Point collocation with mollified piecewise polynomial approximants for high-order partial differential equations

Dewangga Alfarisy^{1,2} | Lavi Zuhal¹ | Michael Ortiz³ | Fehmi Cirak⁴ | Eky Febrianto^{1,2}

¹Faculty of Mechanical and Aerospace Engineering, Bandung Institute of Technology, Bandung, Indonesia

²Glasgow Computational Engineering Centre, University of Glasgow, Glasgow, UK

³Graduate Aerospace Laboratories, California Institute of Technology, Pasadena, California, USA

⁴Department of Engineering, University of Cambridge, Cambridge, UK

Correspondence

Eky Febrianto, Faculty of Mechanical and Aerospace Engineering, Bandung Institute of Technology, Bandung, 40116, Indonesia. Email: eky.febrianto@glasgow.ac.uk

Abstract

The solution approximation for partial differential equations (PDEs) can be substantially improved using smooth basis functions. The recently introduced mollified basis functions are constructed through mollification, or convolution, of cell-wise defined piecewise polynomials with a smooth mollifier of certain characteristics. The properties of the mollified basis functions are governed by the order of the piecewise functions and the smoothness of the mollifier. In this work, we exploit the high-order and high-smoothness properties of the mollified basis functions for solving PDEs through the point collocation method. The basis functions are evaluated at a set of collocation points in the domain. In addition, boundary conditions are imposed at a set of boundary collocation points distributed over the domain boundaries. To ensure the stability of the resulting linear system of equations, the number of collocation points is set larger than the total number of basis functions. The resulting linear system is overdetermined and is solved using the least square technique. The presented numerical examples confirm the convergence of the proposed approximation scheme for Poisson, linear elasticity, and biharmonic problems. We study in particular the influence of the mollifier and the spatial distribution of the collocation points.

K E Y W O R D S

collocation, convolution, high-order, mollification, polytopic meshes, smooth basis functions

1 | INTRODUCTION

1.1 | Motivation

Finding approximate solutions to high-order partial differential equations (PDEs) is a foundational task in various scientific and engineering problems, including gradient theories of elasticity and plasticity,^{1–3} phase-field modelling of sharp interfaces,^{4–6} and plate and shell models.^{7–9} The strong form of these PDEs typically impose stringent smoothness requirements on the approximation schemes. In certain scenarios, it is advantageous to pursue solutions that adhere to the weak form of PDEs, where the smoothness requirements are less stringent. Nevertheless, constructing an approximate solution

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2024 The Author(s). International Journal for Numerical Methods in Engineering published by John Wiley & Sons Ltd.

^{2 of 26} WILEY

for these PDEs commonly involves an ascending sequence of smooth basis functions, which are a subset of the solution space.

On a parallel note, there has been a growing interest in isogeometric analysis (IGA) that employs smooth basis functions prevalent in computer-aided design (CAD), such as the spline-based NURBS ^{10,11} and subdivision surfaces, ¹² to solve various PDEs. These smooth basis functions found applications within the framework of the finite element method (FEM) ^{7,13–16} based on the weak form of PDEs and the Galerkin method. Although the primary aim of IGA is to streamline design and analysis, it has undoubtedly leveraged the broader utilisation of smooth basis functions in computational mechanics.

Solving the strong form of PDEs through the point collocation method (PCM) is feasible when using sufficiently smooth basis functions. In PCM, the PDEs are directly evaluated at specific spatial points, known as collocation points, with boundary conditions imposed at the boundary collocation points. Because no integration is required for evaluating the strong form, collocation presents a straightforward and usually cost-effective alternative to the traditional FEM.^{17,18}

1.2 | Previous work

Various smooth basis functions have been employed in the collocation framework, and many of them are also known in the context of IGA, such as B-splines,^{19–21} T-splines,²² and NURBS.^{23–25} These functions are typically mesh-based, and their smoothness may degrade when the tensor product structure of the mesh breaks, particularly at extraordinary vertices and edges.^{11,26–30} Mesh-free approximants, including moving least squares,³¹ radial basis functions,^{32,33} reproducing kernel,^{34,35} and the maximum entropy (max-ent) approximants ^{36–38} have also been applied for point collocation. Most of these approximants are not polynomial and often need to satisfy certain consistency criteria ^{39,40} to maintain polynomial reproducibility.

Smooth basis functions can be easily constructed through mollification, or convolution, of piecewise polynomials with a smoothing kernel.⁴¹ In a conceptual sense, mollification is fundamentally different from traditional interior approximation schemes. Interior methods, such as FEM, approximate the solution using interpolants belonging to the solution space, for example H^1 . In contrast, exterior methods, such as mollification, enable solution approximation using functions outside of the solution space, that is, using a piecewise polynomial basis. While interior methods are well-established, exterior methods, such as the mollification approach, are simpler, more general, and relatively newer. As highlighted in Febrianto et al.,⁴² mollified approximants maintain the order of the piecewise polynomial basis, while improving smoothness as determined by the kernel, or mollifier. This gives rise to an appealing characteristic of the mollified basis functions where the smoothness and polynomial order can be arbitrary. This smooth basis construction strategy shares similarities with the convolutional definition of B-splines ^{43,44} and those of simplex splines.^{45,46}

In the mollified approach, the polynomials are defined over cells or meshes, enabling a faithful evaluation of the convolution integral, particularly when employing a compactly supported polynomial mollifier. The resulting basis functions span the same space as the piecewise polynomial, therefore avoiding the need for corrections that imposes the polynomial reproducibility on the kernel.^{39,40,47,48} Furthermore, unlike in most mesh-based approximants, the construction of mollified basis functions is not restricted to a specific type of domain discretisation, and their smoothness remains unaffected by extraordinary vertices. This versatility is particularly advantageous when a specific type of discretisation is faster and more robust to obtain, such as the Cartesian grid or Voronoi tessellation,^{49–53} which arguably aligns with the objective of reducing the cost of domain discretisation.

1.3 | Contributions

In this paper, we present a point collocation method for solving the strong form of PDEs using mollified basis functions. The high smoothness of the mollified basis functions makes them particularly suitable for collocation. Compared to its Galerkin implementation,⁴² the mollified-collocation proposed in this work is simpler and more straightforward to implement. In particular, the mollified-collocation sidesteps the need for accurate and variationally consistent integrations when evaluating the domain and surface integrals in the weak form.^{54,55} Furthermore, the mollified-collocation facilitates the strong imposition of boundary conditions, thus circumventing the need for auxiliary methods required to impose stably Dirichlet boundary conditions.^{42,56–58} In this work, we numerically investigate the convergence of the approximate solutions of the Poisson, linear elasticity, and biharmonic problems over polytopic elements, see, for example, Figure 1.



FIGURE 1 The proposed mollified-collocation method. First the domain (A) is discretised into a set of polytopic cells through Voronoi tessellation (B). Each cell has an associated piecewise linear polynomial and a C^{k-1} smooth kernel is used to obtain smooth high-order basis functions. The resulting basis functions are used to solve a *k*-th order PDE problem giving the solution (C). (A) Domain definition. (B) Voronoi tessellation. (C) Solution contour.

Additionally, we assess the effect of the polynomial order, as well as mollifier smoothness and width on the accuracy of numerical approximation.

The unstructured shape of the support of mollified basis functions is incompatible with knot-based collocation point distributions, such as the Greville and Demko abscissae, commonly employed in IGA collocation methods.^{19,23,59,60} Moreover, in the proposed mollified-collocation, multiple basis functions might overlap over each cell, implying that collocating solely at a cell's centroid or the Voronoi seeds would result in an underdetermined system matrix. To address this challenge, more than one collocation point in a cell is distributed according to the selected scheme. Additionally, a constraint is imposed to ensure that the total number of collocation points exceeds the number of basis functions involved in the computation, a quantity that can be predetermined based on the number of cells and the polynomial order. This approach results in an overdetermined (non-square) system matrix, which can be solved using a standard least square technique. In this study, we explore three schemes for distributing the collocation points: uniform, Gauss quadrature, and quasi-random schemes. Furthermore, we analyse their effects on the convergence of the approximation error. Specifically, for quasi-random collocation points, we perform stochastic experiments, and sample perturbations *n* times to ascertain the mean and standard deviation of errors.

1.4 | Overview

The structure of this paper is as follows. First, we revisit the mollification approach for smoothing piecewise polynomials. We then apply this principle to construct smooth basis functions in one and higher dimensions. Subsequently, we discuss the evaluation of the basis functions and their derivatives at a point in space. Later, we describe the use of the mollified basis functions in the collocation framework, including considerations on distributing collocation points in space. Finally, we present several numerical Poisson, linear elasticity, and biharmonic examples in one, two, and three dimensions.

2 | REVIEW OF MOLLIFIED PIECEWISE POLYNOMIAL APPROXIMANTS

In this section, we review the mollified piecewise polynomial approximants used to discretise PDEs. We begin by describing the mollification of piecewise polynomial functions, resulting in a global function smoother than the smoothing kernel, that is, mollifier. This characteristic is particularly advantageous when approximating the solution of PDEs using the collocation method. Next, we derive the mollified basis functions based on the piecewise polynomial defined in each domain partition, referred to as a cell. This work focuses on polytopic meshes, such as the Voronoi tessellation, for domain partitioning. The values of the basis functions can be obtained at a point in space by evaluating a convolution integral, which is also detailed in this section.

