is preferred as it leads to a much higher error reduction than the exponential basis for the considered problems. This basis function is expected to give a reasonably low error for most of the problems in the domain of linear elasticity.

### 4.4 Number of Support Nodes

**Support radius selection** In this work, we have selected the support radii of the collocation nodes based on the number

![Fig. 10 GFD Weight sensitivity—2D L-shape. \( L_2 \) (Left) and \( L_\infty \) (Right) errors. Comparison for 3rd and 4th order splines weight functions composed with a power function of various exponents. The linear weight function is also given for reference purposes. The 4th order spline leads to a low error than the 3rd order spline for both stress components and both error norms. The linear function leads to the lowest error in terms of \( L_\infty \) error norm.](image)

![Fig. 11 DC PSE weight functions: comparison of the profile of typical exponential and spline functions used as weight in the DC PSE approximation](image)
of nodes within the support they define. The number of nodes in the support of a collocation node shall be of at least the number of approximated derivatives. In practice, in order to account for the node distribution, a larger number of nodes is used.

**GFD** In this Section, we study the impact of the number of nodes in the support of a collocation node for the GFD method. Various combinations of inner node and boundary node support sizes have been considered. Results are presented in Figs. 16, 17 and 18, respectively, for the 2D cylinder, the 2D L-shape and the sphere under internal pressure. The $L_2$ and the $L_\infty$ norms are presented as a function of the number of support nodes for collocation nodes located on the boundary. The results are presented for three sizes of inner node supports for the 2D problems and four sizes for the 3D problem.
The results in terms of $L_2$ and $L_\infty$ errors present a similar trend for the $\sigma_{11}$ and $\sigma_{12}$ stress components. We can see that increasing the number of nodes in the inner nodes supports does not necessarily lead to an error reduction. The loss in terms of resolution is not compensated by the gain in solution smoothness. Increasing the number of support nodes for the boundary nodes steadily (and rapidly) reduces the error for the 2D cylinder problem. An error reduction of a
factor of approximately one hundred is observed when the number of support nodes for boundary collocation nodes is increased from 13 to 18.

The number of support nodes on the boundary does not affect much the observed error for the 2D L-shape problem as the model is loaded via Dirichlet boundary conditions.
The size of the inner nodes support has also little impact on the error for this problem.

A combination of 11 support nodes for interior nodes and 19 support nodes for boundary nodes is selected as it leads to a low error for both problems while maintaining the fill of the system matrix reasonably low.

It can be observed from Fig. 18 that, for the 3D sphere, a minimum error is obtained for 75 support nodes for boundary collocation nodes. Increasing the size of the support from 55 to 75 for boundary nodes reduces in average by a factor 10 the observed error both in terms of $L_2$ and $L_\infty$ norms. The number of support nodes for the collocation nodes located in the domain has a smaller impact on the error. 37 support nodes appears to be a reasonable choice as it leads to a low error while keeping the fill of the matrix reasonable.

The sparsity of the problem matrix is reduced when the number of support nodes increases. However, the impact of an increase in the number of support nodes on the boundary is limited as it only affects a fraction of the nodes of the domain.

DC PSE As for the GFD method, the impact of the support size on the observed error is presented in this section for the DC PSE method. The results are presented in Figs. 19, 20 and 21, respectively, for the 2D cylinder, the 2D L-shape and the sphere under internal pressure for various combinations of inner node and boundary node support sizes.

It can be observed from Fig. 19 that an inner node support composed of 13 nodes leads almost always to the minimum error. It can also be observed that an increasing number of nodes in the support of the boundary nodes reduces relatively steadily the error for the 2D cylinder. An error reduction of a factor two is observed when increasing the number of support nodes from 13 to 19 for most inner nodes support sizes.

It can be observed from Fig. 20 that the number of support nodes on the boundary has little effect on the observed error for the L-shape problem. This is because the model is loaded via Dirichlet boundary conditions. The number of support nodes for inner collocation nodes has little impact on the error.

Based the results from Figs. 19 and 20, 13 support nodes for inner collocation nodes and 19 support nodes for boundary collocation nodes is a reasonable choice for most problems as it leads to a minimum error.

It can be observed from Fig. 21 that, for the 3D sphere, a minimum error is observed for inner node supports composed of 37 nodes. The number of boundary support nodes has little impact on the error. Increasing the number of support nodes from 60 to 90 reduces by only 8% in average the observed error. 75 support nodes for boundary collocation nodes has been selected as for the GFD method in order to keep the fill of the system matrix as low as possible while maintaining the error low.
Fig. 19  DC PSE support node number sensitivity—2D cylinder. $L_2$ (Left) and $L_\infty$ (Right) errors for various combinations of inner nodes and boundary nodes support sizes. Inner collocation nodes with 13 support nodes lead relatively constantly to a low error. The error starts to increase for the 13 inner nodes case when the number of boundary nodes is larger than 19.

Fig. 20  DC PSE support node number sensitivity—2D L-shape. $L_2$ (Left) and $L_\infty$ (Right) errors for various combinations of inner nodes and boundary nodes support sizes. All the combinations of inner and boundary nodes support size lead to similar errors. This is due to the system being loaded via Dirichlet boundary conditions.
4.5 Results Summary

Based on the results presented in the above sections, the parameters that lead to a minimum error, while maintaining the computational expense reasonably low, are presented in Table 2. These parameters are expected to lead to a low error for a wide variety of linear elasticity problems, including singular problems. They have been used as a base case for the studies presented in the next sections of the paper.

5 Improvement Methods

In this section, we present three methods which are expected to improve the accuracy of the GFD and DC PSE methods.

| Parameter                  | GFD          | DC PSE       |
|----------------------------|--------------|--------------|
| Weight function type       | 4th Order spline | Exponential  |
| Weight function parameter  | $\gamma = 0.75$ | $\alpha = 1$, $\epsilon = 0.30$ |
| Correction function        | N/A          | Polynomial   |
| Size of inner nodes support (2D/3D) | 11/37        | 13/37        |
| Size of boundary nodes support (2D/3D) | 19/75        | 19/75        |

5.1 Use of a Voronoi Diagram in Collocation

5.1.1 General

A Voronoi diagram is a partition of a selected region over which nodes are distributed. A cell is associated to each node. The boundaries of the cell are defined so that all the points contained in it are closer to the cell reference node than to any other node of the domain. Figure 22 shows a 2D Voronoi diagram on the disc support of a collocation node. The cells associated to each node are delimited by gray lines, while the boundary of the support is drawn in blue.
5.1.2 Application to the GFD and DC PSE Methods

**GFD** The principle of the GFD method has been presented in Sect. 2.2. When more nodes than derivatives are present in the node support, a mean least square approximation is used to determine the field derivatives that best fit the distribution. The contribution of each node in the least square approximation is weighted by a function which only depends on the distance between the reference node and the support node. A Voronoi diagram can be used to determine an additional weight based on the spatial arrangement of the nodes. This weight is the area or volume $v$ of the considered Voronoi cell and is multiplied with the distance based weight $w$. Equation (12) becomes:

$$m_{ij} = \sum_{k=1}^{m} w(X_{pk} - X_{c})v(X_{pk})P_{ik}(X_{c})P_{jk}(X_{c}).$$  \hspace{1cm} (51)

**DC PSE** One of the key aspects of the DC PSE method presented in Sect. 2.3 is the convolution of the Taylor’s series expansion with a correction function $\eta$. The domain integral is transformed into a discrete summation with a volume $V_p$ associated to each particle $X_p$ of the support. In a first approximation, all $V_p$ values are set to unity. In order to improve the accuracy of the method, a Voronoi diagram can be used to set $V_p$ equal to the volume of the Voronoi cell associated to each node $X_p$.

