ORIGINAL PAPER

h-Adaptive radial basis function finite difference method for linear elasticity problems

Balázs Tóth[1] · Alexander Düster[2]

Abstract

In this research work, the radial basis function finite difference method (RBF-FD) is further developed to solve one- and two-dimensional boundary value problems in linear elasticity. The related differentiation weights are generated by using the extended version of the RBF utilizing a polynomial basis. The type of the RBF is restricted to polyharmonic splines (PHS), i.e., a combination of the odd m-order PHS $\phi(r) = r^m$ with additional polynomials up to degree p will serve as the basis. Furthermore, a new residual-based adaptive point-cloud refinement algorithm will be presented and its numerical performance will be demonstrated. The computational efficiency of the PHS RBF-FD approach is tested by means of the relative errors measured in ℓ_2-norm on several representative benchmark problems with smooth and non-smooth solutions, using h-adaptive, uniform, and quasi-uniform point-cloud refinement.

Keywords Radial basis functions · Finite differences · Polyharmonic splines · Polynomials · Adaptivity · Linear elasticity

1 Introduction

In the last two decades, the application of the RBF technique as a mesh-free scheme has become popular for the numerical solution of partial differential equations (PDE). The fundamental concept of RBF-based numerical approximation of differential operators originates from the research works [41,42], and a large number of papers have since contributed to the continuous development of the RBF-FD method [8,17,19,34,36,37,43] and dealt with the approximative solution of numerous BVPs and initial-BVPs appearing in science and engineering [6,25,27,38]. The main advantage of the advanced FD methods based on the RBFs is that they are well-suited for arbitrarily-scattered (non-structured) point layouts. Furthermore the stencil topology can be chosen in different ways, thereby avoiding the fixed rectangular

✉ Balázs Tóth
  balazs.toth@uni-miskolc.hu

  Alexander Düster
  alexander.duester@tuhh.de

[1] Institute of Applied Mechanics, University of Miskolc, Miskolc-Egyetemváros, Hungary

[2] Numerical Structural Analysis with Application in Ship Technology (M-10), Hamburg University of Technology, Hamburg, Germany

grid-like point layout considered as a restriction of the standard numerical methods applied to the solution of PDEs [21].

During the application of the RBF-FD scheme, the so-called differential weights are locally computed at a certain evaluation point within a stencil, using an RBF-based approximation. In this case, the stencil is constructed by a fixed number of points that are closest to the evaluation point under consideration. Accordingly, we start off by computing the differential weights at each point of the point-cloud generated on the considered computational domain, before storing them for the global assembling procedure—which is the next important step in solving the PDEs. Thus, the weight computations are performed as the pre-processing step of the solution algorithm. Since this computation can be carried out independently for all evaluation points, this part of the RBF-FD method is well parallelizable.

The RBFs can be divided into two main groups: piecewise- and infinitely smooth RBFs. The latter ones—such as the Gaussian-, the multiquadratic and inverse quadratic and inverse multiquadratic RBFs—provide spectral convergence behavior for smooth BVPs. However, they are not independent of a so-called shape parameter, which controls the gradient of the RBFs, thereby having significant effect on the condition number of the RBF matrices and thus on the solution accuracy. The optimal value of the shape parameter has to be found for all evaluation points during the weight

computation in order to achieve numerical stability and accuracy [5,7,10,12,22]. The RBF coefficient matrix conditioning problem can be overcome by a stabilization algorithm using small shape parameter values [20,26] or by applying RBFs with a hybrid kernel [29].

The piecewise smooth RBFs (such as the PHS) have a great advantage over the infinitely smooth RBFs, since they are not dependent on a shape parameter and since their usage can yield higher-order accuracy—though only with algebraic *h*-convergence characteristics. Additionally, the piecewise smooth RBFs perform well for the approximation of the differential operators near to the boundaries where the stencil geometry becomes half-sided, i.e., the deterioration of the numerical approximation accuracy due to the Runge phenomenon can be avoided, see [1–4].

In accordance with the above-mentioned beneficial properties of the piecewise smooth RBFs, in this paper, the PHS RBFs supplemented with polynomial functions will be introduced as the basis function space in order to develop an advanced FD scheme for the numerical solution of 1D and 2D BVPs in linear elasticity. The robust convergence behavior of the FD method based on this type of augmented basis function spaces has been verified in [1,2,4,14]. Furthermore, it has been experienced in numerical studies presented in [4,14] that the highest polynomial degree specifies the convergence rate, while the PHS RBF order has a significant influence only on the achievable accuracy. One order in the convergence rate is lost in the case of the numerical approximation of the first (spatial) derivatives, while two orders are lost for approximating the second derivative. Thus, the convergence rate depends on the order of the differential operator [4,5,13,14,30].

Interestingly, the increase in the stencil size does not lead to better accuracy and faster convergence when a fixed higher-degree polynomial basis is used. However, if lower-degree polynomials are applied and their degree is kept fixed, increasing the stencil size can yield improved accuracy while the convergence order remains unchanged [2]. This type of independence on the stencil size motivated the construction of the *p*-adaptive PHS RBF-FD method. Here the desired accuracy is achieved by locally increasing the polynomial degree and adapting the required stencil size, see [30], for which the non-uniform point-cloud is generated by the code NodeLab [28].

Adaptive strategies such as *h*-, *p*- and *hp*-refinement are popular for finite element methods (FEMs), see [11,35] for example. The basic idea of the adaptive schemes is to introduce an error indicator function that guides the non-uniform *h*- and/or *p*-refinement until the prescribed stopping criteria are met. Although the FEM-based adaptive strategies are more prevalent than the FD method-based ones, the construction of the adaptive point refinement procedure is much easier for RBF-FD schemes than the implementation of the adaptive

mesh refinement process in the case of FEMs. Residual error- and Zienkiewicz-Zhu type indicator-based adaptive RBF-FD algorithms are also developed using the Gaussian RBF, for example in [23,24,31]. It is worth emphasizing here that, in the case of *h*-adaptive point refinement, the DistMesh code is one of the most suitable for fast generation of 2D point distributions with variable density [15,31,33].

In accordance with the above findings, the aim of this paper is to develop a residual-based *h*-adaptive point refinement algorithm for 1D BVPs, which is based on the odd *m*-order PHS supplemented with additional polynomials. Firstly, the computational efficiency of the approach will be investigated on smooth and non-smooth 1D BVPs, using the uniform- and the adaptive point-cloud refinement schemes. Secondly, the *h*-convergence behavior of the PHS RBF-FD method complemented with polynomials will be tested on smooth and non-smooth 2D BVPs in linear elasticity, applying uniform, quasi-uniform and *h*-adaptive point distributions.

The paper is organized as follows. In Sect. 2, the basic system of second-order differential equations and the related Dirichlet- and Neumann-type boundary conditions are summarized for 2D linear elasticity problems. Section 3 briefly presents the procedure to evaluate the local RBF matrix and to compute the related differentiation weights from the local linear algebraic system of equations using PHS combined with polynomials. In Sect. 4, a comprehensive overview of the applied stencil geometry is given, discussing how to treat the Neumann-type boundary conditions with the aid of ghost points. Furthermore, a new adaptive point-cloud refinement strategy is presented. In Sect. 5, the *h*-convergence behavior of the PHS RBF-FD method is studied for 1D BVPs, applying uniform as well as the proposed adaptive point-cloud refinement. Afterwards, the PHS RBF-FD method is also investigated for 2D BVPs of linear elasticity—applying the uniform, the quasi-uniform, and the newly-developed *h*-adaptive point-cloud refinements. Section 6 summarizes the results and suggests possible extensions of the presented approach.

2 Basic system of differential equations in plane elasticity

The second-order partial differential equations for the 2D elastostatic problem are expressed in a Cartesian coordinate frame as

$$
\begin{bmatrix} \mu\,\mathcal{L}_{\Delta} + (\mu+\lambda)\,\mathcal{L}_{xx} & (\mu+\lambda)\,\mathcal{L}_{xy} \\ (\mu+\lambda)\,\mathcal{L}_{xy} & \mu\,\mathcal{L}_{\Delta} + (\mu+\lambda)\,\mathcal{L}_{yy} \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix} + \begin{bmatrix} f_x \\ f_y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad (1)
$$

where u_x, u_y, and f_x, f_y are the coordinates of the displacement- and body force density vectors $\boldsymbol{u} = u_x\,\boldsymbol{i} + u_y\,\boldsymbol{j}$

and $\boldsymbol{f} = f_x \, \boldsymbol{i} + f_y \, \boldsymbol{j}$. Further, μ and λ denote the Lamé coefficients, while $\mathcal{L}_\Delta$ as well as $\mathcal{L}_{xx}$, $\mathcal{L}_{yy}$, and $\mathcal{L}_{xy}$ are, respectively, the Laplacian as well as the pure and mixed 2D second-order partial differential operators—defined as

$$
\begin{aligned}
\mathcal{L}_{xx} &:= \frac{\partial^2}{\partial x^2} \,, \\
\mathcal{L}_{yy} &:= \frac{\partial^2}{\partial y^2} \,, \\
\mathcal{L}_{xy} &:= \frac{\partial^2}{\partial x \, \partial y} \,, \\
\mathcal{L}_\Delta &:= \mathcal{L}_{xx} + \mathcal{L}_{yy} \,.
\end{aligned}
\tag{2}
$$

The coefficient matrix in Eq. (1) can be rearranged in the form

$$
\begin{aligned}
\begin{bmatrix} \mu \, \mathcal{L}_\Delta + (\mu + \lambda) \, \mathcal{L}_{xx} & (\mu + \lambda) \, \mathcal{L}_{xy} \\ (\mu + \lambda) \, \mathcal{L}_{xy} & \mu \, \mathcal{L}_\Delta + (\mu + \lambda) \, \mathcal{L}_{yy} \end{bmatrix} &= \\
\mu \, \mathcal{L}_\Delta \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + (\mu + \lambda) \begin{bmatrix} \mathcal{L}_{xx} & \mathcal{L}_{xy} \\ \mathcal{L}_{xy} & \mathcal{L}_{yy} \end{bmatrix}
\end{aligned}
\tag{3}
$$

The Neumann-type boundary conditions prescribed on the surface part $S_p \subset \partial\Omega$ are expressed in terms of the first-order partial derivatives of the displacements u_x and u_y:

$$
\begin{aligned}
p_x &= n_x \, (2\mu + \lambda) \, \mathcal{L}_x u_x + n_y \, \mu \, \mathcal{L}_x u_y \\
&\quad + n_x \, \lambda \, \mathcal{L}_y u_y + n_y \, \mu \, \mathcal{L}_y u_x \\
p_y &= n_y \, (2\mu + \lambda) \, \mathcal{L}_y u_y + n_x \, \mu \, \mathcal{L}_y u_x \\
&\quad + n_y \, \lambda \, \mathcal{L}_x u_x + n_x \, \mu \, \mathcal{L}_x u_y
\end{aligned}
\quad \text{on } S_p \,,
\tag{4}
$$

using the operator-notations

$$
\mathcal{L}_x := \frac{\partial}{\partial x} \,, \qquad \mathcal{L}_y := \frac{\partial}{\partial y} \,.
\tag{5}
$$

Here, $\partial\Omega = S_p \cup S_u$ is the boundary of the body for which $S_p \cap S_u = \emptyset$ is valid, while n_x and n_y denote the coordinates of the outward unit normal vector $\boldsymbol{n} = n_x \, \boldsymbol{i} + n_y \, \boldsymbol{j}$ to $\partial\Omega$. The Dirichlet-type boundary condition

$$
\boldsymbol{u} = \tilde{\boldsymbol{u}} = \tilde{u}_x \, \boldsymbol{i} + \tilde{u}_y \, \boldsymbol{j} \qquad \text{on } S_u
$$

is prescribed on the remaining boundary part of the solid body $S_u \subset \partial\Omega$. The Lamé coefficient λ corresponds to

$$
\begin{aligned}
&\frac{E\nu}{(1+\nu)(1-2\nu)} && \text{if plane strain} \\
&\frac{E\nu}{(1+\nu)(1-\nu)} && \text{if plane stress}
\end{aligned}
$$

conditions are considered. Here, λ converges to ∞, as ν tends to $\frac{1}{2}$ in the plain strain case, i.e., this model problem will be prone to volumetric locking effects.