2.1 | Mollification of the piecewise polynomial

For brevity, we illustrate the mollification of piecewise polynomials in a one-dimensional setting. We begin by considering the domain $\Omega \in \mathbb{R}^1$ discretised into a set of n_c non-overlapping cells $\{\omega_i\}$, such that,

$$\Omega = \bigcup_{i=1}^{n_c} \omega_i. \tag{1}$$

On each cell ω_i , we define a local polynomial

$$f_i(x) = \begin{cases} \boldsymbol{p}_i(x) \cdot \boldsymbol{\alpha}_i & \text{if } x \in \omega_i \\ 0 & \text{if } x \notin \omega_i \end{cases},$$
(2)

where $p_i(x)$ is a vector containing a local polynomial basis of order r_p and α_i are the respective coefficients of the basis. Common choices for the basis $p_i(x)$ include, but are not limited to, monomials, Lagrange polynomials, and Bernstein polynomials. While it is possible to vary the polynomial order in each cell, this study exclusively focuses on a uniform polynomial order for all cells. The sum of the local polynomials defined over the entire domain Ω constitutes the global polynomial

$$f(x) = \sum_{i} \boldsymbol{p}_{i}(x) \cdot \boldsymbol{\alpha}_{i}.$$
(3)

Note that across the cell boundaries this function will be discontinuous.

We consider the smoothing of the piecewise polynomial f(x) through convolution, referred to as *mollification*, with a smooth kernel referred to as a *mollifier*. The mollification of f(x) with a mollifier m(x) is defined as

$$\widehat{f}(x) = m(x) * f(x) = \int_{\Omega} m(x - y) f(y) dy.$$
(4)

We require the mollifier to be non-negative, have a unit volume, and have finite support per our previous work.⁴² An important characteristic of mollification is that the mollified functions $\hat{f}(x)$ can exactly reproduce polynomials of order r_p .⁴²

When the derivative of the mollifier m(x) exists, the *k*-order derivative of the mollified function $\hat{f}(x)$ is given by

$$\frac{\mathrm{d}^k}{\mathrm{d}^k x} \hat{f}(x) = \int_{\Omega} \frac{\mathrm{d}^k m(x-y)}{\mathrm{d}^k x} f(y) \mathrm{d}y.$$
(5)

For the mollified function $\hat{f}(x)$ to be C^k -smooth, suitable for instances where it is considered a candidate solution for a *k*-th order PDE, the mollifier must be C^{k-1} -smooth. Note that the original function f(x) is discontinuous across the cell boundaries.

An intriguing observation is the parallel between mollification (4) and the widely used convolutional neural networks (CNN). A single CNN layer can be represented as $\hat{f}(x) = \sigma(\sum_{j} m(x - x_{j})f(x_{j}) + b)$, where σ represents the nonlinear activation function and *b* is the bias. This equation can be viewed as the discrete form of the mollification in (4), specifically, $\hat{f}(x) = \sum_{j} m(x - x_{j})f(x_{j})w_{j}$, where x_{j} and w_{j} are the position and weight of the quadrature points, respectively. In CNNs, the convolution operator is often referred to as a *feature map*, with the convolution kernel containing shared weights, or hyperparameters, which require learning. As an example, Figure 2 illustrates the mollification of a piecewise polynomial function f(x) with a C^{1} -smooth symmetric (left) and asymmetric mollifier (right). Moreover, choosing a different mollifier here is equivalent to selecting a different feature map in CNN. Although the parallels are evident, this paper does not pursue this comparison further.

2.2 | Mollified basis functions

We now use the mollification approach to derive uni- and multivariate basis functions. In the univariate case, following the previous discussion in Section 2.1, we partition the domain Ω into a set of non-overlapping cells { ω_i } where the



FIGURE 2 Mollification of piecewise linear functions with C^1 -smooth quadratic B-spline mollifier. The resulting function $\hat{f}(x)$ is C^2 -smooth. Mollification with symmetric and asymmetric kernels is used to exemplify the parallel between the convolution operator and feature maps in CNN. (A) Mollification with a symmetric kernel. (B) Mollification with an asymmetric kernel.

piecewise polynomials $\{f_i(x)\}\$ are defined. By introducing the piecewise definition of the global polynomial f(x) from (3) into the definition of mollification (4), we obtain

$$\widehat{f}(x) = \sum_{i} \alpha_{i} \cdot \int_{\omega_{i}} m(x - y) \, \boldsymbol{p}_{i}(y) \mathrm{d}y.$$
(6)

Here, a polynomial $f_i(x)$ is zero outside of the respective cell ω_i . This allows us to consider as an integration domain ω_i instead of the whole domain Ω . We can then express (6) as a linear combination of basis functions and their coefficients

$$\widehat{f}(x) = \sum_{i} \alpha_{i} \cdot N_{i}(x).$$
(7)

The mollified basis function $N_i(x)$ is obtained by convolving the piecewise polynomial basis $p_i(y)$ belonging to each cell ω_i with the mollifier m(x) such that

$$\boldsymbol{N}_{i}(x) = \int_{\omega_{i}} m(x - y) \boldsymbol{p}_{i}(y) \mathrm{d}y, \qquad (8)$$

where the convolution is individually evaluated for each component of $p_i(x)$. In this paper, we consider the vector $p_i(x)$ as the monomial basis defined locally in each cell ω_i . The local monomials are centred at the centroid c_i of each cell, that is,

$$\boldsymbol{p}_i(x) = \begin{pmatrix} 1 & \xi & \xi^2 & \xi^3 & \dots \end{pmatrix} \quad \text{with } \boldsymbol{\xi} = \frac{2(x-c_i)}{h_c}, \tag{9}$$

where h_c is the cell size. The scaling by $2/h_c$ ensures that all mollified basis functions have a similar maximum value, which improves the conditioning of the system matrix. The derivatives of the basis functions can be obtained using (5) by considering the derivative of the mollifier

$$\frac{\mathrm{d}}{\mathrm{d}x}\boldsymbol{N}_{i}(x) = \int_{\Omega} \frac{\mathrm{d}m(x-y)}{\mathrm{d}x} \boldsymbol{p}_{i}(y)\mathrm{d}y.$$
(10)

As an illustrative example, we consider a one-dimensional domain $\Omega = (0, 1)$ discretised into $n_c = 6$ non-overlapping cells with uniform spacing $h_c = 1/6$. Piecewise polynomials up to degree $r_p = 3$ consisting of monomial basis functions defined over each cell are mollified with a quadratic B-spline mollifier with a width $h_m = 2 h_c$ and a unit volume. Figure 3 depicts the obtained mollified basis functions and their respective second derivatives. We stress that each mollified basis can reproduce polynomials up to degree r_p . Collocating at positions where either the basis or the second derivative is

0.8



FIGURE 3 Basis functions (left column) over $n_c = 6$ uniform cells for $r_p \in \{0, 1, 2, 3\}$ using a quadratic B-spline mollifier and their corresponding second derivatives (right column). The blue dots situated along the *x*-axis denote the cell boundaries. (A) $N_i^0(x)$. (B) $\frac{d^2 N_i^0(x)}{dx^2}$. (C) $N_i^1(x)$. (D) $\frac{d^2 N_i^1(x)}{dx^2}$. (E) $N_i^2(x)$. (F) $\frac{d^2 N_i^2(x)}{dx^2}$. (G) $N_i^3(x)$. (H) $\frac{d^2 N_i^3(x)}{dx^2}$.

zero is often undesirable. For this specific example, such zero-valued positions can be predicted, as shown in Figure 3. For instance, the linear $N_i^1(x)$ and cubic $N_i^3(x)$ are zero at the cell's centre, and the second derivative of $N_i^0(x)$ coincides with the cell boundary. However, predicting such locations becomes challenging for non-uniform arrangements in both univariate and multivariate cases, especially for polytopic partitions.

Without loss of generality, in the following we continue our discussion on multivariate basis functions, focusing on the bivariate case. In the two-dimensional setting, for the mollifier, we consider the tensor-product of its one-dimensional description

$$m(\mathbf{x}) = m(x^{(1)}) \cdot m(x^{(2)}).$$
(11)

Consequently, the support of the mollifier \square_x is a square, and generally, a hypercube in the multi-dimensional case. Similar to the univariate case, the domain Ω is subdivided into non-overlapping cells { ω_i }. For the multivariate case considered in this work, a polytopic discretisation of the domain Ω is used. Nevertheless, other non-overlapping partitions, such as Delaunay triangulation/tetrahedralisation, quadrilateral/hexahedral mesh, and Cartesian grid, are also viable options.

In each cell ω_i , we consider a set of monomial basis functions centred at the centroid of the cell x_c ,

$$\boldsymbol{p}_{i}(\boldsymbol{x}) = \begin{pmatrix} 1 & \xi^{(1)} & \xi^{(2)} & (\xi^{(1)})^{2} & (\xi^{(2)})^{2} & \dots \end{pmatrix} \quad \text{with } \xi^{(j)} = \frac{2\left(x^{(j)} - x_{c}^{(j)}\right)}{h}, \tag{12}$$

where h is the average cell size. In our numerical experiments, h is usually computed as the square root of the average area of the cells in the domain. The bivariate basis functions are obtained by evaluating the convolution integral

$$\boldsymbol{N}_{i}(\boldsymbol{x}) = \int_{\omega_{i}} m(\boldsymbol{x} - \boldsymbol{y}) \, \boldsymbol{p}_{i}(\boldsymbol{y}) \mathrm{d}\boldsymbol{y}.$$
(13)

As an example, we consider a five-sided polygonal cell ω_i . Mollifying either the constant or linear monomials with a tensor product C^2 -smooth spline mollifier (34) yields the basis functions presented in Figure 4. Likewise, the derivatives of the obtained multivariate basis functions with respect to the coordinate axes can be obtained as in (10).

Next, we focus on the support $\hat{\omega}_i$ of the mollified basis functions corresponding to piecewise polynomials of the cell ω_i . The support $\hat{\omega}_i$ is given as the Minkowski sum of the cell ω_i with the support of the mollifier \Box_0 , that is,

$$\widehat{\omega}_i = \omega_i \oplus \square_{\mathbf{0}} = \{ \mathbf{x} + \mathbf{y} | \ \mathbf{x} \in \omega_i, \mathbf{y} \in \square_{\mathbf{0}} \}.$$
(14)

In our implementation, to obtain the support $\hat{\omega}_i$, we first position the mollifier at the vertices of the cell $\mathbf{v}_{i,j}$, that is, $\Box_{\mathbf{v}_{i,j}}$. The subscript *j* is enumerated over the indices of all the vertices of ω_i . We then take the union of the vertices of $\Box_{\mathbf{v}_{i,j}}$ for all *j* and combine them using a convex hull algorithm to obtain $\hat{\omega}_i$. For a comprehensive introduction to Minkowski sums, we refer the readers to Reference 61.