5.1.3 Results

In this section, we present the results for the 2D cylinder and the 2D L-shape problems. The errors are compared between a model where Voronoi weights are used, and a model where these weights are not considered. Two types of node distributions are considered: a structured and a free node distribution. The structured node distribution is created using a constant angle and radius increment for the 2D cylinder. For the 2D L-shape, a grid-type arrangement is used. The free node distribution uses a Delaunay triangulation of the domain for both problems. The two types of node arrangements are presented in Fig. 23 for the 2D cylinder problem.

**GFD** The results obtained with the GFD method are presented in Figs. 24 and 25 respectively for the 2D cylinder and the 2D L-shape problems for the both node distributions. A slight error reduction can be observed for the $L_2$ and the $L_\infty$ error norms when Voronoi weights are used but this reduction is not observed for all node densities. For the 2D cylinder and the 2D L-shape, an error reduction of around 2% is observed when Voronoi based weights are used with a regular discretization of the domain. A more significant error reduction is observed for the 2D cylinder with a free discretization of the domain. The error reduction is of around 17%. For the 2D L-shape, an error increase of 3% is observed.

**Fig. 23** 2D cylinder node distribution—structured consisting of 1680 Nodes (Left) and free consisting of 1762 nodes (Right). The structured node distribution is based on constant angle and radius increments while the free node distribution uses a Delaunay triangulation of the domain.
Fig. 24  Impact of Voronoi based weights on the errors for the GFD method—2D cylinder. $L_2$ (Left) and $L_\infty$ (Right) errors for structured and free node distributions. A reduction in the error is only observed for the free node distribution.

Fig. 25  Impact of Voronoi based weights on the errors for the GFD method—2D L-shape. $L_2$ (Left) and $L_\infty$ (Right) errors for structured and free node distributions. A slight reduction in the error is observed for the structured node distribution.
when Voronoi weights are used with a free discretization of the domain.

DC PSE As for the GFD method, we now assess the impact of Voronoi based volumes for the two node distributions for the 2D cylinder and the 2D L-shape problems. The results are presented in Figs. 26 and 27.

It can be observed from Fig. 26 that, for the 2D cylinder, the use Voronoi based volumes leads to a large error increase for the structured node distribution. For the free node distribution, an average reduction of 10% is observed.

From Fig. 27 we can see that the trend for the 2D L-shape is the opposite than for the 2D cylinder. An error reduction...
of around 5% is observed when Voronoi based volumes are used with the structured node distribution. A slight error increase (less than 1%) is observed for the free node distribution when Voronoi based volumes are used.

Discussion From Figs. 24, 25, 26 and 27 we can see that the error reduction achieved by the use of Voronoi based volumes is not guaranteed. The use of such volumes for the 2D cylinder problem lead to an error reduction of up to 17% for the 2D cylinder problem. However, for the 2D L-shape, an error increase of 3% has been observed for the free node distribution. From this study, we can conclude that the use of the Voronoi based weights shall be used with care as is may lead to a significant error increase.

5.2 Collocation Method Stabilization

5.2.1 General

Within the framework of collocation, the boundary conditions are applied at the nodes using the strong form of the partial differential equations. This may lead to ill conditioning of the linear system of equations, as both the boundary conditions and the equilibrium equation cannot be enforced simultaneously at a boundary node. To overcome this issue, a stabilization method, known as the Finite Increment Calculus (FIC), has been presented by Oñate [87] for structural problems that are solved with the Finite Point Method. The method is presented in this section. Results for the 2D cylinder and for the 2D L-shape problems are presented with and without stabilization. This stabilization approach is used for both the methods, GFD and DC PSE.

5.2.2 Stabilized Equations

Considering an unknown field \( f \), a partial differential problem is defined by a differential operator \( A \) applied to the interior domain \( \Omega \), a field \( \vec{f} \) set to the boundary \( \Gamma_u \), and a differential operator \( B \) applied to the boundary \( \Gamma_t \) (see Fig. 28).

\[
A(f) = 0 \quad \text{in} \quad \Omega, \\
f - \vec{f} = 0 \quad \text{on} \quad \Gamma_u, \\
B(f) = 0 \quad \text{on} \quad \Gamma_t. 
\]

Based on [87, 88], the stabilized system of equations is:

\[
A(f) - \frac{1}{2} \sum_{j=1}^{m} h_j \frac{\partial A(f)}{\partial x_j} = 0 \quad \text{in} \quad \Omega, \\
f - \vec{f} = 0 \quad \text{on} \quad \Gamma_u, \\
B(f) - \sum_{j=1}^{m} h_j n_j A(f) = 0 \quad \text{on} \quad \Gamma_t, 
\]

where \( m \) is the dimension of the domain, \( h_j \) is the characteristic length of the domain in the direction \( j \), and \( n \) is the unit normal.

For isotropic support weight functions, \( h_j \) reduces to \( h \) and can be expressed as follows:

\[
h = R_{\text{Sup}} \left( \frac{\pi}{N_{\text{Sup}}} \right)^{\frac{1}{2}} \quad \text{for 2D problems}, \\
h = R_{\text{Sup}} \left( \frac{4\pi}{3N_{\text{Sup}}} \right)^{\frac{1}{3}} \quad \text{for 3D problems}. 
\]

where \( R_{\text{Sup}} \) and \( N_{\text{Sup}} \), respectively, represent the radius of the node support, and the number of nodes in the support.

5.2.3 Results

Equation (53) has only been applied to the boundary nodes where the maximum error is usually observed. Also, the stabilized equation on the boundary does not require the approximation of an additional derivative order. The results obtained are presented from Figs. 29, 30, 31, and 32. \( L_\infty \) error results have not been presented in this section as this error highly depends on the proximity of the closest node to the singularity.

It can be observed from the results presented from Figs. 29 and 30 that the stabilization equations lead to an error increase for the 2D cylinder problem. Using the stabilization method increases by a factor 30 the error for the GFD method and by a factor 20 for the DC PSE method. We can see from Figs. 31 and 32 that the error is reduced by the use of the stabilization method for the L-shape problem. An average error reduction of 25% is observed for the GFD method and of 35% for the DC PSE method.

For the 2D cylinder, the loading is applied via Neumann boundary conditions, which represent the pressure loading. The L-shape problem on the other hand is loaded using Dirichlet boundary conditions. It can be concluded from this study that stabilization of the Neumann boundary conditions does not necessarily lead to a reduction of the observed error. Thus, stabilization of Neumann loaded problems does not seem to be an effective solution for the considered methods.

5.3 Support Node Selection for Singular Problems

5.3.1 General

For singular problems, such as the L-shape problem presented in Sect. 3.1, the selection of the support nodes in the vicinity of the singularity impacts the solution.
In 1994, with the Element-Free Galerkin (EFG) method, Belytschko et al. [5] introduced the visibility criterion for support nodes selection. This criterion has been widely used in the context of EFG fracture mechanics, see e.g., Duflot [89]. The support of a collocation node $X_c$ in the domain $\Omega$ is selected so that any point $X_p$ of the support, defined by a radius $R_{sup}$, can be connected to the collocation node by a segment which does not intersect the domain boundary $\Gamma$. The “hidden” zone is the zone within the support radius for which the segment between the collocation node and the support node intersects the boundary.