3 Computing the differentiation weights using the augmented PHS

In this section, our aim is to approximate the first- and second-order partial derivative operators $\mathcal{L}$ introduced in Eqs. (2) and (5) at the evaluation point $\boldsymbol{r}_e$. The derivative $\mathcal{L}u$ at $\boldsymbol{r}_e$ is approximated by a linear combination of the function values $u_i = u(\boldsymbol{r}_i)$ considered at the distinct points $\boldsymbol{r}_i \in \mathbb{R}^d$ along with the evaluation point $\boldsymbol{r}_e$, where $i = 1, \ldots, n$ and d is the space dimension. These points are used to construct the local stencil. Accordingly, we seek the weights w_i as the coefficients of u_i in such a way that

$$
[\mathcal{L}u(\boldsymbol{r})]_{\boldsymbol{r}=\boldsymbol{r}_e} \approx \sum_{i=1}^{n} w_i \, u_i \,,
\tag{6}
$$

in which n and w_i are called the stencil size and the differentiation weights, respectively. In order to determine w_i, we start with a linear combination

$$
s(\boldsymbol{r}) = \sum_{j=1}^{n} k_j \, \phi(\|\boldsymbol{r} - \boldsymbol{r}_j\|) + \sum_{j=1}^{\binom{p+d}{p}} \gamma_j \, P_j(\boldsymbol{r})
\tag{7}
$$

along with the constraints

$$
\sum_{i=1}^{n} k_i \, P_j(\boldsymbol{r}_i) = 0 \,, \quad j = 1, \ldots, \binom{p+d}{p} \,,
\tag{8}
$$

where $P_j(\boldsymbol{r}_i)$ denotes multivariate polynomials up to degree p and $\phi(\|\boldsymbol{r} - \boldsymbol{r}_i\|)$ are the RBFs localized at the points $\boldsymbol{r}_i$, [1, 13,16,30]. Here, $\|\boldsymbol{r} - \boldsymbol{r}_i\| = \|\boldsymbol{r} - \boldsymbol{r}_i\|_2$ denotes the Euclidean norm on $\mathbb{R}^d$. For the mathematical background of the related quadratic constrained minimization problem, we refer to [2, 4].

To derive the equation for the differentiation weights we apply the interpolant (7) to the function values $u_i = u(\boldsymbol{r}_i)$ and consider the $\binom{p+d}{p}$ constraints (8). Next, we use again the interpolant (7) but now for the approximation of the derivative $[\mathcal{L}u(\boldsymbol{r})]_{\boldsymbol{r}=\boldsymbol{r}_e}$ at the evaluation point $\boldsymbol{r}_e$. Then we eliminate the function values u_i with the assumption that k_i and γ_i are non-zero coefficients arriving at the following linear equation system

$$
\begin{bmatrix} \underline{\mathbf{K}} & \underline{\mathbf{Q}} \\ \underline{\mathbf{Q}}^T & \underline{\mathbf{O}} \end{bmatrix} \begin{bmatrix} \underline{\mathbf{w}} \\ \underline{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} \underline{\boldsymbol{\ell}} \\ \underline{\mathbf{q}} \end{bmatrix}
\tag{9}
$$

which can be solved for the weights w_i and v_j, for details see [1,14]. In Eq. (9)

$$
\underline{\mathbf{K}} = \begin{bmatrix} \phi(\|\boldsymbol{r}_1 - \boldsymbol{r}_1\|_2) & \cdots & \phi(\|\boldsymbol{r}_1 - \boldsymbol{r}_n\|_2) \\ \vdots & \ddots & \vdots \\ \phi(\|\boldsymbol{r}_n - \boldsymbol{r}_1\|_2) & \cdots & \phi(\|\boldsymbol{r}_n - \boldsymbol{r}_n\|_2) \end{bmatrix} \quad \text{and}
$$

$$\underline{\mathbf{Q}} = \begin{bmatrix} P_1(\mathbf{r}_1) & \cdots & P_{\binom{p+d}{p}}(\mathbf{r}_1) \\ \vdots & \ddots & \vdots \\ P_1(\mathbf{r}_n) & \cdots & P_{\binom{p+d}{p}}(\mathbf{r}_n) \end{bmatrix}$$

represent the symmetric RBF matrix $\underline{\mathbf{K}} \in \mathbb{R}^{n \times n}$ and the rectangular polynomial matrix $\underline{\mathbf{Q}} \in \mathbb{R}^{n \times \binom{p+d}{p}}$, respectively, while

$$\underline{\boldsymbol{\ell}} = \left[\left[\mathcal{L}\phi(\|\mathbf{r} - \mathbf{r}_1\|_2) \right]_{\mathbf{r}_e} \cdots \left[\mathcal{L}\phi(\|\mathbf{r} - \mathbf{r}_n\|_2) \right]_{\mathbf{r}_e} \right]^T$$

and

$$\underline{\mathbf{q}} = \left[\left[\mathcal{L}P_1 \right]_{\mathbf{r}_e} \cdots \left[\mathcal{L}P_{\binom{p+d}{p}} \right]_{\mathbf{r}_e} \right]^T$$

are the differentiation vectors $\underline{\boldsymbol{\ell}} \in \mathbb{R}^n$ and $\underline{\mathbf{q}} \in \mathbb{R}^{\binom{p+d}{p}}$, as well as

$$\underline{\mathbf{w}} = \begin{bmatrix} w_1 \cdots w_n \end{bmatrix}^T \quad \text{and} \quad \underline{\mathbf{v}} = \begin{bmatrix} v_1 \cdots v_{\binom{p+d}{p}} \end{bmatrix}^T$$

denote the weight vectors $\underline{\mathbf{w}} \in \mathbb{R}^n$ and $\underline{\mathbf{v}} \in \mathbb{R}^{\binom{p+d}{p}}$. Please note that the weights $\underline{\mathbf{v}}$ are omitted, for details see [1,14,16].

The widely-used RBFs are classified into two main groups: the infinitely and piecewise smooth RBFs, see the tables in [1,13,16]. The infinitely smooth RBFs depend on the shape parameter ϵ, influencing the gradient of the RBFs and thereby controlling the condition number of the coefficient matrix in Eq. (9) and thus the accuracy of the result for the differentiation weights. On the contrary, the piecewise smooth PHS is independent of ϵ and provides higher convergence rates under node refinement, without the use of a stable algorithm to invert the RBF matrix, see [1–4,18]. This is the main reason and motivation why the PHS is chosen for the numerical computation of the weights, supplemented with a polynomial basis for the sake of further stabilization and faster convergence.

The RBF-FD methods are well-applicable if the coefficient matrix appearing in Eq. (9) exhibits non-singular behavior, i.e., the system (9) has a unique solution for the choice of the odd m-order PHS

$$\phi[r(x, y)] = r^m = \sqrt{\left[(x - x_e)^2 + (y - y_e)^2\right]^m}, \qquad (10)$$

in which $\mathbf{r}_e = x_e \mathbf{i} + y_e \mathbf{j}$ represents the Cartesian coordinates of the evaluation point of the stencil, supplemented with polynomials up to degree p where

$$m \le 2p + 1. \qquad (11)$$

If, for instance, $m = 3$ is chosen, the degree of the polynomial basis has to be at least 1—or degree p has to be at

least 2 for $m = 5$, see [1]. The stencil size n is chosen to be approximately twice the number of the polynomial basis functions $\binom{p+d}{p}$, which is $p + 1$ in 1D and $(p + 2)(p + 1)/2$ in the 2D case. The increase in stencil size n has only small or no influence on the accuracy and convergence rate for a fixed polynomial degree p, see [4].

4 Solution procedure

4.1 Point cloud and stencil geometry

In what follows, the applied point-cloud will be structured as Cartesian or, in other words, as a rectangular point distribution for the convergence studies of the uniform h-refinement, see Figs. 1 and 7. These Cartesian arrangements are similar to the standard grid-like FD schemes, but the RBF interpolant defined in Eqs. (7)–(8) is of course quite different. Next to the boundaries, the stencil becomes half-sided and non-symmetric. Although the evaluation point comes closer to the boundaries in this case the stencils retain the rectangular shape. However, inside the domain, the stencils remain symmetric and their evaluation point is localized at their center, see Figs. 1 and 7.

Next, we consider Neumann-type boundary conditions (BC). Let us assume that the Neumann-type BC (4) and its 1D version (15), respectively, are prescribed on the right boundary $x = 2$ and on the top boundary $y = 1$ for the 2D case (Fig. 7), as well as at the right boundary $x = L$ for the 1D case (Fig. 1).

In order to satisfy these BCs at a given boundary point, a new point has to be added to the point-cloud outside the domain as a neighbor of the boundary point under consideration. These external points are called *ghost points*. They allow for the addition of constraint equations to the global system of equations arising from the numerical approximation of the 2D PDE (1) or its 1D version (13). The subsidiary equations are required for the enforcement of the Neumann-type BC (4) (or (15) for the 1D case) prescribed at the corresponding boundary points, see [14].

As a result of the latter procedure, new unknown function values have to be introduced at the external points for u in 1D and for u_x and u_y in 2D. After the global assembly process, the resulting linear algebraic system of equations is simultaneously solved for all of the unknown function values located at the external, internal, and boundary points. The total number of the points N in the point-cloud is given as the sum of the number of the external, internal, and boundary points, N_E, N_I, and N_B:

$$N = N_E + N_I + N_B. \qquad (12)$$

As described above, the number of the extra constraint equations associated with the enforcement of the Neumann-type BC is equal to $d \cdot N_E$, where d denotes the space dimension, see [14].

4.2 *h*-adaptive point-cloud refinement strategy

In this subsection, we propose a new adaptive point-cloud refinement scheme for the PHS-RBF method to solve 1D and 2D problems of linear elasticity. In what follows, the *h*-adaptive algorithm will be demonstrated for 1D problems. This type of BVP can be considered as the 1D version of the problem presented in Sect. 2. The basic equation and the boundary conditions can be stated as follows, see [11,39]:

$$-(AEu'(x))' = f(x) \quad \text{in } \Omega = [x|x \in (0, L)] , \tag{13}$$

$$u = \tilde{u} \quad \text{at } x = 0 , \tag{14}$$

$$AEu' = \tilde{F} \quad \text{at } x = L . \tag{15}$$

Here, $f(x)$ represents as a distributed line load, A the cross-sectional area, E the Young's modulus, and L the length of the bar—while $\tilde{u}$ and $\tilde{F}$ are the prescribed values of the axial displacement at $x = 0$ and the axial force at $x = L$, respectively.

The point refinement procedure is guided by an error indicator based on the residual of the second-order differential equation (13), see Algorithm 1. At the very beginning of the numerical process, we set the initial number of points N_0, the number of the iteration steps, the order of the PHS m, and the degree of the supplementary polynomials p, as well as the material parameters and the loading values. Next, the structured point-cloud $\underline{\mathbf{x}}_0$ is generated.

In each iteration step, the problem has to be solved. To this end, the weights for the approximation of $\mathcal{L}_x$ and $\mathcal{L}_{xx}$ at the internal and boundary points are computed by solving the local system (9) and assembling the global first- and second-order differential operator matrices $\underline{\mathbf{D}}_x$, $\underline{\mathbf{D}}_{xx}$ and the global equation system $AE \underline{\mathbf{D}}_{xx} \underline{\mathbf{u}} = \underline{\mathbf{f}}$. Next, Dirichlet-type boundary conditions (14) are applied in strong form, and Neumann-type boundary conditions (15) as new constraint equation are considered. Then, the resulting system is solved in order to determine $\underline{\mathbf{u}} \in \mathbb{R}^N$. Afterwards, the column matrix of the axial force $\underline{\mathbf{F}}$ is computed as $AE \underline{\mathbf{D}}_x \underline{\mathbf{u}}$. By means of the first-order differential-operator matrix $\underline{\mathbf{D}}_x$, we compute the numerical approximation of the right-hand side of Eq. (13) as $\hat{\underline{\mathbf{f}}} = \underline{\mathbf{D}}_x \underline{\mathbf{F}}$. In this way, the residual is expressed in the form

$$\underline{\boldsymbol{\eta}} = \hat{\underline{\mathbf{f}}} - \underline{\mathbf{f}} . \tag{16}$$

In a next step, the absolute value of the residual is checked for two neighboring points within a 'for' loop over all the internal and boundary points. A new point is inserted between the two neighboring points in question if the refinement criterion

$$|\eta_i| > \beta/N \, \eta_{\max} \tag{17}$$

holds true for one of the two investigated points. Here, $\eta_{\max}$ is the maximum norm of the column matrix of the global residual as

$$\eta_{\max} = \max_{1 \leq i \leq N} |\eta_i| \tag{18}$$

whereas N is the total number of points and $\beta \in [0, 1]$ denotes the threshold parameter. The choice of this tolerance value controls how many points are added to the point-cloud in each iteration step. The *h*-adaptive refinement terminates if it meets the stopping criterion, which can be set for both the number of the iterations or the maximum of the residual. During our numerical tests, the value β will be set to 0.01 for 1D BVPs and to 0.1 for 2D BVP. Algorithm 1 summarizes the *h*-adaptive point-cloud refinement scheme.