2.3 | Basis evaluation at a point

In this section, we outline the procedure for evaluating the mollified basis functions at a particular point $x \in \mathbb{R}^d$. For the univariate case, the convolution integral in (8) can be symbolically computed, resulting in a closed-form expression that can be evaluated at any point $x \in \mathbb{R}^1$. Such closed-form expressions of the basis functions (8) can be obtained using symbolic programming tools such as Mathematica and SymPy.

When evaluating the multivariate basis functions, the convolution integral in (13) should be numerically computed. To evaluate the basis $N_i(x)$ at a point $x \in \mathbb{R}^2$, we first position the mollifier centred at the evaluation point x, defining the support as $\Box_x = \text{supp } m(x - y)$, as shown in Figure 5. Both the kernel and polynomial have compact supports, thus, the integrand in (13) is nonzero only within the intersection between the support of the mollifier and the cell. This observation further shrinks the integration domain to

$$\tau_{i,\mathbf{x}} := \Box_{\mathbf{x}} \cap \omega_i, \tag{15}$$



FIGURE 4 Bivariate constant and linear mollified basis functions with a C^2 -smooth spline mollifier on a cell $\omega_i \in \mathbb{R}^2$. The dashed lines in (A) and (C) indicate the boundary of the cell. (A) Mollifier and constant local basis functions. (B) Basis functions of order $r_p = 0$ obtained with the mollifier shown in (A). (C) Mollifier and linear local basis functions. (D) Basis functions of order $r_p = 1$ obtained with the mollifier shown in (C).

where the intersection τ_i is convex because both the cell ω_i and the mollifier support \Box_x are convex. The convolution integral used to compute the basis functions is then simplified to

$$N_i(\mathbf{x}) = \int_{\tau_{ix}} m(\mathbf{x} - \mathbf{y}) \mathbf{p}_i(\mathbf{y}) \mathrm{d}\mathbf{y}.$$
 (16)

Here, we focus on a two-dimensional example where the local polynomial is defined within a cell $\omega_i \in \mathbb{R}^2$ and a polynomial mollifier is employed. To evaluate (16), we first triangulate $\tau_{i,x}$ by connecting its edges with its centroid. Gauss quadrature points are then mapped from a reference triangle to each generated triangle to evaluate (16). We consider an example using a set of polytopic cells $\{\omega_i\}_{i=1}^6$ as shown in Figure 5A. To evaluate basis functions at \mathbf{x} , we place the mollifier support centred at \mathbf{x} , as shown in Figure 5A. The mollifier support $\Box_{\mathbf{x}}$ intersects three cells: $\omega_1, \omega_3, \text{ and } \omega_4$. The basis functions associated with these cells can be computed by first obtaining their intersection with the mollifier support, namely $\tau_{1,\mathbf{x}}, \tau_{3,\mathbf{x}}$, and $\tau_{4,\mathbf{x}}$. Subsequently, we evaluate the convolution integral (16) over these intersection domains, as shown in Figure 5B.

In the three dimensional case, $\tau_{i,x}$ is a convex polyhedron. The convolution integral over $\tau_{i,x}$ can be evaluated by tessellating the polyhedron into tetrahedra before distributing the quadrature points. An alternative approach involves the successive application of the divergence theorem to reduce the dimension of integration domains.⁶² When using a polynomial mollifier, for example, a tensor product spline, the integrand in (16) is also a polynomial function and can be



FIGURE 5 The intersection of the mollifier located at *x* with the cells ω_i (A) generates the domain for evaluating the convolution integral $\tau_{i,x}$ (B). (A) The mollifier \Box_x overlapping some of the cells ω_i . (B) $\tau_{i,x}$.

accurately integrated. For further information on integrating polynomial functions over arbitrary polytopes, interested readers are referred to References 63–65.

3 | POINT COLLOCATION METHOD

In this section, we outline the discretisation using mollified basis functions for solving PDEs. The smoothness and approximation properties of the mollified basis functions render them suitable for solving PDEs in their strong forms. Our approach assumes that the domain Ω is discretised into non-overlapping convex polytopes, in which piecewise polynomials are defined. Following the discussion in Section 2.3, we emphasise that the basis functions and their derivatives can be evaluated at arbitrary points in space, which we later refer to as collocation points. Furthermore, we delve into the considerations involved in determining the collocation points in our numerical studies.

3.1 | Discretisation

For simplicity, we consider the Poisson-Dirichlet equation involving a scalar variable *u* over the domain $\Omega \in \mathbb{R}^d$ with $d \in \{1, 2, 3\}$ as a model problem:

$$-\nabla^2 u(\mathbf{x}) = s(\mathbf{x}) \qquad \text{in } \Omega , \qquad (17a)$$

$$u(\mathbf{x}) = \overline{u}$$
 on Γ_D , (17b)

where *s* is the source and \overline{u} is the prescribed solution field on the Dirichlet boundary Γ_D . The main characteristic of the point collocation method lies in utilising the strong form (17) rather than its weak form. The field variable *u* is approximated by a linear combination of mollified basis functions, according to the discretisation of the domain Ω into n_c cells,

$$u_h(\boldsymbol{x}) = \sum_i \boldsymbol{N}_i(\boldsymbol{x}) \cdot \boldsymbol{u}_i.$$
(18)

10 of 26 WILEY

In each cell ω_i , the basis functions form a vector consisting of contributions from each monomial. To ensure polynomial reproducibility near the boundary, the domain Ω is padded with ghost cells with a size proportional to the mollifier width. Substituting u_h as an approximation to u in the Equation (17a) yields

$$-\sum_{i} \left(\nabla^2 \boldsymbol{N}_i(\boldsymbol{x}) \right) \cdot \boldsymbol{u}_i = s(\boldsymbol{x}).$$
⁽¹⁹⁾

Here, the continuity requirement for the basis functions is at least $C^{2.66}$ Hence, based on the mollification properties discussed in Section 2, the minimum continuity for the mollifier $m(\mathbf{x})$ is C^{1} .

We consider a set of points $\{z_j\}_{j=1}^{n_z}$, referred to as collocation points, where Equation (19) is evaluated and the boundary conditions (17b) are enforced. We subdivide the point set into two subsets, that is, interior collocation points $\{z_k^I\}_{k=1}^{n_z^I}$ and boundary collocation points $\{z_l^R\}_{l=1}^{n_z^R}$ so that

$$\{\mathbf{z}_{j}\}_{j=1}^{n_{z}^{I}+n_{z}^{B}} = \{\mathbf{z}_{k}^{I}\}_{k=1}^{n_{z}^{I}} \cup \{\mathbf{z}_{l}^{B}\}_{l=1}^{n_{z}^{B}}.$$
(20)

Here, the capital superscripts *I* and *B* distinguish points belonging to the interior and boundary set, respectively. Consequently, the total number of collocation points n_z comprises the number of interior points n_z^I and boundary points n_z^B , so that

$$n_z = n_z^I + n_z^B. aga{21}$$

In this approach, we require the total number of collocation points n_z to be greater than, or at least equal to, the total number of the basis functions used in discretising the solution field, denoted as n_b . We evaluate the Equation (19) at each interior collocation point \mathbf{z}_k^I , that is,

$$-\sum_{i} \left(\nabla^2 \boldsymbol{N}_i(\boldsymbol{z}_k^I) \right) \cdot \boldsymbol{u}_i = s(\boldsymbol{z}_k^I), \tag{22}$$

and enumerate the index k from 1 to the total number of interior points n_z^I . The Dirichlet boundary condition is strongly imposed at the boundary collocation points z_l^B , that is,

$$\sum_{i} N_{i}(\boldsymbol{z}_{l}^{B}) \cdot \boldsymbol{u}_{i} = \overline{\boldsymbol{u}},$$
(23)

where the index *l* goes from 1 to the total number of boundary points n_z^B . Similarly, for Neumann type of boundary conditions, the gradient of the basis functions ∇N_i is evaluated at the associated boundary collocation points.

The two discrete Equations (22) and (23) are compactly expressed in matrix notation as follows

$$\mathbf{C}\mathbf{u} = \mathbf{s}.\tag{24}$$

Matrix **C** is a non-square matrix with size $n_z \times n_b$ with $n_z \ge n_b$. Matrix **C** consists of the two blocks **C**^{*I*} and **C**^{*B*}, corresponding to the contributions of the internal and boundary collocation points. Each row of **C**^{*I*} contains the expansion of (22) when evaluated at an interior point, and **C**^{*B*} consists of the evaluation of (23) at boundary collocation points. The entries of **C** are non-zero only when the collocation point z_j is located within the support of a basis, that is, $z_j \in \text{supp } N_i$. The support of the basis is influenced by the mollifier width, thus, we can deduce that matrix **C** is denser when the mollifier is wide, and conversely, sparser when the mollifier is narrow. The right-hand size vector **s** has n_z elements and can be subdivided into $(\mathbf{s}^I \mathbf{u}^B)^T$ containing the source terms $s(\mathbf{z}^I_k)$ at the interior points and boundary values $\overline{u}(\mathbf{z}^B_l)$ at the boundary points.

The choice $n_z > n_b$ implies an overdetermined linear system. This system can be solved in a least-square sense by multiplying (24) with \mathbf{C}^{T} ,

$$\mathbf{C}^{\mathsf{T}}\mathbf{C}\mathbf{u} = \mathbf{C}^{\mathsf{T}}\mathbf{s} \quad \Rightarrow \quad \mathbf{G}\mathbf{u} = \mathbf{w},\tag{25}$$

RIGHTSLINK()

where **G** is a square matrix of size $n_b \times n_b$. In the case of $n_z = n_b$, the linear system of equations (24) has a solution only when **C** is non-singular. Therefore, in our implementation we prefer the number of collocation points to be greater than the number of basis functions $n_z > n_b$ to avoid a situation where the system yields no solution. The selection of collocation points in our analysis will be described in the following section.