The diffraction criterion, introduced by Organ in 1996 [90], is based on the same principle as the visibility criterion. The nodes in the zone “hidden” from the collocation node are only included in the support if the sum of the length between the support node $X_p$ and the singularity $X_s$ and the length between the singularity and the collocation node $X_c$ is smaller than the support radius $R_{sup}$ (see Fig. 34). The weights associated to the nodes in this zone are based on this increased distance to the collocation node.

### 5.3.2 Results

In this section, we assess the impact of the support node selection criterion on the $L_2$ error for the L-shape problem. Results obtained with the visibility criterion and with the diffraction criterion are compared to results where no criterion is considered (i.e. every node within the support radius of a collocation node is included in the support).

Both the GFD and DC PSE methods are considered in this section. The results for the GFD and DC PSE methods are presented in Figs. 35 and 36, respectively.

It can be observed that, for both methods, the use of the visibility criterion leads to a significant error reduction compared to results where no criterion is applied. For the GFD method, the error reduction ranges from 50 to 20%
Fig. 30 Stabilization results comparison—2D cylinder—DC PSE method. $L_2$ error for stabilized and non-stabilized PDE for increasing node numbers. A lower error is observed for the non-stabilized PDE.

Fig. 31 Stabilization results comparison—2D L-shape—GFD method. $L_2$ error for stabilized and non-stabilized PDE for increasing node numbers. A lower error is observed for the stabilized PDE.

Fig. 32 Stabilization results comparison—2D L-shape—DC PSE method. $L_2$ error for stabilized and non-stabilized PDE for increasing node numbers. A lower error is observed for the stabilized PDE.
depending on the node density. For the DC PSE method, the error reduction ranges from 55 to 25%. These results are expected as the singularity of the domain is more accurately represented with the visibility criterion. The diffraction criterion leads to an error increase ranging from a factor 2 to factor 10 for the GFD and the DC PSE methods. For both methods, the convergence rate is larger with the diffraction criterion. The visibility criterion appears to be a sensible choice for convex and singular problems.

6 Benchmarking

In Sect. 4, we presented the studies of the parameters influencing the GFD and DC PSE methods and have selected a set of optimum parameters presented in Table 2. Based on these parameters, we compare in this section first the three variations of the DC PSE method presented in Sect. 2.3.4, and then the methods presented in Sect. 2. The results are assessed in terms of the $L_2$ and $L_\infty$ error norms for the 2D cylinder and the 2D L-shape problems. None of the improvement methods presented in Sect. 5 have been used to derive the results presented in this section.

6.1 DC PSE Variations Comparison

The three variations of the DC PSE method are studied in this section. The results in terms of $L_2$ error are compared in Fig. 37 and in Fig. 38, respectively, for the 2D cylinder and the 2D L-shape problems.

It can be observed from Figs. 37 and 38 that all three methods lead to very similar results. In order to quantify the difference, the relative difference compared to the DCPSE2 is presented in Figs. 39 and 40 for the DCPSE0 and DCPSE1 methods.

6.2 GFD, DC PSE and Other Methods Comparison

6.2.1 General

In this section, the GFD and the DC PSE methods are compared to the MLS, IMLS and RBF-FD collocation methods. The same number of support nodes has been chosen for all methods. A 3rd order spline weight function has been chosen for the MLS method. The weight function presented in Eq. (45) has been considered for the IMLS method, and a Gaussian radial basis function has been selected for the RBF-FD methods. The GFD and the DC PSE methods are based on the parameters presented in Table 2. For reference purpose, results for the finite element method (FEM), obtained using the commercial software ABAQUS [70], are also included in the comparison.
The results from FEM are extrapolated to the nodes. This allows the results from the FEM to be compared with the same error norms to the results obtained with the collocation methods. The same discretization as for the collocation model has been selected for the FE models. The adjacent nodes of the regular distribution have been grouped into bilinear quadrilateral elements with four integration points.

6.2.2 Results for the 2D Cylinder Under Internal Pressure

From Fig. 41, it can be observed that the DCPSE1 method leads to the lowest error in terms of $L_2$ and $L_\infty$ norms for the $\sigma_{11}$ and $\sigma_{12}$ stress components. The GFD method leads to a slightly higher error than the DC PSE method. The MLS and IMLS methods lead to very similar results. The IMLS method lead to an error constantly lower than the MLS method. Finally, the FEM and the RBF-FD method lead to the largest errors. The error obtained with the RBF-FD method does not monotonically decrease as the node density increases.

6.2.3 Results for the L-shape Problem

The different methods are also compared for the 2D L-shape problem. Only the $L_2$ error is presented for this problem as the $L_\infty$ error diverges to infinity as the distance to the singular node tends to zero. The results are presented in Fig. 42.

We can see in Fig. 42 that the results obtained with the FEM are the closest to the analytical solution. The MLS, IMLS, GFD and DC PSE methods lead to similar results. The trend is however opposite to the results presented in Fig. 41. It shows hows the methods are affected by a rapid change in the field solution. The MLS method is the method leading to the lowest error among the collocation methods. Finally, the RBF-FD is the method leading to the highest error.
Fig. 37 DC PSE method variations comparison—2D cylinder. $L_2$ error as a function of the number of nodes in the model for the DCPSE0, DCPSE1 and DCPSE2 variations of the DC PSE method. No distinction can be observed between the different methods.

Fig. 38 DC PSE method variations comparison—2D L-shape. $L_2$ error as a function of the number of nodes in the model for the DCPSE0, DCPSE1 and DCPSE2 variations of the DC PSE method. No distinction can be observed between the different methods.

Fig. 39 DC PSE method variations comparison—2D cylinder. Relative difference to the DCPSE2 $L_2$ error for the DCPSE0 and DCPSE1 methods. The DCPSE2 methods leads to the lowest error followed by the DCPSE1 method.
6.2.4 Convergence Rate and Computational Expense

The methods are also compared in terms of convergence rate and solution time. Results are summarized in Table 3. It can be observed that the RBF-FD method has the lowest convergence rate for the 2D cylinder problem, and the largest convergence rate for the L-shape problem. The IMLS method shows the largest convergence rate for the 2D cylinder problem. The GFD method is the one having the lowest convergence rate for the L-shape problem.

The right column of Table 3 shows that the computation time for the MLS and IMLS method is significantly larger than the computation time for the other methods. This is due to the assembly step which is more time consuming for these methods, as the system presented in Eq. (38) needs to be solved for each derivative.

The computation times presented in Table 3 can be split into four main steps. These steps are:

- Problem initialization;
- Matrix assembly;
- Solution of the linear problem;
- Postprocessing and results output.

The initialization step consists in loading the problem from the input file and searching for the node neighbors (the nodes to be included in the support of each collocation nodes). During the assembly step, all the derivatives of the unknown field are approximated as a function of the field, and the linear problem is assembled. For the solution of the linear problem, while direct solvers can be used for 2D problems of a reasonable size, iterative solvers should be used for large 3D problems, as the matrix of the linear problem is significantly denser than for 2D problems. Finally, the postprocessing step consists of the computation of the quantities of interest (stress components based on the displacement field). In this work, the linear solver MUMPS [91, 92] and the iterative solver PETSc KSP [93, 94] have been used. The analyses have been run using a C++ code developed in-house. The code was run on a machine equipped with an Intel Xeon E5-1650 processor at 3.2 GHz. A single process and a single thread has been used for the analyses.