5 Numerical experiments

In this section, the performance of the PHS-based RBF-FD method augmented with a polynomial basis will be analyzed for both smooth and non-smooth BVPs of 1D and 2D linear elasticity. During the computational tests, the convergence of the relative error measured in ℓ_2-norm of the displacements and stresses will be studied. The number of degrees of freedom (denoted as DOF in the figure captions) is defined as the total number of points N of the point-cloud, ignoring the constraints related to the Dirichlet BC.

Additionally, the influence of the polynomial degree p on the rate of *h*-convergence will be investigated in each problem. In order to avoid the singular behavior of PHS when $x = x_e$ and $y = y_e$ for a second-order operator, the PHS order m has to be chosen high enough. In what follows we set m to 3 for 1D and $m = 5$ for 2D BVPs. Therefore, the smallest possible polynomial degree p, on account of inequality (11), is 1 in 1D and 2 in 2D.

5.1 One-dimensional boundary value problems

In this subsection, the computational performance of the PHS-based RBF-FD method will be analyzed for the 1D problem presented above. For the sake of simplicity, the geometric parameters A and L, as well as Young's modulus E, will be assumed to be constant. In view of this, the values $AE = 1$ N and $L = 1$ m will be assumed in all of the considered 1D benchmark problems.

During the *h*-convergence study, the domain will be refined equidistantly in 10 steps for smooth and non-smooth

Algorithm 1 Adaptive point-cloud refinement strategy

1: **procedure** INPUT VALUES:$(m, p, N_0, \tilde{F}, \tilde{u}, A, E)$
2:    Compute $n = 2p + 5$ ▷ stencil size
3:    Generate $\underline{\mathbf{x}}_0$, using N_0 ▷ initial uniform point-cloud
4:    Set $iterstep$ ▷ maximum number of iterations
5:    Set β ▷ tolerance value for the error
6:    Set $iter = 1$ ▷ initial value of the iteration number
7:    **repeat**
8:        Compute the weights for $\mathcal{L}_x$ and $\mathcal{L}_{xx}$ at the internal and boundary points
9:        $AE \, \underline{\mathbf{D}}_{xx} \, \underline{\mathbf{u}} = \underline{\mathbf{f}}$ ▷ assemble the global equation system Eq. (13)
10:        Apply the BCs   ▷ $\tilde{u}$ and $\tilde{F}$ according to Eqs. (14)–(15)
11:        Solve the resulting linear algebraic equation system for the column matrix $\underline{\mathbf{u}}$
12:        $\underline{\mathbf{F}} = AE \, \underline{\mathbf{D}}_x \, \underline{\mathbf{u}}$ ▷ compute the column matrix of the internal force $\underline{\mathbf{F}}$
13:        $\hat{\underline{\mathbf{f}}} = \underline{\mathbf{D}}_x \, \underline{\mathbf{F}}$ ▷ compute the right-hand side column matrix $\hat{\underline{\mathbf{f}}}$
14:        $\underline{\boldsymbol{\eta}} = \hat{\underline{\mathbf{f}}} - \underline{\mathbf{f}}$ ▷ compute the column matrix of the residual
15:        $\eta_{max} = \max |\eta_i|$ ▷ determine the maximum of the residual according to Eq. (18)
16:        $N_{new} = 0$ ▷ set the initial number of the newly-inserted points to zero
17:        **for** $i = 1$ to N (loop over all the internal and boundary points) **do**
18:            Check whether a new point needs to be put between two neighboring points
19:            **if** $(\eta_{i+1}/(\eta_{max}/N_{new}) > \beta$ or $\eta_i/(\eta_{max}/N_{new}) > \beta)$ **then**
20:                $x_{new} = (x_{i+1} + x_i)/2$ ▷ put a point between the points $i + 1$ and i
21:                $N_{new} = N_{new} + 1$ ▷ increase the number of the points by one
22:            **else**
23:                Do nothing.
24:            **end if**
25:        **end for**
26:        $N = N + N_{new}$ ▷ update the total number of the points
27:        $iter = iter + 1$ ▷ increment the iteration number
28:    **until** $((\eta_{max}/N < \beta)$ or $(iter > iterstep))$
29: **end procedure**

(such as singular- and shock) BVPs, while the degree of the polynomial basis p is fixed in each step. In order to investigate the effect of the polynomial degree p on the rate of the h-convergence, p will be varied from 1 to 7 and 2 to 5, respectively, for smooth and non-smooth problems. Therefore, the odd stencil size n has to be adjusted to the actual p-value as $n = 2p + 5$.

The efficiency of the newly developed adaptive point refinement strategy will also be investigated by comparing its convergence behavior to the numerical results obtained using equidistant point sets.

The symmetric stencil geometry applied within the domain and the half-sided, non-symmetric stencil shapes used near to the boundaries and at the boundaries are illustrated for the lowest degree case in Fig. 1. Here, the polynomial degree is set to $p = 0$—and the stencil size inside the domain is $n = 2p + 5 = 5$ while the stencil bandwidth can

be determined as $b = (n - 1)/2$. The total number of points of the uniformly structured point-cloud is set to $N = 20$.

5.1.1 Smooth problem

In the first example, let the distributed line load be chosen as the smooth function $f(x) = \alpha(\alpha - 1)x^{\alpha-2}$, where $\alpha = 10$. Furthermore, the homogeneous BCs $\tilde{u} = 0$ m and $\tilde{F} = 0$ N are given, respectively, at the ends $x = 0$ and $x = L$. In this case, the exact solution is a smooth function, i.e. $u_{EX} = -x^\alpha + \alpha x$.

The h-convergence curves obtained for the relative error in ℓ_2-norm of the axial displacement u and force F are plotted against the number of degrees of freedom on log-log scales in Fig. 2. In addition, their rates are also indicated by the slopes. Accordingly, algebraic convergence rates are observed for all refinements. The order of the convergence $O(h^g)$ depends on the polynomial degree p, closely correlated to the theoretical estimate

$$g \simeq p - k + 1, \tag{19}$$

see [4,14,28]. Herein, k denotes the differential order of the considered BVP ($k = 2$).

As expected, higher convergence rates are exhibited for higher-degree polynomial bases. For higher-degrees ($p = 6$ and $p = 7$) however, the stagnation—or, in other words, saturation error—emerges at a certain (but very low) level of the relative error, see [1]. For lower polynomial degrees, the stagnation error is reached at a relatively high error level only with the use of a much larger number of points on the same domain, see [1]. In this example, however, the stagnation error does not yet appear for lower-degree polynomials because the number of points applied is relatively low. Besides, it follows from Eq. (19) that the h-convergence is lost for $p = 1$, as shown in Fig. 2 as well. The convergence rates are very close to the estimation (19), see also the results for the numerically approximated Laplacian of a smooth function in [14].

5.1.2 Singular problem

In the second example, we consider again a line load $f(x) = \alpha(\alpha - 1)x^{\alpha-2}$—but now in such a way that the exact solution $u_{EX} = -x^\alpha + \alpha x$ is non-smooth. Here, the parameter α controls the smoothness of the solution. If α is small, i.e., $\alpha < 1$, then the first derivative of the exact solution will become singular at $x = 0$, see [11]. Thus, we set α to 0.55 in order to study the convergence for a problem with a singular solution. The homogeneous Dirichlet- and Neumann-type BCs, $\tilde{u} = 0$ m and $\tilde{F} = 0$ N, are specified, respectively, at the boundaries $x = 0$ and $x = L$.

Fig. 1 Symmetric and half-sided stencil shape representations near to the boundaries (below), at the boundaries (above), and inside the domain (above in red) for the lowest polynomial degree $p = 0$
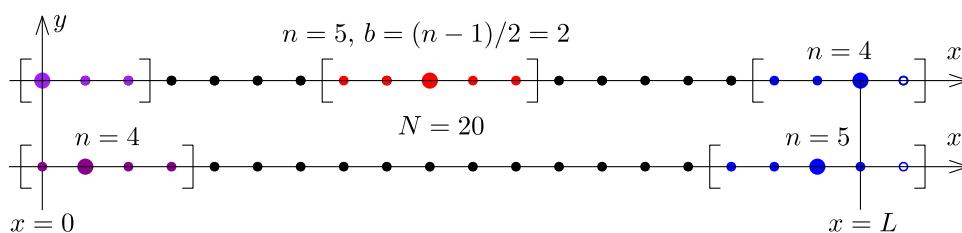
Fig. 2 Convergence curves for the relative error in ℓ_2-norm of the displacement u and the axial force F—uniform point cloud for smooth problem
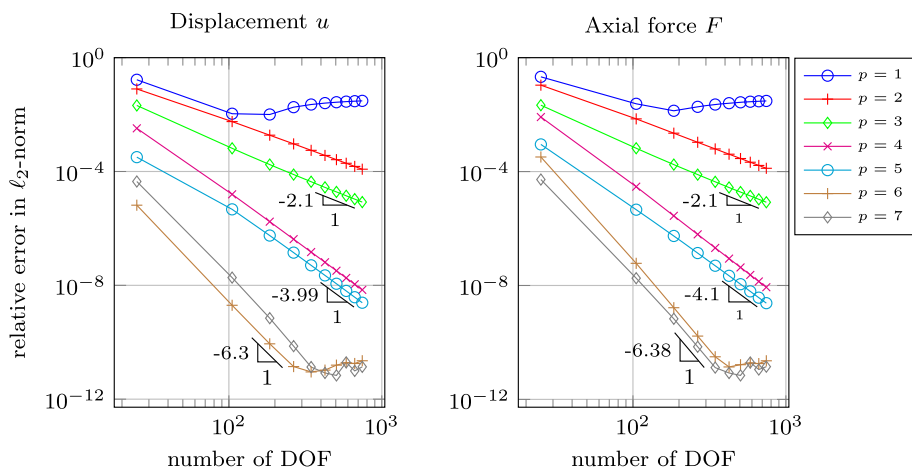
Again, the h-convergence curves of the relative ℓ_2-error are plotted as the function of the number of degrees of freedom on log-log scales for the displacement u and the axial force F in Fig. 3. The convergence rates are indicated by the corresponding slopes.

The results obtained for $p = 1$ are not shown since we already observed (in the previous example) that the p-value has to be equal to 2 or higher, as predicted by the theoretical estimate.

In the case of uniform point-cloud refinement, the h-convergence in the displacement error computations is of an algebraic type with the same rate for all polynomial degrees. Concerning the convergence of the axial force, we also observe an algebraic type of convergence with the same rate for different polynomial degrees. The convergence curves are shifted down without changing their slopes as the degree p is increased indicating a lower error constant.

Next, we compare the h-adaptive refinement to the uniform approach. To this end, different polynomial degrees are compared, see Fig. 3. We start the refinement with a uniformly distributed point-cloud with $N_0 = 25$. It can be observed that the h-adaptive algorithm helps a lot to improve the convergence rates, thereby increasing the accuracy at any considered p-level, not only in the displacement but also in the axial force. A comparison of the numerical and exact solution is given for the 6th iteration step in Fig. 4 for the highest considered polynomial degree $p = 5$. In this iteration step, the relative ℓ_2-error of the numerical solution computed for the displacement u by the h-adaptive algorithm is already

less than 3 %. This error level is much lower than what was achieved by means of the uniform point-distribution.

5.1.3 Shock problem

In the third example, we consider a shock problem with the exact solution

$$u_{EX} = \arctan[\alpha(x - 1/\pi)] \tag{20}$$

which has a high gradient within the domain at $x = 1/\pi$, controlled by the parameter α, see [11,32]. Here, α will be set to 120, and $f(x)$ will be defined accordingly. Non-homogeneous Dirichlet-type BCs are prescribed at both ends of the domain, i.e., at $x = 0$ and $x = L$, consistent with the exact solution (20).

As in the previous examples, the relative error measured in ℓ_2-norm is plotted in Fig. 5 for both the displacement and the axial force computed with a uniform h-refinement.

In the pre-asymptotic range, it is possible to observe an exponential type of convergence that slows down to an algebraic rate. The convergence rate indicated by the measured slope is almost independent of the chosen polynomial degree.

Henceforward, the adaptive PHS RBF-FD method is studied in 5 refinement steps for $p = 2, 3, 4, 5$. The results of the adaptive approach are compared to the uniform refinement in Fig. 5. As expected, h-adaptivity significantly improves the convergence rates. The desired accuracy is achieved with much less degrees of freedom for any polynomial degree p, both in the displacement and in the axial force.