Our final note concerns the conditioning of the collocation matrix **C**. The mollified basis functions typically involve high-order polynomials due to the order of the local approximants and the mollifier, which potentially leads to poor conditioning of **C** if untreated. Such conditioning problems become more severe for high-order PDEs. In our implementation, we improve the conditioning of **C** using two scaling techniques. The first pertains to scaling for the basis functions and the second involves scaling for the derivatives. The first type of scaling is introduced in the description of local approximants (9), where each monomial is scaled according to the factor $(h_c/2)^p$, where *p* is the monomial degree. This scaling ensures that the maximum value of monomials of any order is the same. The second type of scaling adjusts the *n*-th derivative of the basis function using factor $(h_m)^n$ to account for magnitude discrepancies across the derivatives. Evidently, the inverse of the second scaling factor should be applied upon acquiring the numerical solution of the linear system.

3.2 | Spatial distribution of collocation points

The entries in the linear system (24) depend on the location of the collocation points z_j where the basis functions and their second derivatives are evaluated. Various strategies have been proposed for selecting the position of the collocation points. When piecewise polynomial approximants are used, Gauss quadrature can be devised as collocation points.^{67,68} For mesh-free approximation schemes, such as the reproducing kernel particle method (RKPM) and local max-ent approximation, collocation points can be chosen either similarly to the points representing the basis functions,^{34,37} or differently.^{33,36} In the context of isogeometric collocation using B-spline and NURBS basis functions, Greville and Demko abscissae are employed as the locations for collocating the strong form of the governing Equation (19),^{19,23,59,60} which exploits the knot structure of B-splines. Moreover, recent studies have reported the existence of collocation points with improved convergence properties similar to the Galerkin schemes.^{25,69,70}

As mentioned in Section 2.2, mollified basis functions are obtained by convolving piecewise polynomials defined over a polytopic cell with a mollifier. The support of the basis functions generally do not coincide with the cell boundaries. Therefore, it is challenging to exploit a specific structure to find optimal coordinates for collocation points when discretising with mollified basis functions. This is unlike the case with B-splines, where the tensor-product structure can be used to identify optimal or privileged collocation points. Nonetheless, this work does not pursue the identification of such optimal coordinates for collocation points. Furthermore, collocating at Voronoi seeds, similar to the nodes in meshfree methods, leads to underdetermined system matrices because each cell has multiple associated basis functions. Therefore, we explore several strategies for distributing the collocation points, including uniform, Gauss quadrature, and quasi-random point distributions.

To ensure stability, we choose the number of collocation points so that the linear system matrix is overdetermined. This allows for a less stringent approach to selecting the position of collocation points. In our approach, the number of collocation points, including those on the boundary, can be described by expanding the interior points n_z^I in terms of the number of cells

$$n_{z} = n_{z}^{I} + n_{z}^{B} = \beta \ n_{c} + n_{z}^{B} \ge n_{b} \ . \tag{26}$$

Here, the factor $\beta \ge 1$ accounts for the total number of monomial basis functions in each cell, which directly corresponds to the polynomial order r_p . The number of boundary collocation points n_z^B is explicitly separated in (26) so that β can conveniently be chosen as an integer. For example, in the set of basis functions shown in Figure 3, the total number of basis functions is $(n_c + 2) |\mathbf{p}|$, where $n_c = 6$ and $|\mathbf{p}| = r_p + 1 = 4$. Moreover, the addition with two accounts for the appended ghost cells. Therefore, the total number of basis functions involved is $n_b = 32$. In this case, the closest integer β is selected so that (26) is satisfied, which is $\beta = 6$.

In this paper, we consider three methods for distributing collocation points: Gauss quadrature, uniform, and quasi-random point distribution. In the univariate setting, obtaining these point distributions is straightforward, as depicted in Figure 6. In the multivariate setting, we can obtain the Gauss point arrangement by first tessellating the convex polytopic cells into simplices (i.e., triangles or tetrahedra in two or three dimensions, respectively). Gauss points of order γ are then mapped from the reference simplex. The uniform collocation point distribution can be obtained by



FIGURE 6 One-dimensional collocation point distributions.

placing the points in an equidistant manner in each coordinate axis and subsequently taking their tensor products in higher dimensions. Note that we only consider uniformly distributed collocation points for numerical examples with tensor product domains. Lastly, the quasi-random point collocation distribution is obtained by introducing perturbation to an initial point distribution, which is usually chosen as uniform. The random perturbation $\epsilon \sim \mathcal{U}(-\sigma, \sigma)$ is generated in each coordinate axis from a univariate uniform density function with maximum perturbation σ . The coordinates of collocation points are detailed for each example in Section 4.

4 | EXAMPLES

In this section, we present several numerical experiments with increasing complexity to investigate the convergence property of the proposed mollified-collocation approach. We initially investigate the performance of the mollified-collocation scheme in solving one-dimensional Poisson and biharmonic problems. The effects of the piecewise polynomial degree, mollifier, and collocation points on the convergence are studied. We then proceed with the numerical study of a two-dimensional elastic plate, a plate with a hole, a plate bending, and a three-dimensional cube. In all the examples, convergence is studied using the relative discretisation errors as in Reference 37. The relative error between field \boldsymbol{v} and its numerical approximation \boldsymbol{v}_h can be defined as

$$e(\mathbf{v}, \mathbf{v}_h) = \left(\frac{\sum_{k=1}^{n_z} (\mathbf{v}^k - \mathbf{v}_h^k) (\mathbf{v}^k - \mathbf{v}_h^k)^{\mathsf{T}}}{\sum_{k=1}^{n_z} \mathbf{v}^k (\mathbf{v}^k)^{\mathsf{T}}}\right)^{\frac{1}{2}},\tag{27}$$

where \boldsymbol{v}^k and \boldsymbol{v}_h^k are the evaluation of the field and its approximation at collocation point \boldsymbol{z}_k .

4.1 | One-dimensional examples

4.1.1 | One-dimensional Poisson problem

As a first example we consider the solution of the one-dimensional Poisson-Dirichlet problem $-d^2u(x)/dx^2 = s(x)$ on the domain $\Omega = (0, 1) \in \mathbb{R}^1$. The source term s(x) is chosen such that the solution is equal to $u(x) = \sin(3\pi x)$. To construct the mollified basis functions, we first consider the decomposition of the domain Ω into a set of non-overlapping cells $\{\omega_i\}_{i=1}^{n_c}$ of size $h_{c,i}$ where the piecewise polynomial basis functions $\mathbf{p}_i(x)$ are defined. We initially consider the coarse $n_c = 6$ non-uniform cells with size of $h_{c,1} = 0.15$, $h_{c,2} = 0.2$, $h_{c,3} = 0.15$, $h_{c,4} = 0.15$, $h_{c,5} = 0.2$ and $h_{c,6} = 0.15$. We then consider refinement through the bisectioning of each cell. The convolution integral is evaluated by assuming a mollifier function m(x) with support size of

$$h_m = 2\left(\max_j h_{c,j}\right). \tag{28}$$

The closed form of the basis functions can be obtained by analytically evaluating the convolution integral involving the piecewise polynomials and the mollifier. To ensure completeness at the boundary, one ghost cell layer of size h_m is padded to each end of the domain. In the following set of experiments, we study the influence of collocation point distribution, local polynomial basis $p_i(x)$, and the mollifier m(x) on the convergence of the solution approximation.

We first study the influence of three types of point distribution: uniform, Gauss quadrature, and quasi-random, on the error convergence. Even though the knot-based abscissae are applicable in the one-dimensional case, they are not extendable for higher dimensions with unstructured polytopic partitions, and therefore, are not considered in the present analysis. The uniform point distribution is obtained by placing the collocation points over the domain Ω in an equidistant manner without considering the cell boundaries. By contrast, the Gauss quadrature points are distributed by mapping the standard Gauss quadrature from the parametric domain onto each cell. Finally, the quasi-random point distribution is obtained by adding a small random perturbation sampled from a uniform distribution $\epsilon \sim \mathcal{U}(-\sigma, \sigma)$ to the uniformly distributed interior collocation points. Here the parameter σ is chosen as 10% of the equidistant spacing of the uniform configuration.

The total number of collocation points n_z should not be less than the number of basis functions n_b to ensure that the system matrix is not underdetermined. In our one dimensional example, the total number of basis functions involved in the computation n_b depends on the number of cells n_c and the order of the local polynomial r_p

$$n_b = (n_c + 2) (r_p + 1) .$$
⁽²⁹⁾

The above expression considers the contribution from one layer of ghost cell on each side. We consider the total number of collocation points n_z following (21)

$$n_z = (\beta \ n_c) + 2 \ , \tag{30}$$

where the term inside the bracket resembles the internal collocation points. For consistency, we use $\beta = 6$ in the examples throughout this section following the minimum number of unknown coefficients imposed by the cubic polynomial basis $r_p = 3$ at the coarsest level $n_c = 6$. The constant two in (30) accounts for the boundary collocation points where the Dirichlet boundary conditions are imposed. Figure 6 depicts the three distributions of the collocation points for the one-dimensional test case with cubic polynomials $r_p = 3$ and $n_c = 6$ cells.

Figure 7 shows the error convergence in both the L^2 -norm and H^1 -seminorm when using a normalised quadratic B-spline mollifier and quadratic polynomial order $r_p = 2$. In particular, for the perturbed point distribution, the random perturbation ϵ is first obtained for each interior collocation point, generating a set of quasi-random points. We obtain such a point distribution one hundred times, from which the mean and standard deviation of solution errors can be obtained. Figure 7 shows that the three collocation point distributions yield a convergence rate of 2 in both the L^2 -norm



FIGURE 7 One-dimensional Poisson problem. Convergence with normalised quadratic B-spline mollifier and quadratic polynomial basis for three collocation point distributions: Uniform, Gauss quadrature, and quasi-random. (A) L^2 -norm error. (B) H^1 -seminorm error.