For various node densities, the fraction of the total analysis time of the matrix assembly step and of the solution step is presented in Figs. 43 and 44 for the L-shape problem. The duration of the problem initialization and of the postprocessing steps is negligible compare to the two other steps. These steps have not been presented in those figures. The total analysis duration has also been presented in these figures on a secondary axis. The results are presented for the GFD and for the IMLS methods as these methods are, respectively, the fastest and the slowest methods for the node density presented in Table 3.

It can be observed from Figs. 43 and 44 that for both methods the trend of the results is similar. The assembly step represents the largest proportion of the total analysis time for 2D problems of small dimensions. As the number of degrees of freedom increases, the fraction of the solution step (here with a direct solver) in the analysis time increases and becomes larger than the fraction of the assembly step. This result is expected as the assembly time increases linearly with the number of nodes while the solution time increases exponentially with the number of degrees of freedom. The increased computation effort required by the IMLS method during the assembly step is observed on this graph as the fraction of the analysis spent assembling the matrix is larger than for the GFD method.
Fig. 41  Methods comparison—2D cylinder. $L_2$ (Left) and $L_\infty$ (Right) errors as a function of the number of nodes for various collocation methods (i.e. GFD, DCPSE1, MLS, IMLS and RBF-FD) and for the FEM. The lowest error is obtained with the DCPSE1 method.

Fig. 42  Methods comparison—2D L-shape. $L_2$ errors as a function of the number of nodes for various collocation methods (i.e. GFD, DCPSE1, MLS, IMLS and RBF-FD) and for the FEM. The lowest error is obtained with the FEM. The collocation method leading to the lowest error is the MLS method.
Three Dimensional Problems

The parametric study presented earlier allowed the selection of “optimal” weight functions and support sizes for solving problems from different fields of application. In this section, we present the results from the stress analysis of various three-dimensional problems. The results in terms of von Mises stress obtained with the GFD method are compared to the results obtained with ABAQUS. For a consistent comparison, the same discretization has been used for both methods. The results obtained from the FEA are extrapolated to the nodes. This tends to overestimate the error for the FEA method as the stresses are less accurate at the nodes than at the integration points. This allows, however, a comparison, at each node, of the results derived with the FE method to the results derived with the GFD method. In order to obtain convergence of the 3D problems with the collocation method, a relatively large number of nodes is used. Such a high density is required to capture the details of the geometry.

Flange model The first problem considered in this section is an ISO flange. Only a quarter of the flange has been modeled due to the symmetries of the domain. The model and the various surfaces, on which boundary conditions are applied, are presented in Fig. 45. Two load cases have been considered for this model: an internal pressure loading and an axial traction imposed by the connected pipe. The boundary conditions associated to each load case are presented in Table 4.

Figures 46 and 47 show the von Mises stresses obtained with the GFD and the FE methods, respectively, for the internal pressure and traction load cases.

The results obtained from the GFD method and from the FEM are very close. In order to visually assess the difference between the two solutions, the difference between the von Mises stress results obtained from the GFD model and from the FE model are presented in Fig. 48 for both load cases.

It can be observed from Fig. 48a that the stresses on the inner surface of the flange are larger for the GFD method. From Fig. 48b, it can be observed that the stresses obtained in the neck and in the bottom of the conical section are larger for the GFD method.

Blade model Fig. 49 presents a simplified model of a high pressure blade. The surfaces, on which the boundary conditions are applied, are presented in this figure. The nodes in the planes YZ, XZ and XY are, respectively, fixed in the X, Y and Z directions. A constant pressure resulting from a gas flow is applied on the pressurized surface. The remaining surfaces of the blade are considered stress free.

Figure 50 shows the von Mises stress results for the GFD and FE methods. As for the flange problem, the difference between the two von Mises stress solutions is presented in Fig. 51.

If can be observed from Figs. 50 and 51 that the stress concentration in the zone between the blade and the support is larger for the GFD model.

Horseshoe model In 2005, the horseshoe model was solved by Hughes et al. [33] using the IGA method. This model has been reproduced and is presented in Fig. 52. The nodes of the top left plane are fixed in the X and Z
The nodes of the top left plane and of the top right plane are, respectively, subjected to a positive and a negative displacement applied in the Y direction. These displacements are equal in absolute value. The external surfaces of the horseshoe are considered stress free.

Figure 53 shows the von Mises stress results for the GFD and FE models. The two figures on the left show two different views of the solution of the problem solved with the GFD method. The two figures on the right show the solution of the problem solved with the FEM. It can be observed from Fig. 53 that the stress concentration in the bottom of
the horseshoe is larger in the GFD solution. A higher stress concentration is also observed at the edges of the top left plane in the GFD solution.

It should be noted that the computation time was lower for the FE method than for the GFD method. This is due to the loading, which creates a singularity at the edges of the top left plane and of the top right plane. The impact of this singularity affects more the GFD model as it is solved by collocation (strong form of the PDE).

The difference between the two von Mises stress solutions is presented in Fig. 54. This figure confirms that higher stress concentrations are observed for the GFD model at the top planes edges and in the bottom section of the horseshoe.

**Fichera’s corner model** The Fichera’s corner model analyzed in [95–97] is presented in Fig. 55. The characteristic
planes, on which boundary conditions have been applied, are highlighted and labeled in this figure. The nodes in the planes YZ, XZ and XY are, respectively, fixed in the X, Y and Z directions. A uniform traction is applied on the front face of the truncated cube. The remaining surfaces are considered stress free. When solved using the GFD method, the internal corner nodes have not been included in the Fichera’s corner model. These nodes lead to the divergence of the solution as the stress is infinite at these locations. The visibility criterion presented in Sect. 5.3.1 is applied to this problem.

Figure 56 shows the von Mises stress results for the GFD and FE methods. As for the three other problems, the difference between the two von Mises stress solutions is presented in Fig. 57.

It can be observed from Figs. 56 and 57 that the stress concentration near the internal edges is larger for the FE model than for the GFD model. The GFD model leads to larger results only in the center of the corner. In order to visualize more precisely the results of this analysis, von Mises stress results are plotted in Fig. 58 along the axes D, D’ and D” presented in Fig. 56a. The axes D and D’ follow the edge of the model, while the axis D” is slightly offset from the internal corner as results are not available at the corner nodes for the GFD method. The truncated cube has an edge length of 4. The coordinate of the internal corner is (0,0,0).

Figure 58 shows that, depending on the considered axis, either the GFD method or the FE method leads to the largest results. For this model, the FE method is expected to give a higher stress concentration as the corner nodes have not
been included in the GFD model but have been included in the FE model.

Discussion From the figures presented in this section for the various problems considered, it can be observed that the results obtained with the GFD method are very close to the results obtained with the FEM. The stress concentrations are slightly larger for the flange, the blade and the horseshoe problems when the GFD method is used. This might be due to the use of the strong form of the equations, which allows solving the loading equations on the boundary of the domain. In the FEM, the problem is solved in a weak form using an integration over the domain. For the Fichera's corner problem, the largest von Mises stress concentration is observed for the FE method. This is due to the internal corner nodes which have not been included in the GFD method. The FEM is thus deemed more accurate.

8 Conclusions

The aim of the paper was three-fold:

Brief review and primer We briefly reviewed Taylor-series expansion based collocation/meshfree methods.
Our aim here was not to be exhaustive, but to cover the main material available, to our knowledge. We also presented detailed derivation of the system matrices, in order to facilitate the entry of newcomers into the field and attempted to unify the generalised finite difference method and the discrete correction particle strength exchange method under one umbrella.

**Performance benchmarking** We provided a detailed benchmarking strategy for Taylor-series expansion based collocation methods as well as all data files including all input files, boundary conditions, point distribution and solution fields, so as to facilitate future benchmarking of new methods.