Fig. 3 Convergence curves for the relative error in ℓ_2-norm of the displacement u and the axial force F—h-adaptive versus uniform point-cloud refinement for singular problem, $p = 2$, $p = 3$, $p = 4$ and $p = 5$, number of the iteration steps: 10, $\beta = 1\%$
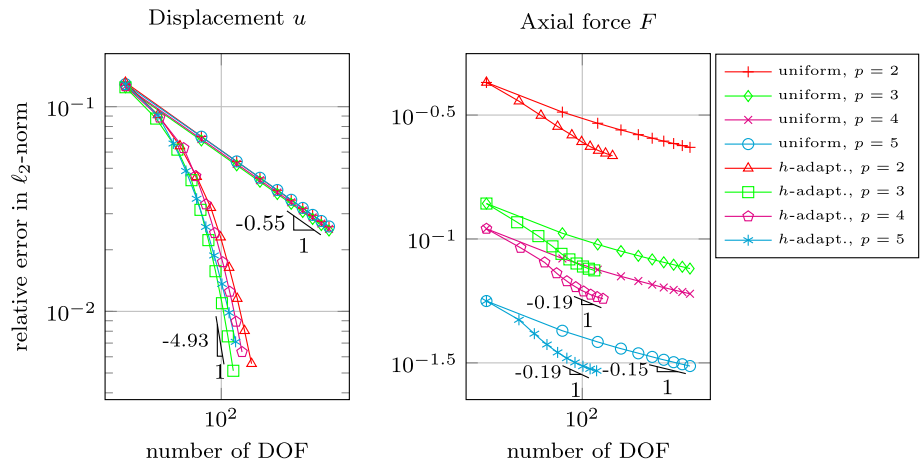
Fig. 4 Displacement u and axial force F along x—exact and approximate solutions of the singular problem, using adaptive and uniform refinement in the 6th iteration step for $p = 5$
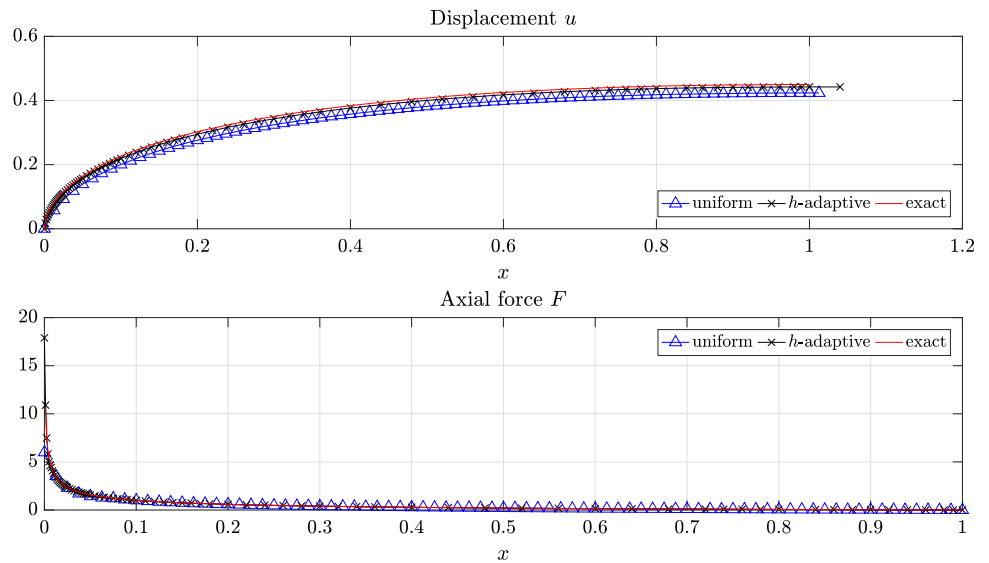

Fig. 5 Convergence curves for the relative error in ℓ_2-norm of the displacement u and the axial force F—h-adaptive versus uniform point-cloud refinement for shock problem, $p = 2$, $p = 3$, $p = 4$, and $p = 5$, number of the adaptive iteration steps: 5, $\beta = 1\%$
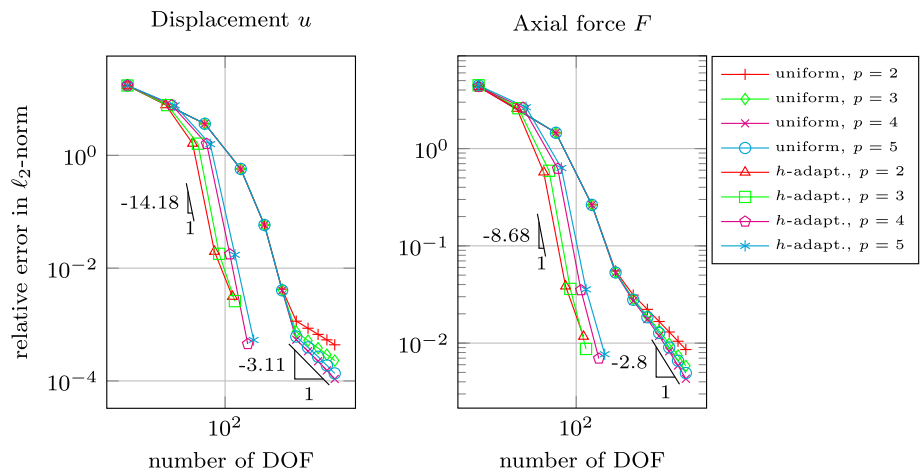
Fig. 6 Displacement u and axial force F along x—exact and approximate solutions of the shock problem, using h-adaptive and uniform refinement in the 4th iteration step for $p = 5$
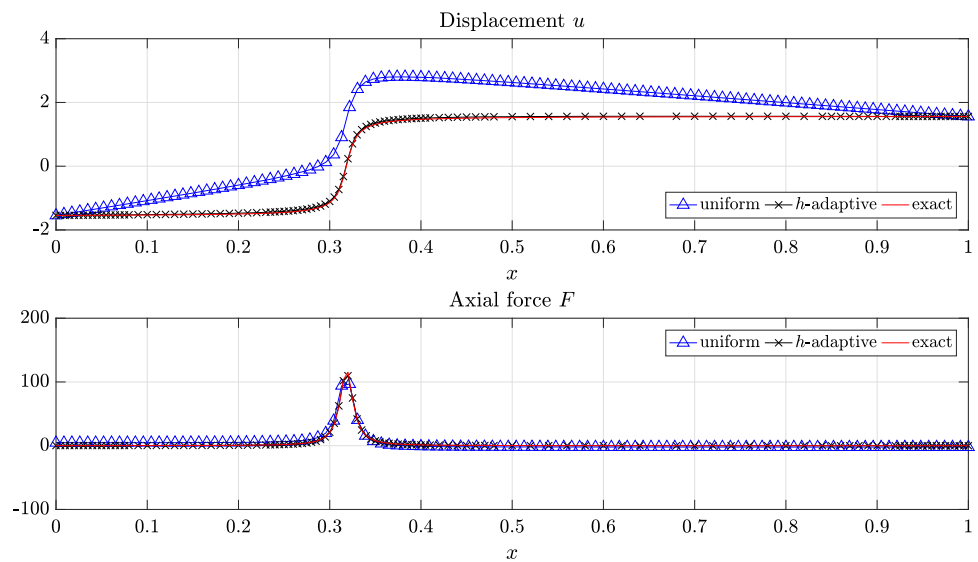
To illustrate the numerical approach, the results of the 4th refinement step are once again compared to the uniform approach for the highest investigated polynomial degree $p = 5$, see Fig. 6. In this adaptive refinement step, the relative ℓ_2-error of the displacement u and the axial force F is already less than 4 %. Again, the accuracy of the adaptive approach is much higher than that of the uniform refinement.

5.2 Two-dimensional boundary value problems

In this section, the convergence behavior of the PHS-based RBF-FD method will be investigated for several representative smooth and non-smooth 2D BVPs of linear elasticity. During the convergence studies, the domain in question is uniformly or quasi-uniformly refined in 6 steps, from 2601 up to 8281 points for all of the benchmark problems. In each step, the degree of the polynomial basis p is kept fixed. In order to observe the influence of the polynomial degree, p will be varied from 2 to 5. Hence, the odd stencil size n has to be adjusted to the actual p-level as $n = [(p+1)(p+2)+5]^2$ within the domain. The 2D version of the h-adaptive strategy presented in Algorithm 1 will also be tested on the "shock problem" as non-smooth 2D BVP, and its computational performance will be studied.

As an example for the definition of the stencil in the case of uniform point-cloud refinement, a square domain $(x, y) \in [1, 2] \times [1, 2]$ discretized with a uniformly distributed set of points is considered, see Fig. 7. Different stencils are presented: the symmetric square stencil shape used inside the domain and the half-sided non-symmetric stencil applied next to the boundaries and at the boundaries.

In this example, for the sake of illustration, the stencil size n is set to $5 \cdot 5$ within the domain. The equidistantly structured point set involves $N = N_I + N_B + N_E = 15 \cdot 15 + 2 \cdot 15 = 255$
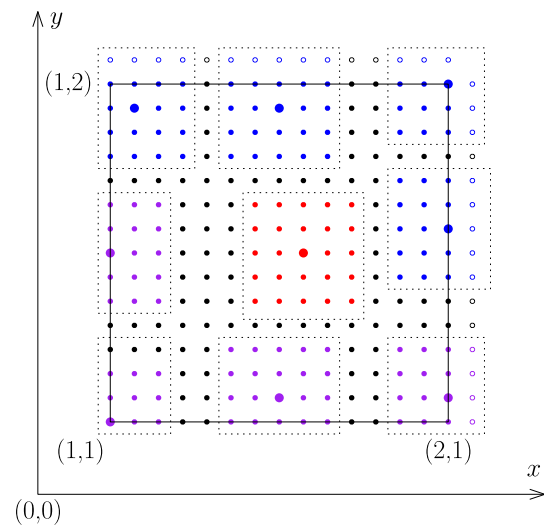
Fig. 7 Cartesian stencils near to the boundaries, at the boundaries, and inside the domain—uniformly structured point-cloud

points on the basis of Eq. (12), where $N_I + N_B = 225$ and $N_E = 30$.

### 5.2.1 Approximation of the two-dimensional differential operators

In the first 2D example, the partial derivatives introduced in Eqs. (2) and (5) are approximated with the RBF-FD method, based on the PHS supplemented with polynomials applying the point-cloud defined on the square domain $(x, y) \in [1, 2] \times [1, 2]$. Having determined the differential operator matrices $\underline{\mathbf{D}}_x$, $\underline{\mathbf{D}}_y$, and $\underline{\mathbf{D}}_{xx}$, $\underline{\mathbf{D}}_{yy}$, as well as $\underline{\mathbf{D}}_{xy}$, they are applied to the given smooth test function

$$u(x, y) = 1 + \sin 4x + \cos 3x + \sin 2y , \qquad (21)$$

Fig. 8 Convergence histories for the relative error in ℓ_2-norm of approximating the first-order differential operators $\mathcal{L}_x$ and $\mathcal{L}_y$ acting on the smooth function u
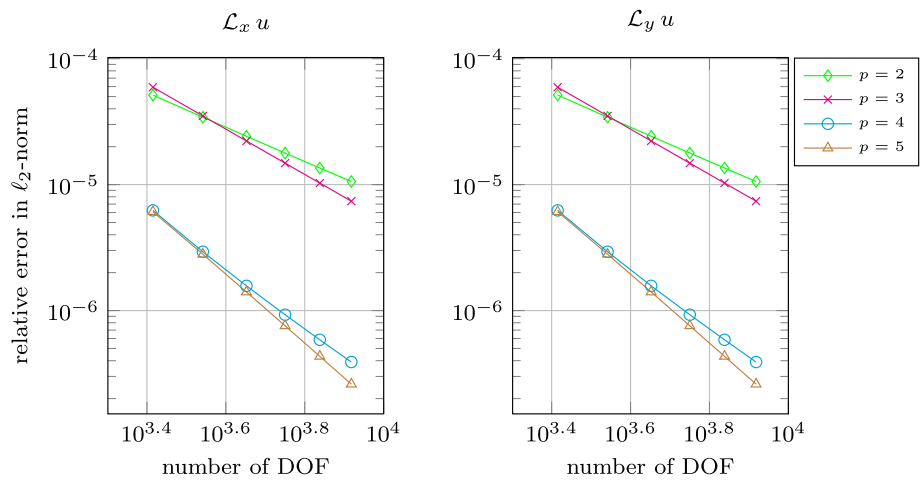
Fig. 9 Convergence histories for the relative error in ℓ_2-norm of approximating the second-order differential operators $\mathcal{L}_{xx}$ and $\mathcal{L}_{yy}$ acting on the smooth function u
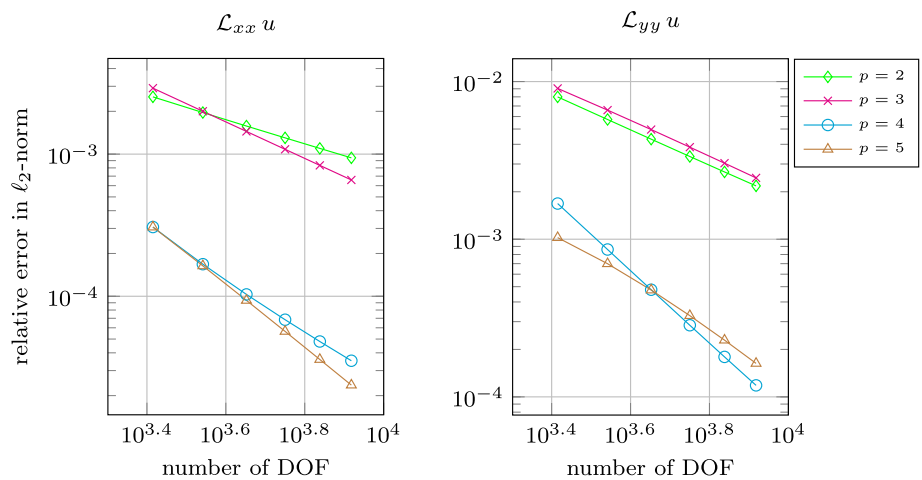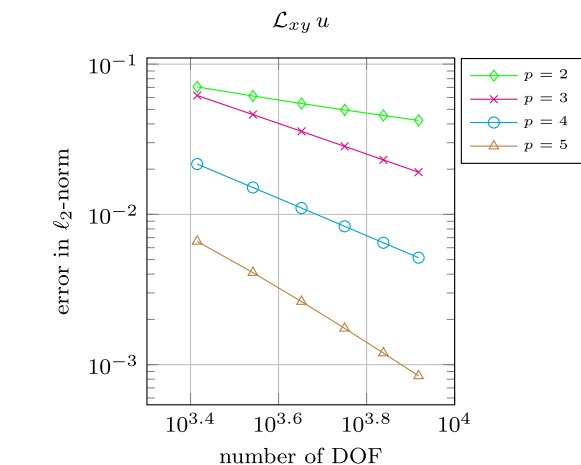