14 of 26 WILFY

and H^1 -seminorm, which aligns with the collocation methods using IGA and max-ent basis functions.^{23,37} In Figure 7, the three point distributions yield only small differences in the convergence constants. Furthermore, the quasi-random point arrangement has the highest mean error among the three point distributions considered. On the other hand, the quasi-random point distribution has the lowest convergence rate of mean errors among the three point distributions. Moreover, for this distribution, the standard deviation of the errors increases as refinement progresses.

The second experiment aims to study the influence of the local polynomial order $r_p \in \{1, 2, 3\}$ on the error convergence of the numerical approximation. We consider a normalised quadratic B-spline mollifier with width chosen according to (28). In this example, we choose uniformly distributed collocation points with factor $\beta = 6$ and spacing $1/(n_z + 1)$. Figure 8 presents the error convergence for each polynomial degrees r_p in both the L^2 -norm and H^1 -seminorm. The convergence rates in the L^2 norm are 0.80, 2.27, and 2.58 for polynomial orders $r_p = 1$, $r_p = 2$, and $r_p = 3$, respectively. A similar trend can be observed in the H^1 -seminorm, where the average convergence rates are 0.85, 2.21, and 3.16 for polynomial orders $r_p = 1$, $r_p = 2$, and $r_p = 3$, respectively. These results suggest that the proposed approach converges with rate r_p in both error norms with lower rates for the odd polynomial order, particularly in the L^2 -norm, which resembles the findings reported in previous studies.^{23,37}

Subsequently, we investigate the influence of mollifier width and smoothness on the error convergence. We use in this example uniformly distributed collocation points with factor $\beta = 6$ and a piecewise polynomial of order $r_p = 2$. For the first case, we modify the width of a quadratic B-spline mollifier through scaling factor κ , that is,

$$h_m = 2\kappa \left(\max_j h_{c,j} \right) \quad \text{with } \kappa \in \{0.75, \ 1., \ 1.25\}.$$
 (31)

The increase in mollifier width leads to larger support of the basis functions. This implies that more basis functions have non-zero values at a collocation point which leads to a denser matrix system. Figure 9 shows that changing the mollifier width leads to only a slight difference in the error magnitude while keeping the convergence rate as $r_p = 2$ for both the L^2 -norm and H^1 -seminorm.

We next investigate the effect of mollifier smoothness on the convergence of the approximation error. Here, we consider quadratic and cubic B-spline mollifiers which are C^1 - and C^2 -smooth, respectively. The obtained mollified basis functions are C^2 - and C^3 -smooth, respectively. This aligns with the smoothness requirement from the PDE that the solution has to be at least C^2 -continuous. The mollifier width is chosen according to (28), that is, $\kappa = 1$. Figure 10 displays the convergence plot for both the L^2 -norm and H^1 -seminorm errors. It is evident that higher mollifier smoothness improves the convergence constants while keeping a convergence rate of $r_p = 2$.

Finally, we address the cost comparison between the proposed mollified-collocation method and the mollified-FEM ⁴² by estimating the number of point evaluations required to construct the linear system. We consider a quadratic polynomial with $r_p = 2$ over 48 cells, mollified with a quadratic B-spline mollifier. The resulting basis functions have, as described



FIGURE 8 One-dimensional Poisson problem. Convergence with normalised quadratic B-spline mollifier and piecewise polynomial of order $r_p = \{1, 2, 3\}$. (A) L^2 -norm error. (B) H^1 -seminorm error.



FIGURE 9 One-dimensional Poisson problem. Convergence with normalised quadratic B-spline mollifier of width factor $\kappa = \{0.75, 1., 1.25\}$ and quadratic piecewise polynomial. (A) L^2 -norm error. (B) H^1 -seminorm error.



FIGURE 10 One-dimensional Poisson problem. Convergence with normalised quadratic and cubic B-spline mollifier and quadratic piecewise polynomial ($r_p = 2$). (A) L^2 -norm error. (B) H^1 -seminorm error.

in (8), monomials up to order 5. Consequently, the stiffness integrand in the mollified-FEM has monomials up to order 8, necessitating 5 Gauss quadrature points on each non-overlapping polynomial segment, thus yielding a total of 240 point evaluations. In contrast, the proposed mollified-collocation method only requires evaluation at a minimum 150 collocation points. This discrepancy is further accentuated in higher dimensions and when using a finer discretisation. The number of quadrature points in mollified-FEM can be somewhat reduced by using the variationally consistent integration (VCI) approach. VCI requires the solution of a small dense matrix to derive the corrective coefficients in each cell.

One-dimensional biharmonic problem 4.1.2

We next consider the one-dimensional biharmonic problem $d^4u(x)/dx^4 = s(x)$ over the domain $\Omega = (0, 1) \in \mathbb{R}^1$. The source term s(x) is chosen such that the solution is $u(x) = \sin(3\pi x)$. Both the value u and the first derivative u' are prescribed at the boundaries x = 0 and x = 1. In this example, the domain Ω is uniformly discretised into n_c cells of size



FIGURE 11 One-dimensional biharmonic problem. Convergence with the C³ spline mollifier with a piecewise polynomial of order $r_p = \{5, 6\}$. (A) L^2 -norm error. (B) H^1 -seminorm error.

 $h_c = 1/n_c$. In each cell, we consider a polynomial of order $r_p \in \{5, 6\}$ consisting of $r_p + 1$ monomial basis and their respective coefficients. Because of the higher derivatives that appear in the PDE, the smoothness requirement for the solution u(x) is C^4 , which requires that the mollifier should be at least C^3 smooth. Hence, we consider a C^3 spline mollifier

$$m(x) = \begin{cases} \frac{315}{128h_m} \left(1 - 16\left(\frac{x}{h_m}\right)^2 + 96\left(\frac{x}{h_m}\right)^4 - 256\left(\frac{x}{h_m}\right)^6 + 256\left(\frac{x}{h_m}\right)^8 \right) & \text{if } |x| < \frac{h_m}{2} \\ 0 & \text{if } |x| \ge \frac{h_m}{2} \end{cases}$$
(32)

where the mollifier width h_m is chosen to be twice the cell size $h_m = 2h_c$. We uniformly distribute n_z collocation points with the factor β according to (30). We analyse the effect of the total number of collocation points in terms of the factor β , where $\beta = \{8, 10\}$ are compared for the fifth-order polynomial and $\beta = \{10, 12\}$ are considered for the sixth-order polynomial. These factors are larger than the ones specified in the Poisson examples (Section 4.1.1) because of the higher order polynomials involved.

The convergence of relative errors in the L^2 -norm and H^1 -seminorm are shown in Figure 11. In the L^2 -norm, the error converges approximately with rate $r_p - 2$ for both $r_p = 5$ and $r_p = 6$. In the H^1 -seminorm, a convergence rate of $r_p - 2$ can also be observed for both polynomial orders. The number of collocation points, dictated by factor β , affects the convergence constants in both norms. For the same polynomial order, a higher β indicates more collocation points, which yields lower convergence constants. Morever, a higher number of collocation points improves the conditioning of the system matrix (24).

4.2 **Two-dimensional examples**

4.2.1 Two-dimensional elasticity

We next consider the linear elasticity problem $\nabla \cdot \sigma(\mathbf{x}) = \mathbf{b}(\mathbf{x})$ on the two-dimensional square domain $\Omega = (0, 1) \times (0, 1)$. The material of the plate has Young's modulus E = 1000 and its Poisson's ratio is v = 0.3. The body force b(x) is chosen such that the solution equals to

$$u^{(1)} = u^{(2)} = \sin(\pi x^{(1)}) \sin(\pi x^{(2)}).$$
(33)

The domain Ω is partitioned into n_c cells using the Voronoi diagram of n_c non-uniformly distributed seeds. The cells corresponding to $n_c = \{16, 64, 256\}$ are depicted in Figure 12. One ghost layer is padded around the plate to ensure 0970207, 2024, 18, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/nne.7548 by Test, Wiley Online Library on [05/11/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



FIGURE 12 Voronoi diagram consisting of $n_c = 16$ (A), $n_c = 64$ (B), and $n_c = 256$ (C). The Voronoi seeds are obtained from Reference 42. (A) $n_c = 16$. (B) $n_c = 64$. (C) $n_c = 256$.

completeness near the plate's boundaries. We consider linear and quadratic local polynomials $r_p \in \{1, 2\}$ mollified with a C^2 -smooth spline mollifier obtained from the tensor product of a one-dimensional spline curve

$$m(x) = \begin{cases} \frac{35}{16h_m} \left(1 - 12\left(\frac{x}{h_m}\right)^2 + 48\left(\frac{x}{h_m}\right)^4 - 64\left(\frac{x}{h_m}\right)^6 \right) & \text{if } |x| < \frac{h_m}{2} \\ 0 & \text{if } |x| \ge \frac{h_m}{2} \end{cases}$$
(34)

where the mollifier width h_m is obtained by averaging the total area of the domain Ω by the total number of internal cells, that is,

$$h_m = 2\left(\frac{1}{n_c}\right)^{0.5}.$$
(35)

We determine the total number of collocation points n_z according to the total number of basis functions n_b . In particular, we require $n_z \ge n_b$ regardless of the type of point distributions used. The number of basis functions is determined by

$$n_b = (n_c + n_g) |\boldsymbol{p}_i| \tag{36}$$

where n_c is the number of cells and n_g is the number of ghost cells. The notation $|\mathbf{p}_i|$ indicates the number of monomial basis functions in each cell, which depends on the order r_p and dimension d. For the two-dimensional case, $|\mathbf{p}_i| = 3$ applies for the linear and $|\mathbf{p}_i| = 6$ applies for the quadratic polynomial order. For example, when the number of internal cells is $n_c = 16$ and the number of ghost cells is $n_g = 20$, the total number of basis functions for the quadratic case is $n_b = (16 + 20) \cdot 6 = 216.$

We arrange the collocation points according to the uniform, Gauss quadrature, and quasi-random distribution in a similar way as in the previous one-dimensional examples. In the uniform case, we consider a tensor product of the one-dimensional uniform point distribution over the domain Ω . In the above example with $n_c = 16$ cells and the quadratic polynomial order, we select $n_z = 16^2$ collocation points as depicted in Figure 13A. For the Gauss quadrature case, we first triangulate the polytopic cells and subsequently distribute the quadrature points by mapping them from a reference triangle. In the case of quadrilateral cells, we use Gauss quadrature points mapped from a quadrilateral reference element, as illustrated in Figure 13B. The quadrature order is uniformly chosen for all cells so that the criterion $n_z \ge n_b$ is satisfied. For the quasi-random point distribution, we first consider the uniform point arrangement and apply a small perturbation to each point. Moreover, in each coordinate axis, we consider a perturbation sampled from a uniform distribution $\epsilon \sim$ $\mathcal{U}(-\sigma, \sigma)$, where σ is 15% of the equidistant point spacing (Figure 13C).