**New methods for non-smooth solutions** We proposed a few improvements to the original methods, both DCPSE and GFD, in order to treat problems with non-smooth solutions, including discontinuities, singularities or sharp gradients.

---

**Fig. 53** Horseshoe under shear loading—von Mises stress results from GFD method (a, c) and FEM (b, d) (521,326 nodes). The stress concentration in the inner surface of the horeshoe is slightly larger for the GFD method.
We noted that the various parameters involved in the methods have a significant impact on the solution, and that they should therefore be carefully chosen. In itself, this is a drawback compared to more parameter-robust methods, in particular the finite element method. Another main conclusion of this work is that common approaches used in practice to improve collocation methods must be used with caution as they do not always lead to the reduction of the overall error. We observed the following:

1. For the GFD method, the weight function based on the 4th order spline leads to the minimum error for problems with a polynomial solution such as the pressurized cylinder. For singular problems, such as the L-shape in mode I loading, both linear and 4th order spline weight functions lead to a minimum error.

2. For the DC PSE method, the weight function based on the exponential functions leads to the minimum error for both polynomial and singular problems.

3. For the problems with a polynomial solution, a polynomial correction function basis leads to an error approximately fifteen times lower than with an exponential basis function. For the singular problems, an exponential correction function basis leads to an error approximately 5% lower than with an exponential basis function. A polynomial correction function is recommended for most problems as the solution type is not known a priori.

4. Increased size of the node supports on the boundary helps decreasing the overall error, while increasing only slightly the number of non-zero elements in the system matrix. For the polynomial problem considered, a reduction of a factor one hundred is observed between boundary support nodes composed of thirteen nodes.
and eighteen nodes. For the singular problem, no significant error reduction is observed.

(5) Voronoi diagrams can be used to give additional information to the collocation methods on the spatial arrangement of the nodes over the domain.

(a) For the GFD method, Voronoi diagrams allow the selection of weights which depend on the node placement over the collocation node support.

(b) For the DC PSE method, Voronoi diagrams are expected to improve the accuracy of the convolution, but they may also lead to an increased error for some node distributions.

The use of Voronoi diagrams helps in reducing the error for the 2D cylinder problem with a free node distribution (based on a Delaunay triangulation). A reduction of up to 17% is observed for the GFD method and of up to 10% for the DC PSE methods. For a regular node distribution an error increase of 3% is observed for the considered problems when Voronoi diagrams are used. For the L-shape problem, the use of Voronoi

---

Fig. 56 Fichera’s corner subjected to a uniform traction on the front side—von Mises stress results from the GFD method (a) and FEM (b) (264,726 nodes)

Fig. 57 Fichera’s corner subjected to a uniform traction on the front side—Difference between von Mises stress results obtained from the GFD method and FEM (264,726 nodes). The stress concentration in the internal corners is larger for the FE model.
diagrams has no significant impact on the error. It can be concluded that the use of Voronoi diagrams does not allow a significant error reduction for the considered node arrangements, and their use is not recommended in the general case.

(6) The stabilization method reduces the error for the L-shape problem by respectively 25% and 35% for the GFD and DC PSE methods. A large error increase (up to a factor 30) is observed for both methods for the pressurized cylinder problem. This difference is due to the type of boundary conditions imposed. The stabilization method is more suitable to Dirichlet loaded problems than to Neumann loaded problems.

(7) For problems with singularities and concave geometries, the visibility criterion improves the convergence of the solution when solved with iterative solvers. Moreover, it allows to significantly reduce the observed error. A reduction of up to 50% and 55% are respectively observed for the GFD and DC PSE methods when the visibility criterion is used. The use of this criterion for support node selection is recommended for all singular and concave problems.

(8) Compared to other typical collocation methods (e.g., MLS, IMLS, RBF-FD), the GFD and DC PSE methods have shown good performance both in terms of observed error and computation time. The results obtained with the collocation methods are close to
the results obtained using the FEM and more accurate for some problems. For large 3D problems, the GFD method leads to very similar results as those obtained using FEM.

(9) A slightly larger stress concentration has been observed for the flange, the blade and the horse-shoe problems when solved with the GFD method compared to the results from FEA. For the Fichera’s corner problem, the stress concentration obtained with the FEM is slightly more pronounced than that obtained using the GFD method. This is due to the fact that the corner nodes have not been included in the GFD method, and thereby, the FEM results represent more accurately the actual solution.

To summarize, we have proposed for the GFD and DC PSE methods a set of optimal parameters that can be used to readily solve any linear elastic problem. We also showed that point collocation methods may be used effectively for problems with singularities and 3D problems of industrial size. Using the visibility criterion for concave and singular problems improves the convergence of the methods and leads to a significant error reduction. A logical next step in collocation methods is to investigate the use of enriched weight functions and enriched stencils near singularities in order to improve the results obtained in regions of rapid field change. Similarly, a posteriori error estimation driven local refinement, vastly simplified in collocation methods, should be investigated, which is the topic of ongoing work in our teams. Finally, a massively parallel approach, if possible based on graphical processing units, should be investigated to accelerate the solution scheme.

**Acknowledgements** Stéphane P.A. Bordas and Satyendra Tomar thank partial funding for their time provided by the European Research Council Starting Independent Research Grant (ERC Stg Grant Agreement No. 279578) RealTCut “Towards real time multiscale simulation of cutting in non-linear materials with applications to surgical simulation and computer guided surgery”. The authors are also grateful for the funding from the Luxembourg National Research Fund (INTER/FWO/15/10318764).

**Appendix: GFD and DC PSE Methods Comparison for 1D Problems**

The purpose of this appendix is to present a detailed exposition of the GFD and the DC PSE methods for a simple 1D problem. We have selected the case of a second order PDE for illustration purposes.

Considering a differential operator $A$ and a field $f : \mathbb{R} \to \mathbb{R}$, the following PDE can be written over the domain $\Omega$:

$$A(f) = 0 \quad \text{in} \quad \Omega.$$  

(55)

The field $f$ shall also verify the conditions imposed on the Dirichlet and Neumann boundaries, which are denoted by $I_\alpha$ and $I_\gamma$, respectively (see Fig. 59). The field values are set to $\tilde{f}$ on $I_\alpha$. On $I_\gamma$, the field shall verify a lower order PDE defined by an operator $B$.

$$f - \tilde{f} = 0 \quad \text{on} \quad I_\alpha,$$

$$B(f) = 0 \quad \text{on} \quad I_\gamma.$$  

(56)

In order to solve this problem by collocation, we need to transform it into a linear system of the form $A\mathbf{F} = \mathbf{B}$, where $A$ is the problem matrix, $\mathbf{F}$ is a vector containing the field values at each node of the domain, and $\mathbf{B}$ is a vector containing various constraints of the problem. The field derivatives at the collocation centers need to be approximated as a function of the values at the nodes. Collocation is typically performed at the nodes but can also be performed in other locations.

**Principle** Both the GFD and the DC PSE methods are based on a Taylor’s series expansion of the unknown field around the collocation node.

---

**GFD**

The field derivatives at the collocation node are determined so that the field values at the support nodes can be reproduced using a Taylor’s series expansion.

The field derivatives are approximated simultaneously at each node of the domain.

**DC PSE**

A convolution function is selected so that the approximated derivative in the Taylor’s series expansion only depends on the field values at the support nodes. All the other unknown derivatives are canceled by the selected function.