to compute the derivatives for all points of the structured point-cloud. The ℓ_2-error of the approximation of the first- and second-order partial differential operators, $\mathcal{L}_x$, $\mathcal{L}_y$, and $\mathcal{L}_{xx}$, $\mathcal{L}_{yy}$, $\mathcal{L}_{xy}$ applied to the smooth function (21) is plotted on a double logarithmic scale against the number of degrees of freedom in Figs. 8, 9 and 10.

From this, it is evident that the rate of convergence depends on the degree of the polynomial basis p. A higher degree leads to a high algebraic rate of convergence when increasing the number of points. Furthermore, it can be observed that the first-order derivatives can be approximated more accurately as compared to the second-order derivatives.

Fig. 10 Convergence histories for the ℓ_2-error-norm of approximating the second-order mixed differential operator $\mathcal{L}_{xy}$ acting on the smooth function u

5.2.2 Smooth problem

Next, the convergence behavior of the PHS RBF-FD method for a unit square plate under linear elastic plane stress conditions is studied. Dirichlet-type BCs are imposed on all of the four boundaries. The body force densities f_x, f_y and the BCs correspond to the smooth "manufactured" solution:

$$u_x = -\frac{1}{10} \cos \pi (x-1) \sin \pi (y-1),$$

$$u_y = \frac{1}{10} \sin \frac{\pi (x-1)}{7} \sin \frac{\pi (y-1)}{3}.$$

Fig. 11 Convergence curves for the relative error in ℓ_2-norm of the displacements u_x and u_y—smooth problem
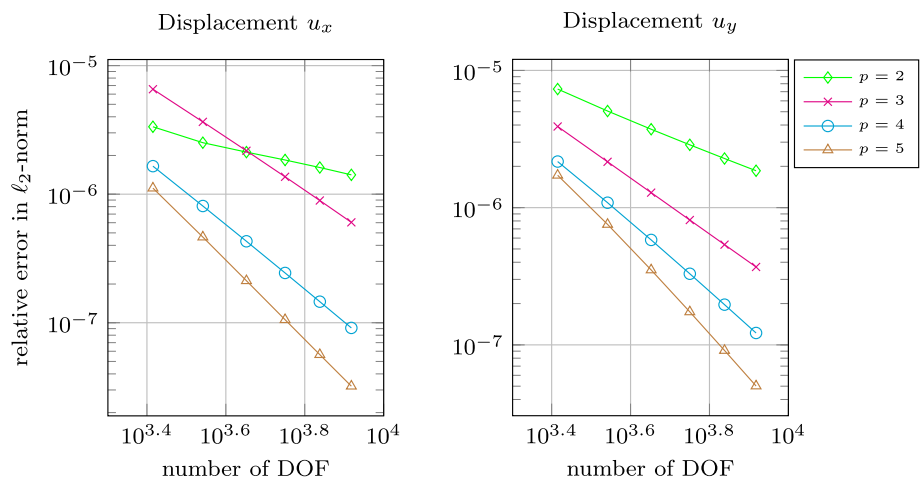
Fig. 12 Convergence curves for the relative error in ℓ_2-norm of the stress coordinates σ_x, σ_y, and τ_{xy}—smooth problem
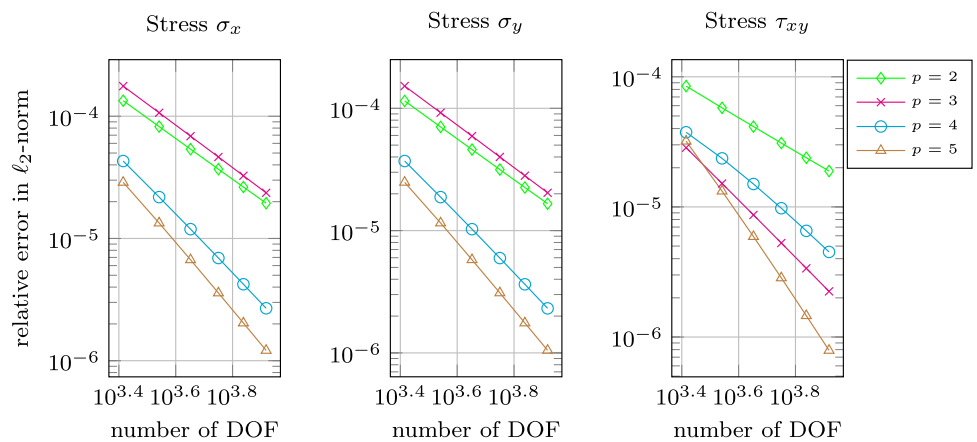
In Figs. 11 and 12 the relative error in ℓ_2-norm of the displacements u_x and u_y and the stresses σ_x, σ_y, and τ_{xy} is plotted versus the number of degrees of freedom in double logarithmic style. The stresses are computed by post-processing the first-order partial derivatives of the displacements:

$$\sigma_x = (2\mu + \lambda)\,\underline{\mathbf{D}}_x\,\underline{\mathbf{u}}_x + \lambda\,\underline{\mathbf{D}}_y\,\underline{\mathbf{u}}_y$$
$$\sigma_y = (2\mu + \lambda)\,\underline{\mathbf{D}}_y\,\underline{\mathbf{u}}_y + \lambda\,\underline{\mathbf{D}}_x\,\underline{\mathbf{u}}_x \quad \text{in } \Omega \,,$$
$$\tau_{xy} = \mu \left(\underline{\mathbf{D}}_y\,\underline{\mathbf{u}}_x + \underline{\mathbf{D}}_x\,\underline{\mathbf{u}}_y \right)$$

where the differential operator matrices $\underline{\mathbf{D}}_x$ and $\underline{\mathbf{D}}_y$ include the weights for the numerical approximation of the partial derivatives $\mathcal{L}_x$ and $\mathcal{L}_y$ at the internal and boundary points, while the column matrices $\underline{\mathbf{u}}_x$ and $\underline{\mathbf{u}}_y$ consist of the previously-computed displacements u_x and u_y, respectively. From the computational results, it can be seen that algebraic convergence rates are obtained when increasing the number of points. Again, raising the degree of the additional polynomials improves the convergence rate.

A contour plot of the numerical approximation is compared to the exact solution for the displacements and stresses in Figs. 13 and 14. The numerical solution involves $N =$ $59 \cdot 59$ points, a stencil size of $n = 17 \cdot 17$, and a polynomial degree of $p = 2$. This comparison demonstrates the good agreement of the numerical approximation with the exact solution.

5.2.3 Shock problem

In our next example, the PHS RBF-FD method will be investigated for the 2D version of the "shock problem" introduced in Sect. 5.1.3. Again, the linear elasticity equation system (2) is numerically solved on a unit square domain, imposing non-homogeneous Dirichlet-type BCs on all parts of the boundary. The body force densities f_x, f_y and the BCs correspond to the manufactured solution

$$u_x = \arctan \alpha \left(\sqrt{(x - 3c)^2 + (y - c)^2} - r_0 \right),$$
$$u_y = \arctan \alpha \left(\sqrt{(x - c)^2 + (y - c)^2} - r_0 \right),$$

where $c = 0.75$ m, $r_0 = 1$ m and $\alpha = 60$ [11,32].

Since—from an engineering point of view—a sufficiently accurate numerical approximation of the stresses is more of