Figure 14 shows the solution contour to the elastic plate problem computed over $n_c = 16$ polytopic cells. Figure 15 illustrates the convergence of the solution approximation in the L^2 -norm and H^1 -seminorm. The convergence rates of r_p are achieved in both norms for both the linear $r_p = 1$ and quadratic $r_p = 2$. For the linear case $r_p = 1$, the error constants



FIGURE 13 Collocation points according to the uniform (A), Gauss quadrature (B), and quasi-random (C) distributions for the case with $n_c = 16$ cells. (A) Uniform. (B) Gauss quadrature. (C) Quasi-random.



FIGURE 14 Solution to the elastic plate problem computed over $n_c = 16$ cell discretisation.



FIGURE 15 Two-dimensional elastic plate problem. Convergence with a C^2 hexic spline mollifier with piecewise polynomial of order $r_p = \{1, 2\}$. (A) L^2 -norm error. (B) H^1 -seminorm error.

4.2.2 | Two-dimensional plate bending

In this section, we consider the two-dimensional bending problem on a two-dimensional square plate defined by the domain $\Omega = (0, 1) \times (0, 1)$ with boundary Γ . The governing biharmonic equation and the boundary conditions read:

$$D\Delta^2 u = q \qquad \text{in } \Omega, \tag{37a}$$

$$u = 0 \qquad \text{on } \Gamma$$
, (37b)

$$\nabla u \cdot \boldsymbol{n} = 0 \qquad \text{on } \Gamma \,. \tag{37c}$$

Here, *D* is a constant associated with the material properties and the thickness of the plate. In our computations, we assume D = 1 and the right-hand term $q(\mathbf{x})$ is chosen according to

$$q(\mathbf{x}) = -16\pi^4 \left(\cos(2\pi x^{(1)}) - 4\cos(2\pi x^{(1)}) \cos(2\pi x^{(2)}) + \cos(2\pi x^{(2)}) \right)$$
(38)

which leads to the solution of

$$u(\mathbf{x}) = \left(1 - \cos(2\pi x^{(1)})\right) \left(1 - \cos(2\pi x^{(2)})\right).$$
(39)

The domain Ω is partitioned into n_c non-uniform polytopic cells as used in the previous two-dimensional elastic plate cells mentioned in Section 4.2.1 and Figure 12. One ghost layer is padded around the plate to ensure completeness near the plate's boundaries. We consider quartic and quintic local polynomials $r_p \in \{4, 5\}$ mollified with a C^4 -smooth spline mollifier obtained from the tensor product of a one-dimensional spline curve

$$m(x) = \begin{cases} \frac{2772}{1024h_m} \left(1 - 20\left(\frac{x}{h_m}\right)^2 + 160\left(\frac{x}{h_m}\right)^4 - 640\left(\frac{x}{h_m}\right)^6 + 1280\left(\frac{x}{h_m}\right)^8 - 640\left(\frac{x}{h_m}\right)^{10}\right) & \text{if } |x| < \frac{h_m}{2} \\ 0 & \text{if } |x| \ge \frac{h_m}{2} \end{cases}.$$
(40)

The mollifier width h_m is obtained by averaging the total area of the domain Ω by the total number of internal cells, like in our plate case (Section 4.2.1). In this example, we consider a Gauss quadrature of order γ as collocation points mapped from the simplices division of each cells. The Gauss quadrature order is set as $\gamma = 7$, which satisfies our requirement that $n_z \le n_b$. According to (36), for the polynomial order $r_p = 4$, we consider the total number of monomial basis functions in one cell to be $|\mathbf{p}_i| = 15$. Likewise, for $r_p = 5$, there are $|\mathbf{p}_i| = 21$ monomials in one Voronoi cell.

Figure 16 shows the solution contour to the plate bending problem computed over $n_c = 16$ polytopic cells. The convergence of relative errors in the L^2 -norm and H^1 -seminorm is illustrated in Figure 17. In both the L^2 -norm and H^1 -seminorm, the error converges approximately with a rate of $r_p - 2$ for both $r_p = 4$ and $r_p = 5$. In addition, the scaling of the monomial basis introduced in (9) is crucial for the stability of the linear system because of the high-order polynomial involved. Moreover, because of the higher order derivatives involved in this example, we use another scaling factor of $(h_m)^d$, where *d* is the derivative order, to better condition the linear system, as explained in Section 3.1.

4.2.3 | Two-dimensional infinite plate with a hole

In this section, we consider an infinite plate with a circular hole subjected to uniaxial tension. The tension $\sigma_{\infty} = 10^6$ is applied in the *x*-direction (Figure 18). Due to symmetry, we consider only a quarter of the plate with a unit length and the

19 of 26

WILEY-



FIGURE 16 Solution to the two-dimensional biharmonic problem computed over $n_c = 16$ cell discretisation.



FIGURE 17 Two-dimensional biharmonic plate problem. Convergence with C⁴-smooth spline mollifier with a piecewise polynomial of order $r_p = \{4, 5\}$. (A) L^2 -norm error. (B) H^1 -seminorm error.

a hole radius of a = 0.25. The material has Young's modulus of $E = 70 \times 10^6$ and its Poisson's ratio is v = 0.3. The infinite plate with a hole problem has a closed-form analytic solution⁷¹ as follows

$$u_x = \frac{\sigma_\infty a}{8\mu} \left[\frac{r}{a} (\kappa + 1) \cos(\theta) + \frac{2a}{r} \left((1 + \kappa) \cos(\theta) + \cos(3\theta) \right) - \frac{2a^3}{r^3} \cos(3\theta) \right]$$
(41)

$$u_y = \frac{\sigma_{\infty} a}{8\mu} \left[\frac{r}{a} (\kappa - 3) \sin(\theta) + \frac{2a}{r} \left((1 - \kappa) \sin(\theta) + \sin(3\theta) \right) - \frac{2a^3}{r^3} \sin(3\theta) \right] , \qquad (42)$$

where *r* is the distance from the centre of the hole and κ is the Kolosov constant

$$\kappa = \frac{3 - \nu}{1 + \nu} \tag{43}$$

for plane stress. Dirichlet boundary conditions are imposed over the entire boundary of the plate.

20 of 26

WILEY



FIGURE 18 Schematic of the elastic plate with a hole problem.



FIGURE 19 Distribution of collocation points inside the plate-with-hole domain and on its boundary according to the Gauss quadrature arrangement. (A) $n_c = 16$. (B) $n_c = 64$. (C) $n_c = 256$.

In this example, we consider an initial mesh of $n_c = 16$ quadrilateral cells as shown in Figure 19A. The refined meshes are obtained by introducing new vertices in the middle of each edge and subsequently subdividing the cells into four, see Figure 19B,C for $n_c = 64$ and $n_c = 256$, respectively. As in previous examples, ghost cells are considered to ensure completeness at the boundary and are explicitly shown in Figure 19. Here, we consider Gauss quadrature of order γ as collocation points mapped from the reference quadrilateral to each cell, see Figure 19. The Gauss quadrature order is $\gamma = 3$ and $\gamma = 4$ for linear $r_p = 1$ and quadratic $r_p = 2$ polynomial orders, respectively. Because the cell edges do not align with the domain boundary around the hole, some collocation points may lie outside of the domain Ω . Such collocation points are removed from computation through an auxiliary detection method using the signed distance function $\phi(\mathbf{x})$ of the domain Ω . By definition, the signed distance function $\phi(\mathbf{x})$ is positive inside the domain, negative outside the domain, and the zeroth isosurface $\phi^{-1}(0)$ corresponds to the boundary Γ . Therefore, we consider only quadrature points that satisfy $\phi(\mathbf{z}_k) > 10^{-5}$ as internal collocation points. Furthermore, to enhance accuracy around the hole, boundary collocation points are mapped from a reference one-dimensional line according to the eight-th order Gauss quadrature $\gamma = 8$.

To analyse this problem, the C^2 -smooth spline mollifier is used as described in (34), yielding C^3 -smooth basis functions. Figure 20 shows the convergence of the solution error in the energy norm. It is evident that convergence rate of r_p is achieved for both the linear $r_p = 1$ and quadratic $r_p = 2$ polynomial orders. Furthermore, we would like to compare the basis evaluation between the proposed collocation approach and the finite element (FE) implementation.⁴² As described



FIGURE 20 Elastic plate with a hole. Convergence of the relative energy norm error with a C^2 -smooth spline mollifier and a local polynomial basis of degree $r_p = \{1, 2\}$.

in Section 2.3, both approaches require evaluating basis functions at Gauss quadrature points. It is important to note that the adequate quadrature order needed for the FE is determined by the polynomial order of the integrands, whereas in the collocation approach, the appropriate quadrature order is determined through a loose criterion ($n_z \ge n_b$) to avoid underdetermined matrix system. For instance, to integrate the FE stiffness term in the quadratic case $r_p = 2$ with the C^2 polynomial mollifier, the integrand has a maximum polynomial order of 16. Therefore, the minimum Gauss quadrature order required is $\gamma = 9$. Assuming quadrilateral cells, leads to 81 points per cell. By contrast, the proposed collocation approach requires $\gamma = 4$ to ensure the system is overdetermined. Moreover, although auxiliary techniques such as variationally consistent integration ^{54,55} can be used to aid in the FE integration, they still require an adequate base quadrature order.