Different convolution functions are used to approximate the derivatives of various orders at a collocation node.
The steps associated to each method are presented below for the case of a second order PDE. In the sections below, the nodes are labeled $X_i$ and the 1D coordinate associated to the node is written $x_i$.

**Step 1: Taylor’s series approximation** For both methods, the first step consists in writing an approximation of the Taylor’s series expansion up to the desired order. The approximation order shall be of at least the highest derivative order of the differential operator $A$. In 1D, the Taylor’s series expansion at a point $X_{pi}$ in the vicinity of $X_c$ is:

$$f(X_{pi}) = f(X_c) + \sum_{i=1}^{+\infty} \frac{(x_{pi} - x_c)^i}{i!} \frac{df(X_c)}{dx^i}. \tag{57}$$

Denoting the second order approximation of this expansion by $f_{b}(X_{pi})$, Eq. (57) becomes:

$$f_{b}(X_{pi}) = f(X_c) + (x_{pi} - x_c) \frac{df(X_c)}{dx} + \frac{(x_{pi} - x_c)^2}{2!} \frac{d^2f(X_c)}{dx^2}. \tag{58}$$

**Step 2: Support node selection** The nodes in the vicinity of the collocation nodes are used to approximate the desired derivatives. These nodes are selected within a distance or radius $R_c$ to the collocation nodes $X_c$.

$$\begin{cases} 
\sum_{i=1}^{3} w(X_{pi} - X_c)(x_{pi} - x_c) \left[ f(X_c) - f(X_{pi}) + (x_{pi} - x_c) \frac{df(X_c)}{dx} + \frac{(x_{pi} - x_c)^2}{2!} \frac{d^2f(X_c)}{dx^2} \right] = 0 \\
\sum_{i=1}^{3} w(X_{pi} - X_c) \frac{(x_{pi} - x_c)^2}{2!} \left[ f(X_c) - f(X_{pi}) + (x_{pi} - x_c) \frac{df(X_c)}{dx} + \frac{(x_{pi} - x_c)^2}{2!} \frac{d^2f(X_c)}{dx^2} \right] = 0
\end{cases} \tag{62}$$

**Step 3: derivatives approximation**

**GFD**

The Taylor’s series expansion presented in Eq. (58) can be written for each node of the collocation node support. For the example presented in Fig. 60, three nodes are present in the support of the collocation node. The following system is obtained:

$$\begin{align*}
    f_{b}(X_{p1}) &= f(X_c) + (x_{p1} - x_c) \frac{df(X_c)}{dx} + \frac{(x_{p1} - x_c)^2}{2!} \frac{d^2f(X_c)}{dx^2} \\
    f_{b}(X_{p2}) &= f(X_c) + (x_{p2} - x_c) \frac{df(X_c)}{dx} + \frac{(x_{p2} - x_c)^2}{2!} \frac{d^2f(X_c)}{dx^2} \\
    f_{b}(X_{p3}) &= f(X_c) + (x_{p3} - x_c) \frac{df(X_c)}{dx} + \frac{(x_{p3} - x_c)^2}{2!} \frac{d^2f(X_c)}{dx^2}.
\end{align*} \tag{59}$$

If the number of support nodes is larger than the approximated derivative order (two for the selected example), the system is overdetermined. In that case, the derivatives that best reproduce the field values are determined using the mean least square method. The associated minimization problem is written in the form of a functional $B$. A weight function $w$ is used to balance the contribution of the support nodes as a function of their distance to the collocation node.

$$B(X_c) = \sum_{i=1}^{3} w(X_{pi} - X_c) \left[ f(X_c) - f(X_{pi}) + (x_{pi} - x_c) \frac{df(X_c)}{dx} \right. \\
\left. + \frac{(x_{pi} - x_c)^2}{2!} \frac{d^2f(X_c)}{dx^2} \right]^2. \tag{60}$$

The derivatives $\mathbf{Df}(X) = \left[ \frac{df(X)}{dx}, \frac{d^2f(X)}{dx^2} \right]^T$, that best approximate the known field values based on the Taylor’s series expansion, minimize $B$ when:

$$\frac{dR(X)}{d\mathbf{Df}(X)} \big|_{X=X_c} = 0. \tag{61}$$

This system can be rearranged in a matrix form as follows:

$$\begin{bmatrix} m_{11} & m_{12} \\
 m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} \frac{df(X_c)}{dx} \\
 \frac{d^2f(X_c)}{dx^2} \end{bmatrix} = \begin{bmatrix} -m_{01} & m_{01.1} & m_{01.2} & m_{01.3} \\
 -m_{02} & m_{02.1} & m_{02.2} & m_{02.3} \end{bmatrix} \begin{bmatrix} f(X_{p1}) \\
 f(X_{p2}) \\
 f(X_{p3}) \end{bmatrix}. \tag{62}$$
where the moments $m_{ij,k}$ and $m_i$ and the matrix $\mathbf{P}(X_c) \in \mathbb{R}^{3 \times 3}$ correspond to:

$$m_{ij,k} = w(X_{pi} - X_j)p_i^+\delta_{ik}(X_c)p_{i+1}\delta_{ik}(X_c),$$

$$m_i = \sum_{k=1}^3 m_{ij,k},$$

$$\mathbf{P}(X_c) = \begin{bmatrix}
1 & 1 & 1 \\
\frac{(x_{p1} - x_c)}{2!} & \frac{(x_{p2} - x_c)}{2!} & \frac{(x_{p3} - x_c)}{2!}
\end{bmatrix}.$$ (64)

The Eq. (63) can be represented in the form

$$\mathbf{A}(X_c)\mathbf{Df}(X_c) = \mathbf{E}(X_c)\mathbf{F}(X_c).$$

### DC PSE

The Taylor’s series expansion presented in Eq. (58) can be convoluted by a function $\eta$ over the support $\Omega_c$ of the collocation node $X_c$:

$$\int_{\Omega_c} f_h(X_p)\eta(X_p - X_c)dX_p$$

$$= \int_{\Omega_c} f(X_c)\eta(X_p - X_c)dX_p$$

$$+ \int_{\Omega_c} \frac{df(X_c)}{dx}(x_p - x_c)\eta(X_p - X_c)dX_p$$

$$+ \int_{\Omega_c} \frac{d^2f(X_c)}{dx^2}(x_p - x_c)^2\eta(X_p - X_c)dX_p.$$ (65)

The integral can be approximated by a sum, assuming that the nodes are regularly distributed over the support and that the field $f$ is sufficiently smooth.

$$\sum_{i=1}^3 f_h(X_{pi})\eta(X_{pi} - X_c) = f(X_c)M_0(X_c) + \frac{df(X_c)}{dx}M_1(X_c)$$

$$+ \frac{d^2f(X_c)}{dx^2}M_2(X_c).$$ (66)

where the moments $M_i$ are:

$$M_i(X_c) = \sum_{i=1}^3 \frac{(x_{pi} - x_c)^i}{i!}\eta(X_{pi} - X_c).$$ (67)

The convolution function is chosen so that all the moments in Eq. (66) are null except the one multiplying the approximated derivative of order $n_x$, which is denoted by $D^n f(X_c)$.