Fig. 13 Numerical and exact solutions for the displacements u_x and u_y on the undeformed domain—smooth problem, $p = 2$, $n = 17 \cdot 17$, and $N = 59 \cdot 59$
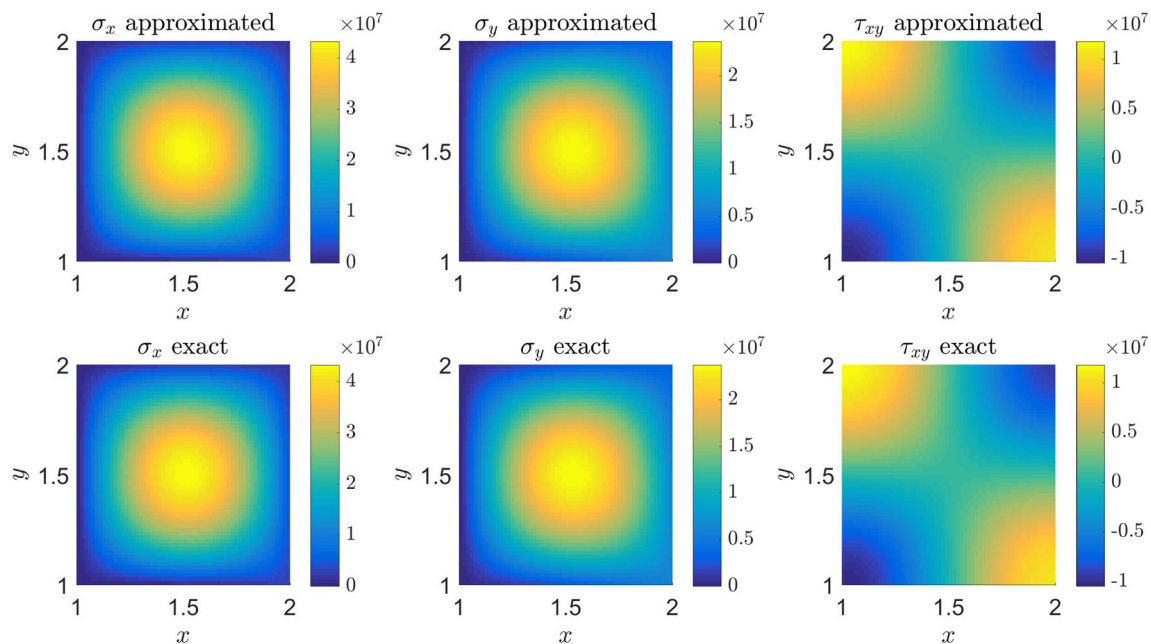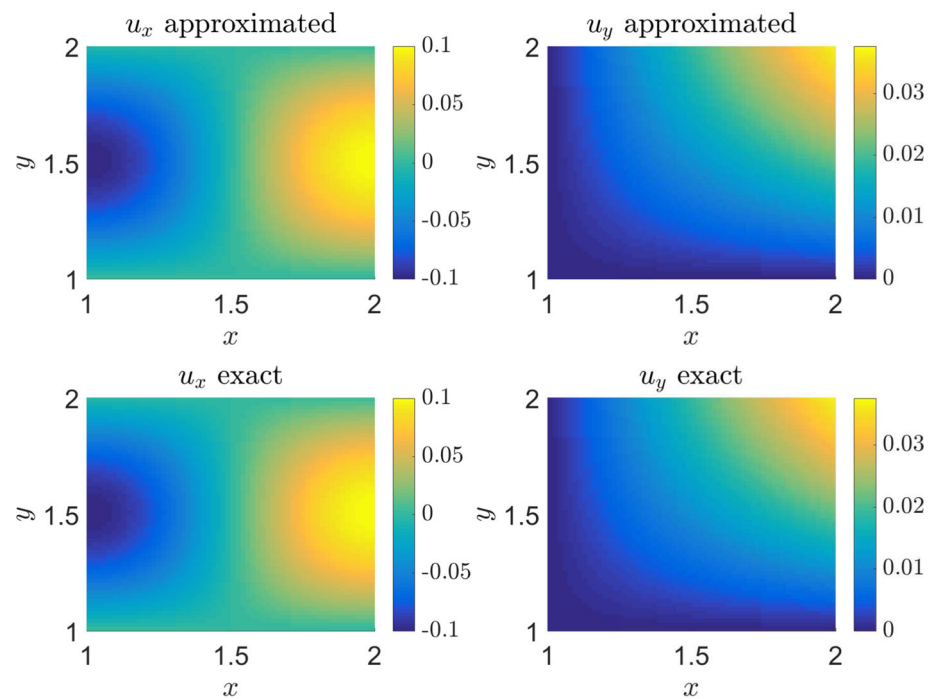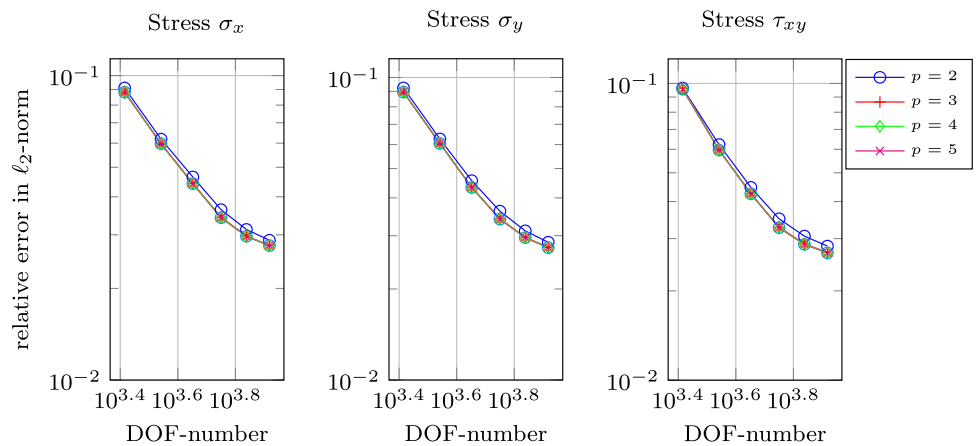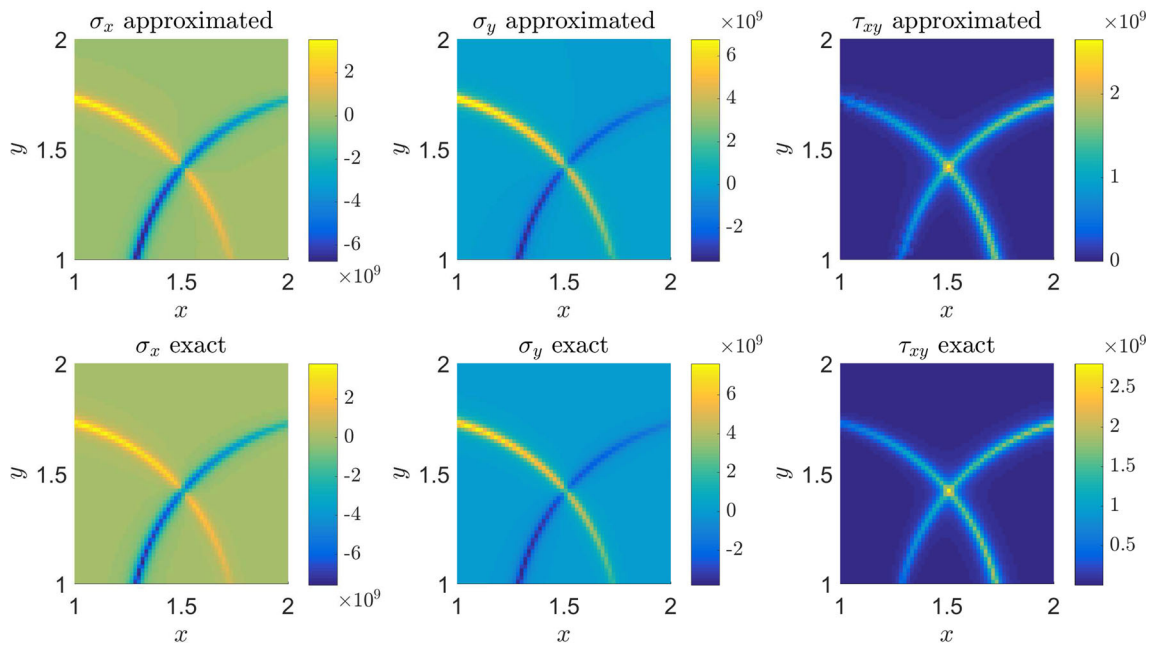

Fig. 14 Numerical and exact solutions for stress coordinates σ_x, σ_y, and τ_{xy} on the undeformed domain—smooth problem, $p = 2$, $n = 17 \cdot 17$, and $N = 59 \cdot 59$

interest than that of the displacements, our numerical investigations are focused on the h-convergence behavior of the stresses. Firstly, in Fig. 15 the relative errors of the uniform h-refinement, measured in ℓ_2-norm, are plotted for the stresses σ_x, σ_y and τ_{xy} against the number of degrees of freedom in a double logarithmic scale. The relative ℓ_2-error achieved in the 6th refinement step is already below an acceptable level (3 %).

Contour plots of the displacements and stresses are compared to the exact solution in Figs. 16 and 17. The numerical results were obtained with a uniform point-cloud consisting of $N = 59 \cdot 59$ points, applying a stencil size of $n = 17 \cdot 17$ and a polynomial degree of $p = 2$.

Next, the h-adaptive refinement strategy is compared to the uniform approach by extending and applying Algorithm 1 to the numerical solution of 2D BVPs. In the uniform as

Fig. 15 Convergence curves for the relative error in ℓ_2-norm of the stress coordinates σ_x, σ_y and τ_{xy}—shock problem

Fig. 16 Numerical and exact solutions for the displacements u_x and u_y on the undeformed domain—shock problem, $p = 2, n = 17 \cdot 17$, and $N = 59 \cdot 59$
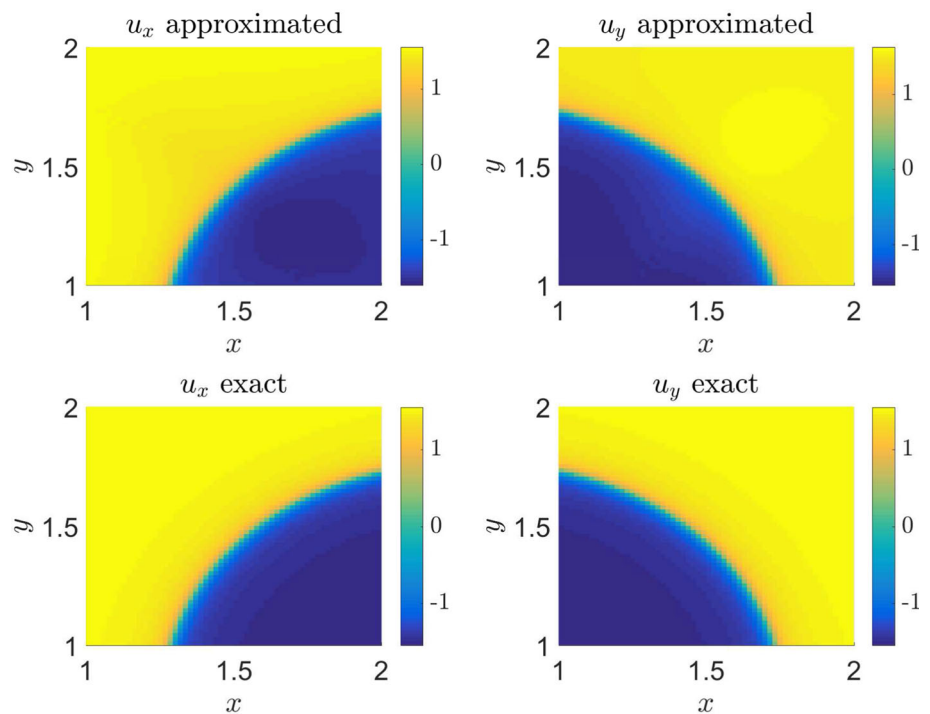
well as adaptive approach, the refinement procedure is started with a quite coarse resolution, i.e., a uniformly distributed point-cloud containing $7 \cdot 7 = 49$ points, keeping the polynomial degree $p = 2$ and the stencil size $n = 17$ fixed at each refinement step. However, during the h-adaptive refinement, the stencils become arbitrary-shaped because the knnsearch algorithm is used to find the n-nearest neighbors to the evaluation point of the stencils. Similarly to the 1D BVPs (see Algorithm 1), the h-adaptivity is driven by the following process: The absolute value of the residual is checked for three neighboring points and—when considering the boundary—for just two neighboring points. A new point is inserted in the center of the neighboring points in question if the refinement criterion (17) along with the definition (18) holds true for one the investigated points.

For quasi-uniform and uniform point-clouds, the convergence behavior of the PHS-RBF methods can be investigated by plotting the relative error curves against either the distance $h = 1/\sqrt{N}$ or against N. The convergence rate can be estimated for smooth BVPs by (19). However, for strongly non-uniform point-clouds, the relation between h and N is problem-dependent and the convergence rate can not be determined straightforwardly in each case. Therefore, for the h-adaptive point-cloud refinement strategy and thus also for this 2D comparative study, it is worth measuring the global convergence of relative errors as the function of the so-called effective fill-distance $h_{\text{eff}} = h_\Omega/N$, where the fill distance

$$h_\Omega = \sup_{\boldsymbol{r} \in Y} \min_{\boldsymbol{r}_i \in \Omega} \|\boldsymbol{r} - \boldsymbol{r}_i\|_2 \,,$$

Fig. 17 Numerical and exact solutions for stress coordinates σ_x, σ_y, and τ_{xy} on the undeformed domain—shock problem, $p = 2$, $n = 17 \cdot 17$, and $N = 59 \cdot 59$

Fig. 18 Convergence curves for the relative error in ℓ_2-norm of the stress coordinates σ_x, σ_y and τ_{xy}—h-adaptive versus uniform refinement, $p = 2$, $n = 17$, number of the iteration steps: 6, $\beta = 10\%$
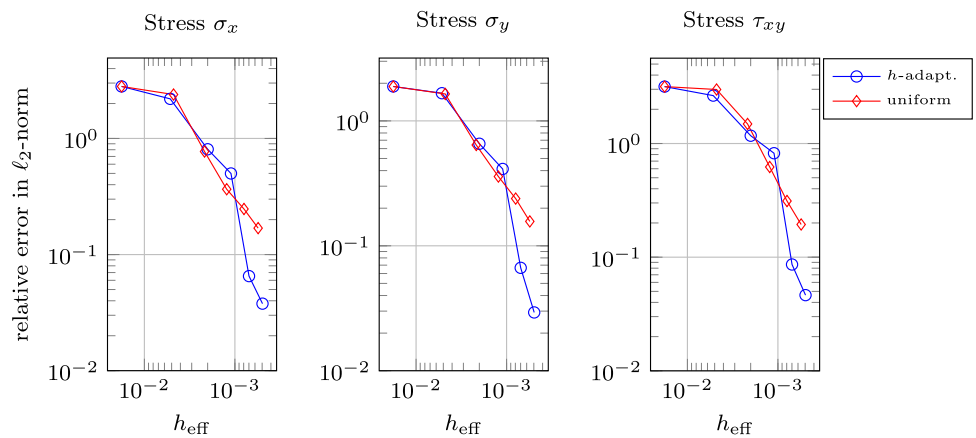