4.3 | Three-dimensional heat transfer on a solid body

In this example, we consider a steady heat transfer with a constant coefficient of 1, that is, $\nabla^2 T(\mathbf{x}) = f(\mathbf{x})$. The source term is applied so that the temperature solution is $T(\mathbf{x}) = \sin(\pi x^{(1)}) \sin(\pi x^{(2)}) \sin(\pi x^{(3)})$. A linear local polynomial basis function is then chosen in each cell and a tensor product of the C^1 -smooth mollifier is used, as described in Reference 42. We use the proposed mollified-collocation method to solve the heat transfer problem over solid bodies discretised using polytopic meshes. Here we consider two objects as our domain: a unit cube and a dodecahedron with a uniform edge length of 0.3, as shown in Figures 21 and 1, respectively. The objects are discretised using the VoronoiMesh function in Mathe*matica* by distributing quasi-uniform Voronoi seeds over a bounding box Ω_{\Box} larger than the domain Ω . Subsequently, we exclude the cells corresponding to zero-valued basis in Ω . The resulting active Voronoi meshes consist of 1000 cells for the box example and 622 cells for the dodecahedron. The discretisation has a total number of basis functions $n_b = 4000$ for the cube and $n_b = 2488$ for the dodecahedron. Furthermore, an auxiliary intersection algorithm is required to determine part of the Voronoi cells inside the domain Ω for distributing the collocation points. After obtaining part of the cell lying inside the domain, we first tessellate this part into tetrahedra and map Gauss points onto each tetrahedron. The boundary collocation points are obtained by mapping the Gauss quadratures onto the surface triangles. This approach results in $n_z = 8373$ collocation points for the cube example, and $n_z = 6522$ points for the dodecahedron. The isocontours of the computed temperature are shown in Figures 21 and 1 for the cube and dodecahedron, respectively. It is worth emphasising that our method allows for non-boundary fitting discretisation, which simplifies the domain discretisation into cells.



FIGURE 21 Three-dimensional box case result. The top left image depicts the cell coordinate system, the top right image depicts the exploded Voronoi mesh that is used in this case, the bottom left image depicts the points distribution, and the bottom right image depicts the temperature contour of the result. (A) Domain definition. (B) Voronoi mesh. (C) Temperature distribution over the collocation points. (D) Temperature contour.

5 | CONCLUSIONS

We presented a point collocation method that uses the smooth mollified basis functions to approximate the solutions of Poisson, linear elasticity, and biharmonic equations. The method attained high-order numerical convergence. The approximation properties of the mollified basis were characterised by the order of local polynomial approximants and the smoothness of the mollifier. The smoothness of the approximation using mollified basis functions remained intact even across meshes with irregular polytopic shapes. Here, we considered polynomial mollifiers with compact support and unit volume. To evaluate the basis functions at a point, a convolution integral is solved by first obtaining a compact integration domain, which is the intersection between the polytope and a box. They represent the support of the piecewise polynomial and the mollifier, respectively. Such a geometric intersection can be robustly computed using polytope clipping and convex hull algorithms implemented in Mathematica, Python, and similar geometry processing libraries.^{50,72,73} In this work, we constructed an overdetermined linear system by choosing the number of collocation points to exceed the number of basis functions. Furthermore, to improve the conditioning of the system matrix, we scaled the basis functions and their derivatives. These treatments yielded good convergence properties irrespective of the mollifier type, support size, or the

24 of 26 WILEY

spatial distribution of the collocation points. Finally, the proposed mollified collocation approach dispensed with the need for integrating the domain and boundary integrals in contrast to the mollified Galerkin approach.⁴²

There are several promising future extensions of the proposed mollified collocation approach. The first proposition concerns the requirement for the padded ghost cells to ensure the polynomial reproduction property of the mollified basis functions throughout the domain. Consequently, the basis functions associated with ghost cells have to be considered in the computation, which ultimately leads to an increased number of bases. A systematic approach to guarantee polynomial reproduction without needing ghost cells will improve the efficiency of the mollified-collocation method. One promising avenue includes morphing the kernel when approaching the boundary. In addition, the recent development of boundary-fitted Voronoi tessellations can be incorporated into the mollified-collocation framework. Efficient implementations of such algorithms have been reported, for example, References 50,51. Furthermore, an obvious extension of the proposed method is to consider local p- and h- refinement. Local p- refinement is possible because of the individual prescription of the local polynomial order for each cell. When Voronoi tessellation is used for domain discretisation, the h- refinement becomes less obvious. One possible approach involves adding more Voronoi seeds to the area of interest and regenerating the Voronoi tessellation. The regularity of cells can then be improved using the standard Lloyd's iteration.^{49,74} Another promising future work involves exploring and establishing the connection between mollification and convolutional neural networks.⁷⁵ This will allow for efficient uni- and multivariate basis evaluations, using open-source machine learning tools such as PyTorch and TensorFlow. Finally, the choice of local monomial basis requires appropriate scaling factors to better condition the system matrix. Some studies are available on the alternative basis functions for polytopic elements, for example, References 76–78, which could be adapted for application in the mollified context.

CONFLICT OF INTEREST STATEMENT

The authors declare no potential conflict of interests.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Dewangga Alfarisy b https://orcid.org/0009-0005-6839-6843 Michael Ortiz b https://orcid.org/0000-0001-5877-4824 Fehmi Cirak b https://orcid.org/0000-0002-9274-6904 Eky Febrianto b https://orcid.org/0000-0001-8838-6240

REFERENCES

- 1. Rudraraju S, Ven VA, Garikipati K. Three-dimensional isogeometric solutions to general boundary value problems of Toupin's gradient elasticity theory at finite strains. *Comput Methods Appl Mech Eng.* 2014;278:705-728.
- 2. Codony D, Marco O, Fernández-Méndez S, Arias I. An immersed boundary hierarchical B-spline method for flexoelectricity. *Comput Methods Appl Mech Eng.* 2019;354:750-782.
- 3. Khakalo S, Laukkanen A. Strain gradient elasto-plasticity model: 3D isogeometric implementation and applications to cellular structures. *Comput Methods Appl Mech Eng.* 2022;388:114225.
- Gómez H, Calo VM, Bazilevs Y, Hughes TJR. Isogeometric analysis of the Cahn–Hilliard phase-field model. Comput Methods Appl Mech Eng. 2008;197(49-50):4333-4352.
- Liu J, Dede L, Evans JA, Borden MJ, Hughes TJR. Isogeometric analysis of the advective Cahn–Hilliard equation: spinodal decomposition under shear flow. J Comput Phys. 2013;242:321-350.
- 6. Ambati M, Gerasimov T, De Lorenzis L. A review on phase-field models of brittle fracture and a new fast hybrid formulation. *Comput Mech.* 2015;55:383-405.
- 7. Cirak F, Ortiz M, Schröder P. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *Int J Numer Methods Eng.* 2000;47(12):2039-2072.
- 8. Kiendl J, Bletzinger KU, Linhard J, Wüchner R. Isogeometric shell analysis with Kirchhoff–Love elements. *Comput Methods Appl Mech Eng.* 2009;198:3902-3914.
- 9. Cirak F, Long Q, Bhattacharya K, Warner M. Computational analysis of liquid crystalline elastomer membranes: changing Gaussian curvature without stretch energy. *Int J Solids Struct.* 2014;51:144-153.
- 10. Piegl L, Tiller W. The NURBS Book. Monographs in Visual Communication. 2nd ed. Springer; 1997.
- 11. Farin G. Curves and Surfaces for CAGD: A Practical Guide. 5th ed. Morgan Kaufmann Publishers Inc.; 2001.
- 12. Peters J, Reif U. Subdivision Surfaces. Springer Series in Geometry and Computing. Springer; 2008.