In order to satisfy this moment condition, the convolution function is chosen as the product of two functions: a correction function $K$ and a weight function $w$. The correction function is typically the product of a coefficient vector $\mathbf{a}$ and a polynomial basis $\mathbf{P}$. For the 1D second order approximation, the polynomial basis $\mathbf{P} = [1, x, x^2]^T$ can be selected. The correction function can then be written as follows:

$$\eta(X_p - X_c) = \mathbf{P}(X_p - X_c)^T \mathbf{a} w(X_p - X_c).$$ (69)

The coefficients of the vector $\mathbf{a}$ are determined in order to satisfy the moment condition set in Eq. (68). For instance, the moment condition associated to the second order derivative approximation is:

$$M_0(X_c) = 0 \iff \sum_{i=1}^3 \mathbf{P}(X_{pi} - X_c)^T \mathbf{a} w(X_{pi} - X_c) = 0$$

$$M_1(X_c) = 0 \iff \sum_{i=1}^3 (x_{pi} - x_c)\mathbf{P}(X_{pi} - X_c)^T \mathbf{a} w(X_{pi} - X_c) = 0$$

$$M_2(X_c) = 1 \iff \sum_{i=1}^3 \frac{(x_{pi} - x_c)^2}{2!} \mathbf{P}(X_{pi} - X_c)^T \mathbf{a} w(X_{pi} - X_c) = 1.$$ (70)

The system of equations can be put in a matrix form as follows:

$$\begin{bmatrix}
A_{11} & A_{12} & A_{13} \\
A_{21} & A_{22} & A_{23} \\
A_{31} & A_{32} & A_{33}
\end{bmatrix} \begin{bmatrix}
a_1 \\
da_2 \\
da_3
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
1
\end{bmatrix}. \quad (71)$$

Considering the vector $\mathbf{Q}(X_c, X_p) = [1, (x_p - x_c), \frac{(x_p - x_c)^2}{2!}]^T$, the correction function basis $\mathbf{P}$ and the weight function $w$, the coefficients of the matrix $\mathbf{A} \in \mathbb{R}^{3 \times 3}$ can be written as:

$$A_{ij}(X_c) = \sum_{i=1}^3 Q_i(X_c, X_p)P_j(X_{pi} - X_c)w(X_{pi} - X_c).$$ (72)
Step 5: assembly of the linear problem

The steps 1 to 4 allowed the approximation of the derivatives at the collocation nodes as a function of the field values at these nodes and at their support nodes. Sets of coefficients $C_n(X_c)$ are obtained for each derivative so that $D^n x f(X_c) = C_n(X_c) \mathbf{F}(X_c)$. Finally, based on the differential operator and on the boundary conditions, the problem matrices $A$ and $B$ are assembled.