Fig. 19 Numerical solutions for the displacements u_x and u_y on the adaptive point-cloud, using h-adaptive refinement in the 6th iteration step—shock problem, $p = 2$ and $n = 17$
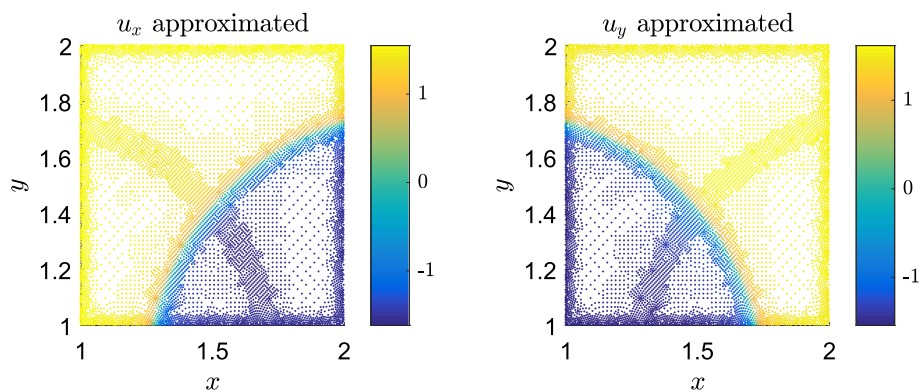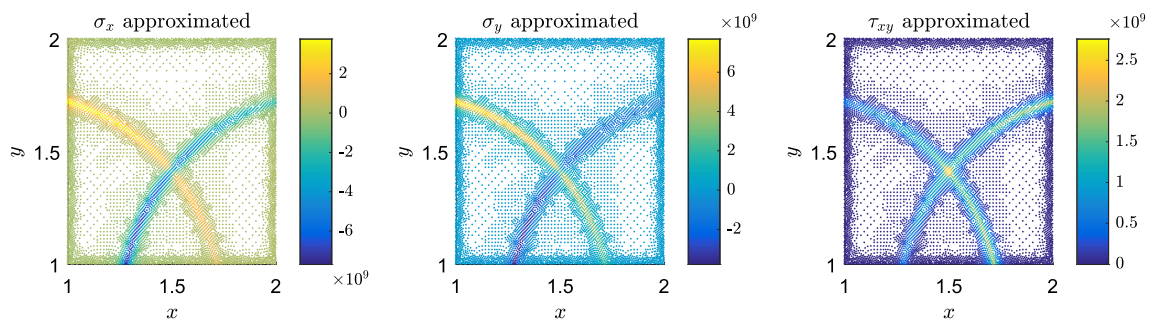
Fig. 20 Numerical solutions for stress coordinates σ_x, σ_y, and τ_{xy} on the adaptive point-cloud, using h-adaptive refinement in the 6th iteration step—shock problem, $p = 2$ and $n = 17$

Fig. 21 Computational domain (**a**) and initial point-cloud (**b**) of the model problem—unstressed circular hole in an infinite solid subjected to the uniaxial tension σ_∞ in direction x

(a) Computational domain

(b) Initial point-cloud
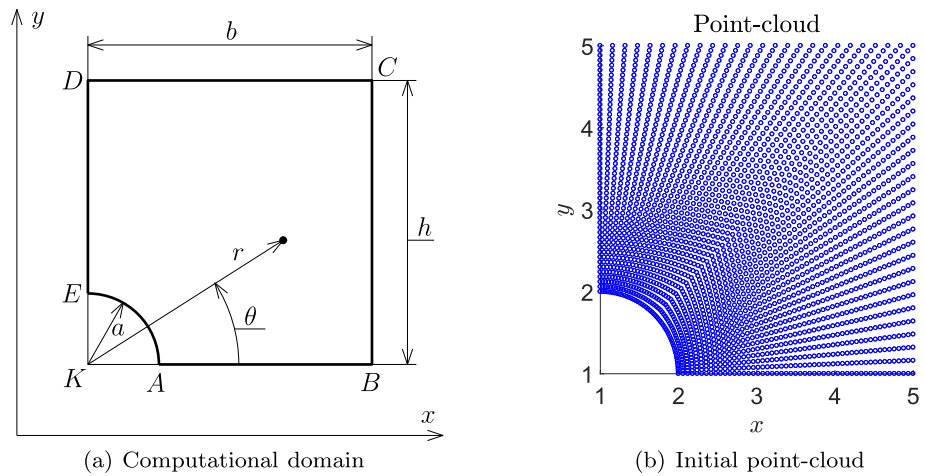
Fig. 22 Convergence curves for the relative error in ℓ_2-norm of the displacements u_x and u_y—circular hole in an infinite solid
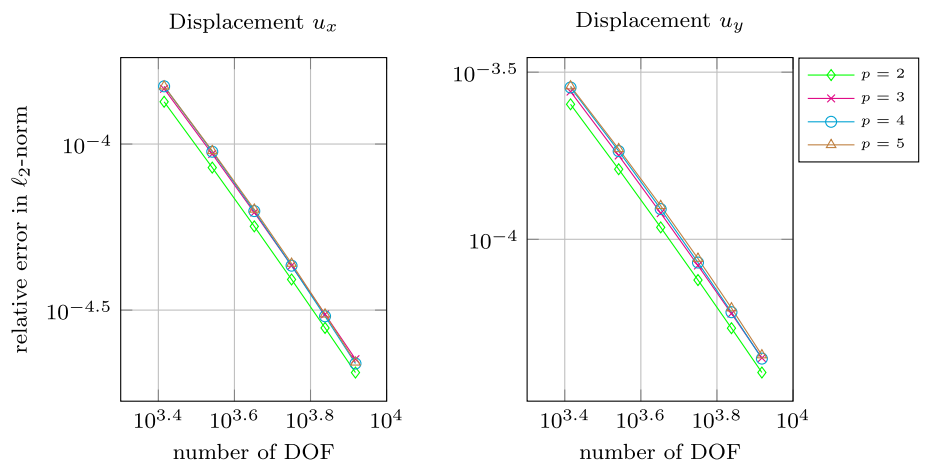
defined as a "mesh size", indicates how well the points in the point-cloud of Ω cover the domain $Y \subseteq \Omega$ and represent the radius of the largest empty circle that can be placed among the point locations inside Y [30]. For this reason, the convergence series obtained for the stresses are plotted against the more representative h_{eff} in Fig. 18.

It can also be observed that the 2D version of the h-adaptive scheme presented in Algorithm 1 helps a lot to increase both the convergence rates and the accuracy.

To illustrate the h-adaptive refinement approach, the numerical results of the 6th iteration step for the displacements and stresses are plotted on the adaptive point-cloud in Figs. 19 and 20. In this iteration step, the relative ℓ_2-error of the numerical solution computed by the h-adaptive algorithm is already less than 5 %. This error level is lower than what was achieved using the equivalent uniform point-distribution. As a consequence of these results, the extension and the application of this h-adaptive algorithm to the numerical solution of 3D BVPs is the aim of our future research work.

Fig. 23 Convergence curves for the relative error in ℓ_2-norm of the stress coordinates σ_x, σ_y, and τ_{xy}—circular hole in an infinite solid
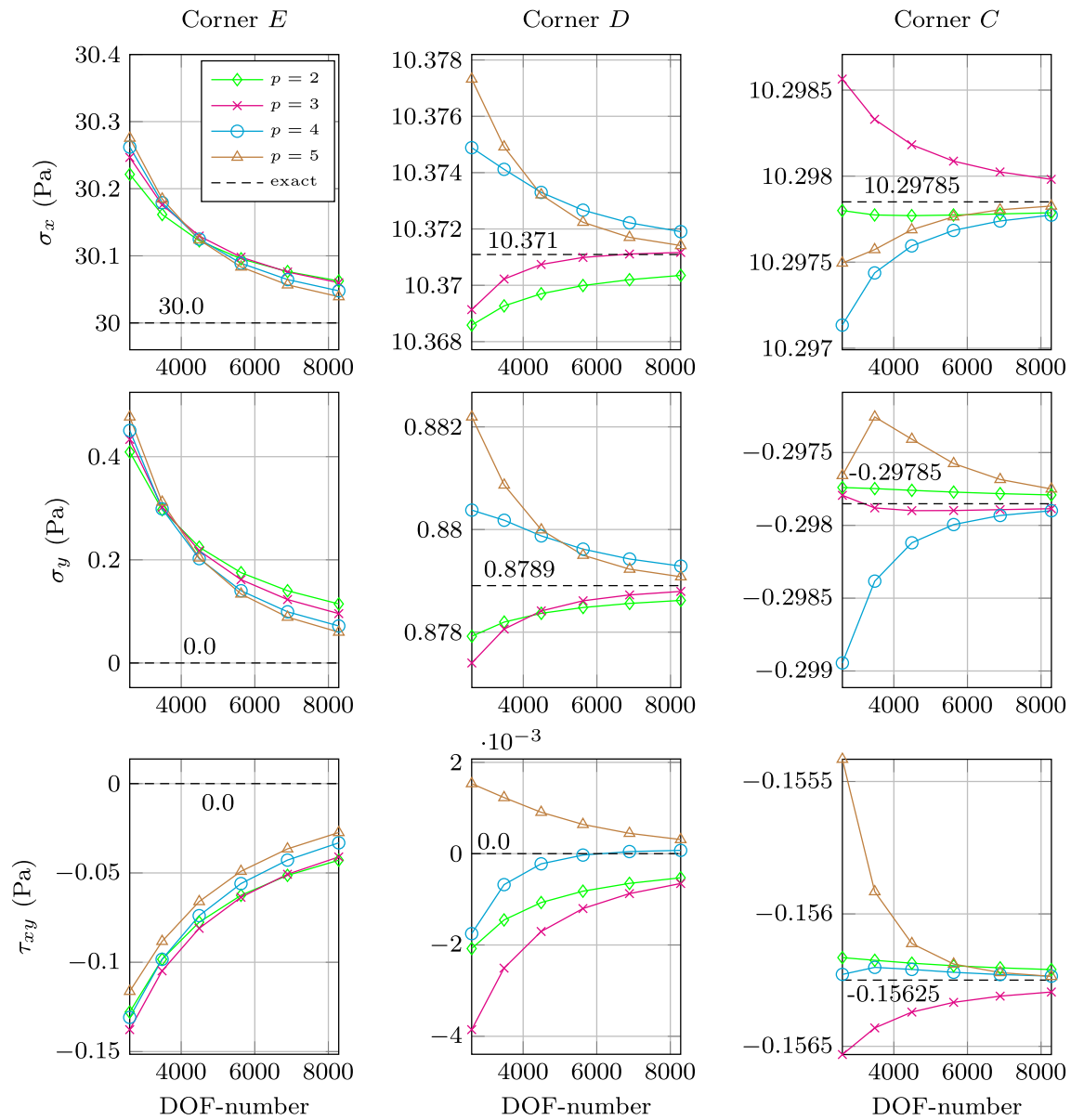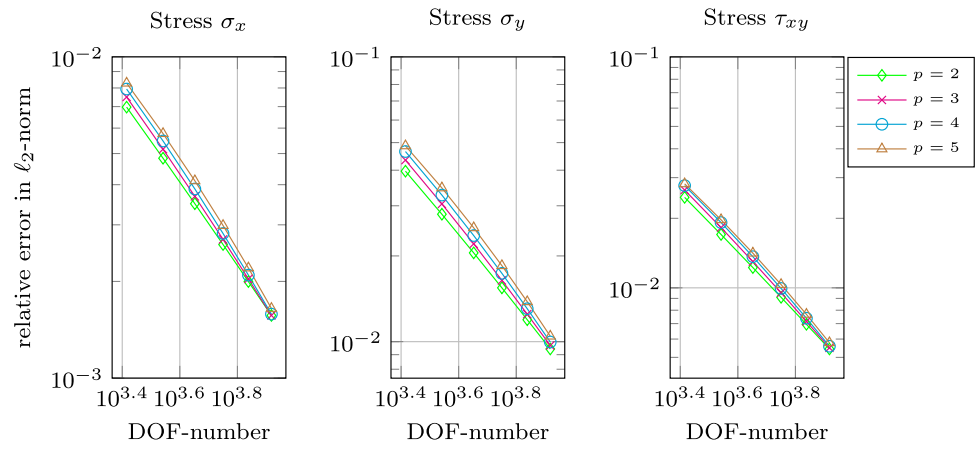
Fig. 24 Convergence of the stress coordinates σ_x, σ_y, and τ_{xy} at the corners C, D, and E—unstressed circular hole in an infinite solid