- 13. Cirak F, Scott MJ, Antonsson EK, Ortiz M, Schröder P. Integrated modeling, finite-element analysis, and engineering design for thin-shell structures using subdivision. *Comput-Aided Des.* 2002;34(2):137-148.
- 14. Höllig K. Finite Element Methods with B-Splines. Society for Industrial and Applied Mathematics; 2003.
- 15. Hughes TJR, Cottrell JA, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput Methods Appl Mech Eng.* 2005;194(39-41):4135-4195.
- 16. Kamensky D, Hsu MC, Schillinger D, et al. An immersogeometric variational framework for fluid-structure interaction: application to bioprosthetic heart valves. *Comput Methods Appl Mech Eng.* 2015;284:1005-1053.
- 17. Schillinger D, Evans JA, Reali A, Scott MA, Hughes TJ. Isogeometric collocation: cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations. *Comput Methods Appl Mech Eng.* 2013;267:170-232.
- Hillman M, Chen JS. Performance Comparison of Nodally Integrated Galerkin Meshfree Methods and Nodally Collocated Strong Form Meshfree Methods. Springer International Publishing; 2018:145-164.
- 19. Manni C, Reali A, Speleers H. Isogeometric collocation methods with generalized B-splines. *Comput Math Appl.* 2015;70(7):1659-1675.
- 20. Torre M, Morganti S, Pasqualini FS, Düster A, Reali A. Immersed isogeometric analysis based on a hybrid collocation/finite cell method. *Comput Methods Appl Mech Eng.* 2023;405:115856.
- 21. Torre M, Morganti S, Nitti A, de Tullio MD, Pasqualini FS, Reali A. Isogeometric mixed collocation of nearly-incompressible electromechanics in finite deformations for cardiac muscle simulations. *Comput Methods Appl Mech Eng.* 2023;411:116055.
- 22. Casquero H, Liu L, Bona-Casas C, Zhang Y, Gomez H. A hybrid variational-collocation immersed method for fluid-structure interaction using unstructured T-splines. *Int J Numer Methods Eng.* 2016;105(11):855-880.
- 23. Auricchio F, Veiga LB, Hughes TJR, Reali A, Sangalli G. Isogeometric collocation methods. *Math Models Methods Appl Sci.* 2010;20(11):2075-2107.
- 24. Auricchio F, Beirão da Veiga L, Hughes T, Reali A, Sangalli G. Isogeometric collocation for elastostatics and explicit dynamics. *Comput Methods Appl Mech Eng.* 2012;249-252:2-14.
- 25. Anitescu C, Jia Y, Zhang YJ, Rabczuk T. An isogeometric collocation method using superconvergent points. *Comput Methods Appl Mech Eng.* 2015;284:1073-1097.
- 26. Toshniwal D, Speleers H, Hughes TJ. Smooth cubic spline spaces on unstructured quadrilateral meshes with particular emphasis on extraordinary points: geometric design and isogeometric analysis considerations. *Comput Methods Appl Mech Eng.* 2017;327:411-458.
- 27. Zhang Q, Sabin M, Cirak F. Subdivision surfaces with isogeometric analysis adapted refinement weights. *Comput-Aided Des.* 2018;102:104-114.
- 28. Zhang Q, Cirak F. Manifold-based isogeometric analysis basis functions with prescribed sharp features. *Comput Methods Appl Mech Eng.* 2020;359:112659.
- 29. Koh KJ, Toshniwal D, Cirak F. An optimally convergent smooth blended B-spline construction for semi-structured quadrilateral and hexahedral meshes. *Comput Methods Appl Mech Eng.* 2022;399:115438.
- 30. Verhelst HM, Weinmüller P, Mantzaflaris A, Takacs T, Toshniwal D. A comparison of smooth basis constructions for isogeometric analysis. *Comput Methods Appl Mech Eng.* 2024;419:116659.
- 31. Wu NJ, Tsay TK. A robust local polynomial collocation method. Int J Numer Methods Eng. 2013;93(4):355-375.
- 32. Zhang X, Song KZ, Lu MW, Liu X. Meshless methods based on collocation with radial basis functions. Comput Mech. 2000;26(4):333-343.
- 33. Hu HY, Chen JS, Hu W. Weighted radial basis collocation method for boundary value problems. *Int J Numer Methods Eng.* 2007;69(13):2736-2757.
- 34. Aluru NR. A point collocation method based on reproducing kernel approximations. Int J Numer Methods Eng. 2000;47(6):1083-1121.
- 35. Chi SW, Chen JS, Hu HY, Yang JP. A gradient reproducing kernel collocation method for boundary value problems. *Int J Numer Methods Eng.* 2013;93(13):1381-1402.
- 36. Fan L, Coombs WM, Augarde CE. The point collocation method with a local maximum entropy approach. Comput Struct. 2018;201:1-14.
- 37. Greco F, Arroyo M. High-order maximum-entropy collocation methods. *Comput Methods Appl Mech Eng.* 2020;367:113115.
- Fan L, Coombs WM, Augarde CE. An adaptive local maximum entropy point collocation method for linear elasticity. *Comput Struct*. 2021;256:106644.
- 39. Liu WK, Jun S, Zhang YF. Reproducing kernel particle methods. Int J Numer Methods Fluids. 1995;20:1081-1106.
- 40. Bonet J, Kulasegaram S. Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations. *Int J Numer Methods Eng.* 2000;47(6):1189-1214.
- 41. Adams RA, Fournier JJF. Sobolev Spaces. Elsevier Science; 2003.
- 42. Febrianto E, Ortiz M, Cirak F. Mollified finite element approximants of arbitrary order and smoothness. *Comput Methods Appl Mech Eng.* 2021;373:113513.
- 43. De Boor C. B(Asic)-Spline Basics. Tech. rep, Madison mathematics research center. Winsconsin University; 1986.
- 44. Sabin M. Analysis and Design of Univariate Subdivision Schemes. Springer; 2010.
- 45. Micchelli CA. Geometric Methods for Piecewise Polynomial Surfaces. SIAM; 1995:149-206.
- 46. Grandine TA. The stable evaluation of multivariate simplex splines. *Math Comput.* 1988;50(181):197-205.
- 47. Liu WK, Han W, Lu H, Li S, Cao J. Reproducing kernel element method. Part I: theoretical formulation. *Comput Methods Appl Mech Eng.* 2004;193(12):933-951.
- Li S, Lu H, Han W, Liu WK, Simkins DC. Reproducing kernel element method part II: globally conforming I^m/Cⁿ hierarchies. Comput Methods Appl Mech Eng. 2004;193(12):953-987.
- 49. Du Q, Faber V, Gunzburger M. Centroidal Voronoi tessellations: applications and algorithms. SIAM Rev. 1999;41(4):637-676.

26 of 26 WILEY-

- 50. Ray N, Sokolov D, Lefebvre S, Lévy B. Meshless Voronoi on the GPU. ACM Trans Graph. 2018;37(6):1-12.
- 51. Abdelkader A, Bajaj CL, Ebeida MS, et al. VoroCrust: Voronoi meshing without clipping. ACM Trans. Graph. 2020;39(3):1-16.
- 52. Bishop JE, Sukumar N. Polyhedral finite elements for nonlinear solid mechanics using tetrahedral subdivisions and dual-cell aggregation. *Comput Aided Geometr Des.* 2020;77:101812.
- 53. Sukumar N, Tupek MR. Virtual elements on agglomerated finite elements to increase the critical time step in elastodynamic simulations. *Int J Numer Methods Eng.* 2022;123(19):4702-4725.
- 54. Chen JS, Hillman M, Rüter M. An arbitrary order variationally consistent integration for Galerkin meshfree methods. *Int J Numer Methods Eng.* 2013;95(5):387-418.
- 55. Hillman M, Chen J, Bazilevs Y. Variationally consistent domain integration for isogeometric analysis. *Comput Methods Appl Mech Eng.* 2015;284:521-540.
- 56. Nitsche JA. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen Aus Dem Mathematischen Seminar der Universität Hamburg*. 1971;36:9-15.
- 57. Rüberg T, Cirak F. Subdivision-stabilised immersed b-spline finite elements for moving boundary flows. *Comput Methods Appl Mech Eng.* 2012;209–212:266-283.
- 58. Boiveau T, Burman E. A penalty-free Nitsche method for the weak imposition of boundary conditions in compressible and incompressible elasticity. *IMA J Numer Anal.* 2015;36(2):770-795.
- 59. Johnson RW. Higher order B-spline collocation at the Greville abscissae. Appl Numer Math. 2005;52(1):63-75.
- 60. Wang D, Qi D, Li X. Superconvergent isogeometric collocation method with Greville points. *Comput Methods Appl Mech Eng.* 2021;377:113689.
- 61. Berg M, Cheong O, Kreveld M, Overmars M. Computational Geometry: Algorithms and Applications. 3rd ed. Springer Publishing Company; 2010.
- 62. Mirtich B. Fast and accurate computation of polyhedral mass properties. J Graph Tools. 1996;1:31-50.
- 63. Mousavi SE, Sukumar N. Numerical integration of polynomials and discontinuous functions on irregular convex polygons and polyhedrons. *Comput Mech.* 2011;47(5):535-554.
- 64. Chin EB, Sukumar N. An efficient method to integrate polynomials over polytopes and curved solids. *Comput Aided Geometr Des.* 2020;82:101914.
- 65. Antolin P, Buffa A, Cirillo E. Region extraction in mesh intersection. Comput-Aided Des. 2023;156:103448.
- 66. Elman HC, Silvester DJ, Wathen AJ. Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics. Oxford University Press; 2014.
- 67. Boor C, Swartz B. Collocation at Gaussian points. SIAM J Numer Anal. 1973;10(4):582-606.
- 68. Finlayson B. The Method of Weighted Residuals and Variational Principles. Society for Industrial and Applied Mathematics; 2013.
- 69. Gomez H, De Lorenzis L. The variational collocation method. Comput Methods Appl Mech Eng. 2016;309:152-181.
- 70. Montardini M, Sangalli G, Tamellini L. Optimal-order isogeometric collocation at Galerkin superconvergent points. *Comput Methods Appl Mech Eng.* 2017;316:741-757.
- 71. Timoshenko S, Goodier JN. Theory of Elasticity. McGraw-Hill; 1970.
- 72. Barber CB, Dobkin DP, Huhdanpaa H. The Quickhull algorithm for convex hulls. ACM Trans Math Softw. 1996;22(4):469-483.
- 73. Rycroft CH. VORO++: A three-dimensional Voronoi cell library in C++. Chaos: an interdisciplinary. J Nonlinear Sci. 2009;19(4):041111.
- 74. Lloyd S. Least squares quantization in PCM. IEEE Trans Inf Theory. 1982;28(2):129-137.
- 75. LeCun Y, Bengio Y. Convolutional Networks for Images, Speech, and Time Series. MIT Press; 1998:255-258.
- 76. Sukumar N, Malsch EA. Recent advances in the construction of polygonal finite element interpolants. *Arch Comput Methods Eng.* 2006;13(1):129-163.
- 77. Schneider T, Dumas J, Gao X, Botsch M, Panozzo D, Zorin D. Poly-spline finite-element method. ACM Trans. Graph. 2019;38:3.
- 78. Bunge A, Herholz P, Sorkine-Hornung O, Botsch M, Kazhdan M. Variational quadratic shape functions for polygons and Polyhedra. *ACM Trans. Graph.* 2022;41(4):1-14.

How to cite this article: Alfarisy D, Zuhal L, Ortiz M, Cirak F, Febrianto E. Point collocation with mollified piecewise polynomial approximants for high-order partial differential equations. *Int J Numer Methods Eng.* 2024;125(18):e7548. doi: 10.1002/nme.7548