References

1. Runge C (1908) Über eine Methode die partielle Differentialgleichung $\Delta u = \text{constans}$ numerisch zu integrieren. Zeits. f. Math. u. Phys. 56:225–232
2. Ritz W (1908) Über eine neue methode zur lösung gewisser variationsprobleme der mathematischen physik. Journal für die Reine und Angewandte Mathematik 135:1–61
3. Galerkin BG (1915) Rods and plates. series occurring in various questions concerning the elastic equilibrium of rods and plates. Vestnik Inzh 19:897–908
4. Monaghan JJ (1992) Smoothed particle hydrodynamics. Ann Rev Astron Astrophys 30(1):543–574
5. Belytschko T, Lu YY, Gu L (1994) Element-free galerkin methods. Int J Numer Method Eng 37(2):229–256
6. Liu WK, Jun S, Zhang YF (1995) Reproducing kernel particle methods. Int J Numer Meth Fluids 20(8–9):1081–1106
7. Armando Duarte C, Tinsley Oden J (1996) Hp clouds—an Hp meshless method. Numer Methods Partial Differ Equ 12(6):673–705
8. Atluri SN, Zhu T (1998) A new meshless local petrov-galerkin (MLPG) approach in computational mechanics. Comput Mech 22(2):117–127
9. De S, Bathe KJ (2000) The method of finite spheres. Comput Mech 25(4):329–345
10. Chen JS, Wu CT, Yoon S, You Y (2000) A stabilized conforming nodal integration for galerkin mesh-free methods. Int J Numer Meth Eng 50(2):435–466
11. Nguyen VP, Rabczuk T, Bordas S, Duflot M (2008) Meshless methods: a review and computer implementation aspects. Math Comput Simul 79(3):763–813
12. Babuška I, Melenk JM (1995) The partition of unity finite element method. Technical report
13. Babuška I, Melenk JM (1997) The partition of unity method. Int J Numer Meth Eng 40(4):727–758
14. Strouboulis T, Copps K, Babuška I (2001) The generalized finite element method. Comput Methods Appl Mech Eng 190(32–33):4081–4193
15. Moës N, Dolbow J, Belytschko T (1999) A finite element method for crack growth without remeshing. Int J Numer Meth Eng 46(1):131–150
16. Sukumar N, Moës N, Moran B, Belytschko T (2000) Extended finite element method for three-dimensional crack modelling. Int J Numer Meth Eng 48(11):1549–1570
17. Dolbow J, Moës N, Belytschko T (2000) Modeling fracture in mindlin-reissner plates with the extended finite element method. Int J Solids Struct 37(48–50):7161–7183
18. Dolbow J, Moës N, Belytschko T (2001) An extended finite element method for modeling crack growth with frictional contact. Comput Methods Appl Mech Eng 190(51–52):6825–6846
19. Sukumar N, Chopp DL, Moës N, Belytschko T (2001) Modeling holes and inclusions by level sets in the extended finite-element method. Comput Methods Appl Mech Eng 190(46–47):6183–6200
20. Moës N, Belytschko T (2002) Extended finite element method for cohesive crack growth. Eng Fract Mech 69(7):813–833
21. Ji H, Dolbow J (2004) On strategies for enforcing interfacial constraints and evaluating jump conditions with the extended finite element method. Int J Numer Meth Eng 61(14):2508–2535
22. Duflot M, Bordas S (2008) A posteriorierror estimation for extended finite elements by an extended global recovery. Int J Numer Meth Eng 76(8):1123–1138
23. Rabczuk T, Belytschko T, Bordas S, Zi G (2007) Enriched meshfree methods for crack problems. In: 9th National congress on computational mechanics, jun 2007
24. Rabczuk T, Bordas S, Zi G (2007) Initiation, nucleation and propagation of cracks in a cohesive way without mesh enriched in the quasi-sensitive materials: large strains, quasi-static and dynamic. In: 8th national symposium in calculation of structures, Giens, France, 21–25 May
25. Babuška T, Zi G, Bordas S (2007) Enriched finite element methods for dynamic crack propagation problems. In: 5th Australasian congress on applied mechanics, ACAM 2007
26. Bordas S, Zi G, Rabczuk T (2007) Three-dimensional non-linear fracture mechanics by enriched meshfree methods without asymptotic enrichment. IUTAM Symposium on discretization methods for evolving discontinuities. Springer, Netherlands, pp 21–36
27. Bordas S, Rabczuk T, Zi G (2008) Three-dimensional crack initiation, propagation, branching and junction in non-linear
materials by an extended meshfree method without asymptotic
erenrichment. Eng Fract Mech 75(5):943–960
28. Talebi H, Samaniego C, Samaniego E, Rabczuk T (2011)
On the numerical stability and mass-lumping schemes for
explicit enriched meshfree methods. Int J Numer Meth Eng
89(8):1009–1027
29. Natarajan S, Kerfriden P, Bordas S, Mahapatra DR, Rabczuk
T (2011) Enriched element free galerkin method for gradient
elasticity. In: XFEM 2011, jun 2011
30. Belytschko T, Parimi C, Moës N, Sukumar N, Usui S (2002)
Structured extended finite element methods for solids defined by
implicit surfaces. Int J Numer Meth Eng 56(4):609–635
31. Moumni S, Bordas S, Figueredo R, Sansen P (2014) Analysis
using higher-order XFEM: implicit representation of geometrical
features from a given parametric representation. Mech Ind
15(5):443–448
32. Rabczuk T, Bordas S, Goangseup Zi (2010) On three-dimen-
sional modeling of crack growth using partition of unity meth-
ods. Comput Struct 88(23–24):1391–1411
33. Hughes TJR, Cottrell JA, Bazilevs Y (2005) Isogeome-
try and mesh refinement. Comput Methods Appl Mech Eng
194(39–41):4135–4195
34. Simpson RN, Bordas S, Trevelyan J, Rabczuk T (2012) A
two-dimensional isogeometric boundary element method for
elasticstastic analysis. Comput Methods Appl Mech Eng
209–212:87–100
35. Simpson RN, Bordas S, Lian H, Trevelyan J (2013) An iso-
geometric boundary element method for elastostatic analysis: 2d
implementation aspects. Comput Struct 118:2–12
36. Scott MA, Simpson RN, Evans JA, Lipton S, Bordas S, Hughes
TJR, Sederberg TW (2013) Isogeometric boundary element analysis using unstructured T-splines. Comput Methods Appl
Mech Eng 254:197–221
37. Lian H, Simpson RN, Bordas S (2013) Stress analysis without
meshing: isogeometric boundary-element method. Proc Inst Civil
Eng Comput Mech 166(2):88–99
38. Peng X, Atroshchenko E, Bordas S (2014) Damage tolerance
assessment directly from cad: (extended) isogeometric boundary
element methods (xigabem). In: 6th International conference on
advanced computational methods in engineering
39. Atroshchenko E, Bordas S (2015) Fundamental solutions and
dual boundary element methods for fracture in plane cosseral
elasticity. Proc R Soc A Math Phys Eng Sci 471(2179):20150216
40. Lian H, Kerfriden P, Bordas S (2016) Implementation of regu-
larized isogeometric boundary element methods for gradient-
based shape optimization in two-dimensional linear elasticity.
Int J Numer Meth Eng 106(12):972–1017
41. Peng X, Atroshchenko E, Kerfriden P, Bordas S (2017) Isoge-
ometric boundary element methods for three dimensional static
fracture and fatigue crack growth. Comput Methods Appl Mech
Eng 316:151–185
42. Lian H, Kerfriden P, Bordas S (2017) Shape optimization directly from CAD: An isogeometric boundary element approach using
t-splines. Comput Methods Appl Mech Eng 317:1–41
43. Atroshchenko E, Hale JS, Videla Javier A, Potapenko S, Bordas
S (2017) Micro-structured materials: inhogenieties and imper-
fect interfaces in plane micropolar elasticity, a boundary element
approach. Eng Anal Boundary Elem 83:195–203
44. Xu G, Mourrain B, Duvigneau R, Galligo A (2011) Param-
eterization of computational domain in isogeometric analysis:
methods and comparison. Comput Methods Appl Mech Eng
200(23–24):2021–2031
45. Xu G, Mourrain B, Duvigneau R, Galligo A (2013) Analysis-
suitable volume parameterization of multi-block computational
domain in isogeometric applications. Comput Aided Des
45(2):395–404
46. Xu G, Li M, Mourrain B, Rabczuk T, Xu J, Bordas S (2018)
Constructing IGA-suitable planar parameterization from com-
plex CAD boundary by domain partition and global/local opti-
mization. Comput Methods Appl Mech Eng 328:175–200
47. Nguyen VP, Bordas S (2015) Extended isogeometric analysis
for strong and weak discontinuities. Isogeometric methods for
numerical simulation. Springer, Vienna, pp 21–120
48. Nguyen VP, Anitescu C, Bordas S, Rabczuk T (2015) Isogeome-
tric analysis: an overview and computer implementation aspects.
Math Comput Simul 117:89–116
49. Atroshchenko E, Tomar S, Xu G, Bordas S (2018) Weakening the
tight coupling between geometry and simulation in isogeometric
analysis: from sub- and super-geometric analysis to geometry-
independent field approximaTION (GIFT). Int J Numer Meth Eng
114(10):1131–1159
50. Burman E, Hansbo P (2010) Fictitious domain finite ele-
ment methods using cut elements: I. a stabilized lagrange multiplier method. Comput Methods Appl Mech Eng
199(41–44):2680–2686
51. Burman E, Hansbo P (2012) Fictitious domain finite element methods using cut elements: II. a stabilized nitsche method. Appl
Numer Math 62(4):328–341
52. Burman E, Hansbo P (2014) Fictitious domain methods using cut elements: III. a stabilized nitsche method for stokes’ problem.
ESAIM Math Model Numer Anal 48(3):859–874
53. Burman E, Claus S, Hansbo P, Larson MG, Massing A (2014)
CutFEM: Discretizing geometry and partial differential equa-
tions. Int J Numer Meth Eng 104(7):472–501
54. Hansbo P, Larson MG, Zahedi S (2014) A cut finite element
method for a stokes interface problem. Appl Numer Math
85:90–114
55. Burman E, Hansbo P, Larson MG (2015) A stabilized cut finite
element method for partial differential equations on surfaces:
the laplace-beltrami operator. Comput Methods Appl Mech Eng
285:188–207
56. Claus S, Burman E, Massing A (2015) CutFEM: a stabilised
nitsche XFEM method for multi-physics problems
57. Claus S, Kerfriden P (2017) A stable and optimally convergent
LaTIn-CutFEM algorithm for multiple unilateral contact prob-
lems. Int J Numer Meth Eng 113(6):938–966
58. Claus S, Bigot S, Kerfriden P (2018) CutFEM method for ste-
fan-signorini problems with application in pulsed laser ablation.
SIAM J Sci Comput 40(5):B1444–B1469
59. Bordas S, Burman E, Larson MG, Olshanskii MA (2017) Geo-
metrically unfitted finite element methods and applications.
Springer, Berlin
60. Lancaster P, Salkauskas K (1981) Surfaces generated by moving
irregularly-spaced data. In: Proceedings of the 1968 23rd ACM
national conference. ACM Press
61. Shepard D (1968) A two-dimensional interpolation function for irregularly-spaced data. In: Proceeding of the 1968 23rd ACM
national conference. ACM Press
62. Macneal RH (1953) An asymmetrical finite difference network.
Q Appl Math 11(3):295–310
63. Forsythe GE, Wasow WR (1960) Finite difference methods for
partial differential equations. Wiley, Hoboken
64. Jensen PS (1972) Finite difference techniques for variable grids.
Comput Methods Appl Mech Eng 1511
109. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P (1996) Meshless methods: an overview and recent developments. Comput Methods Appl Mech Eng 139(1–4):3–47
110. Szabó B, Babuška I (1991) Finite element analysis. Wiley, Hoboken
111. Guennebaud G, Jacob B, et al (2010) Eigen v3. http://eigen.tuxfamily.org
112. Seel M (2018) dD geometry kernel. In: CGAL user and reference manual. CGAL Editorial Board, 4.13 edition
113. Devillers O, Hornus S, Jamin C (2018) dD triangulations. In: CGAL user and reference manual. CGAL Editorial Board, 4.13 edition
114. Rycroft CH (2009) Voro++: a three-dimensional Voronoi cell library in C++. Chaos Interdiscip J Nonlinear Sci 19(4):041111
115. Ahrens J, Geveci B, Law C (2005) Paraview: an end-user tool for large data visualization. Visualization Handbook 717. Elsevier
116. Ayachit U (2015) The paraview guide: a parallel visualization application. Kitware, Incorporated, New York

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.