Fig. 25 Convergence of the stress coordinates σ_x, σ_y, and τ_{xy} at the corners A and B—circular hole in an infinite solid
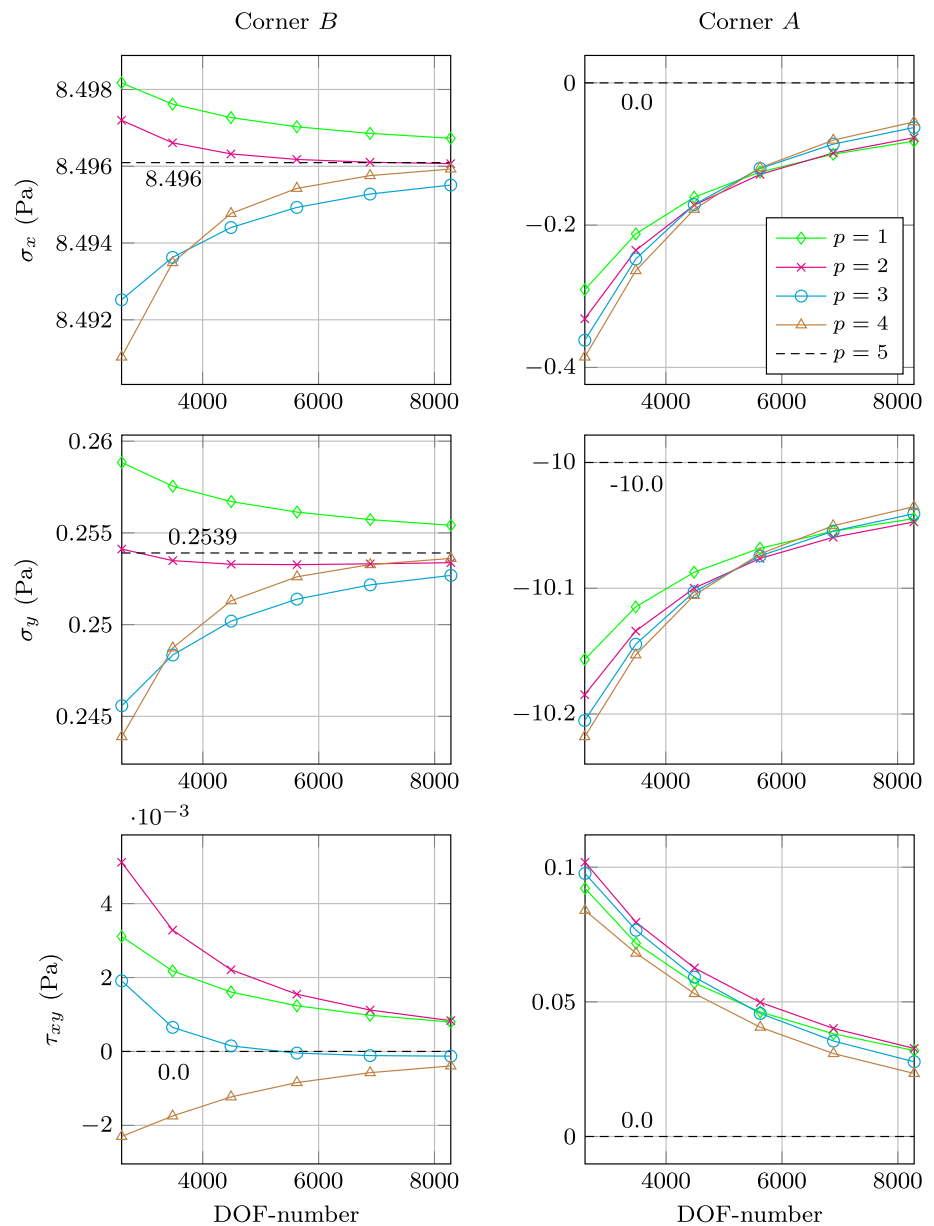
5.2.4 Unstressed circular hole in an infinite plate

Finally, the classical 2D linear elasticity problem of the unstressed circular hole in an infinite plate exposed to unidirectional tension in the xy plane will be studied as a smooth benchmark problem [9,39,40]. Due to symmetry, only one quarter of the plate is investigated. The resulting computational domain is depicted in Fig. 21a.

The exact solution reads

$$u_x = \frac{a\sigma_\infty}{8\mu}\left[\frac{r}{a}\left(\kappa + 1\right)\cos\theta + 2\frac{a}{r}\left((1+\kappa)\cos\theta + \cos 3\theta\right) \right.$$
$$\left. -2\frac{a^3}{r^3}\cos 3\theta\right], \tag{22}$$

$$u_y = \frac{a\sigma_\infty}{8\mu}\left[\frac{r}{a}\left(\kappa - 3\right)\sin\theta + 2\frac{a}{r}\left((1-\kappa)\sin\theta + \sin 3\theta\right) \right.$$
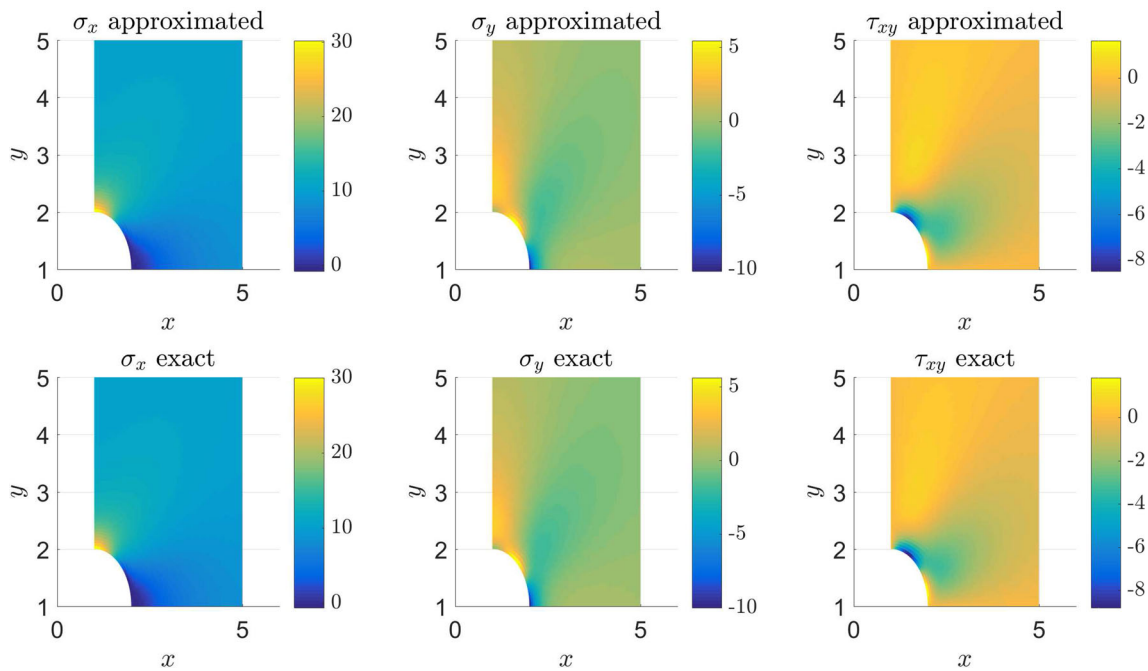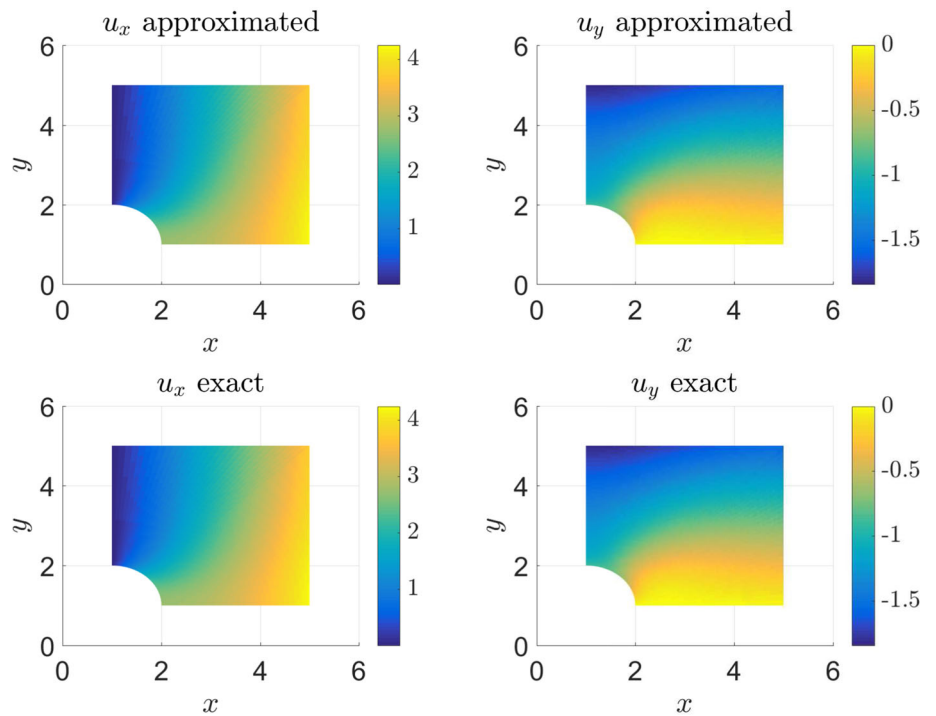$$\left. -2\frac{a^3}{r^3}\sin 3\theta\right] \tag{23}$$

for the displacements and

$$\sigma_x = \sigma_\infty$$
$$\left[1 - \frac{a^2}{r^2}\left(\frac{3}{2}\cos 2\theta + \cos 4\theta\right) + \frac{3a^4}{2r^4}\cos 4\theta\right], \tag{24}$$

$$\sigma_y = \sigma_\infty$$
$$\left[-\frac{a^2}{r^2}\left(\frac{1}{2}\cos 2\theta - \cos 4\theta\right) - \frac{3a^4}{2r^4}\cos 4\theta\right], \tag{25}$$
$$\sigma_z = \nu\left(\sigma_x + \sigma_y\right)$$

Fig. 26 Numerical and exact solutions for the displacements u_x and u_y on the undeformed domain—circular hole in an infinite solid, $p = 2$, $n = 17 \cdot 17$ and $N = 59 \cdot 59$

u_x approximated

u_y approximated

u_x exact

u_y exact

σ_x approximated

σ_y approximated

τ_{xy} approximated

σ_x exact

σ_y exact

τ_{xy} exact

Fig. 27 Numerical and exact solutions for stress coordinates σ_x, σ_y, and τ_{xy} on the undeformed domain—circular hole in an infinite solid, $p = 2$, $n = 17 \cdot 17$ and $N = 59 \cdot 59$

$$= \nu\sigma_\infty \left(1 - \frac{2a^2}{r^2}\cos 2\theta\right),$$

$$\tau_{xy} = \sigma_\infty$$

$$\left[-\frac{a^2}{r^2}\left(\frac{1}{2}\sin 2\theta + \sin 4\theta\right) + \frac{3a^4}{2r^4}\sin 4\theta\right] \quad (26)$$

for the stresses, where σ_∞ denotes the applied uniaxial stress, a is the radius of the unstressed circular hole, and

$$r = \sqrt{(x-1)^2 + (y-1)^2} \quad \text{and} \quad \theta = \arctan\frac{y-1}{x-1}$$

are the polar coordinates. Plain strain conditions are assumed with an isotropic linear elastic behavior, based on Young's modulus $E = 10$ Pa and Poisson's ratio $\nu = 0.3$, resulting in $\kappa = 3 - 4\nu$. The width and height of the plate are set to $b = h = 4a$ with $a = 1$ m. Symmetry boundary conditions are imposed on the edges AB and DE. Dirichlet BCs are prescribed along the edges BC, CD, and AE, according to the exact solution.

Not only the global but also the local convergence behavior of the RBF-FD method based on the PHS supplemented with a polynomial basis will be tested on this benchmark problem. As a first result, the relative error measured in ℓ_2-norm is plotted for the displacements u_x, u_y and for the stresses σ_x, σ_y, τ_{xy} in Figs. 22 and 23. The uniform refinement starts with the point-cloud shown in Fig. 21b. Again, different values for the polynomial degree p are taken into account.

As it is evident from the figures, this setting leads to an algebraic type of convergence that is independent of the degree p.

As a second result, the PHS-based RBF-FD solutions obtained for the stresses σ_x, σ_y and τ_{xy} at the corners A, B, C, D, and E are depicted together with their exact values are plotted in Figs. 24 and 25. From these figures, quite a fast convergence of pointwise quantities can be observed.

Finally, we present contour plots of the displacements and stresses together with their exact distribution in Figs. 26 and 27. The computations are based on the quasi-uniform point set involving $N = 59 \cdot 59$ points, applying a stencil size of $n = 17 \cdot 17$ points together with a polynomial basis of degree $p = 2$.

6 Conclusions

A new error-controlled h-adaptive RBF-FD method was developed for the solution of BVPs in linear elasticity. The PHS supplemented with polynomials was chosen as basis.

Firstly, the computational performance of the method was tested through convergence studies of several representative 1D and 2D smooth and non-smooth BVPs. The relative errors were determined in the ℓ_2-norm. It was demonstrated that fast convergence can be obtained by refining the point-cloud, provided that the polynomial degree of the additional polynomials is high enough.

Secondly, the convergence behavior of the newly-developed adaptive point-cloud refinement algorithm was verified in 1D for both the singular and the shock problem, as well as in 2D for the shock problem. The presented h-adaptive scheme leads to a significant improvement both of the convergence rates and the accuracy—not only for the displacement but also for the stress computations. Besides, the advantageous property of the PHS RBF-FD method supplemented with polynomials is that the differentiation weights can be

generated in either structured or unstructured (randomized) point-sets.

In the light of the promising numerical results and beneficial properties, the presented h-adaptive point-refinement strategy will be extended to the task of solving 3D BVPs.

References

1. Barnett GA (2015) A robust RBF-FD formulation based on polyharmonic splines and polynomials. PhD thesis, University of Colorado at Boulder
2. Bayona V (2019) An insight into RBF-FD approximations augmented with polynomials. Comput Math Appl 77(9):2337–2353. https://doi.org/10.1016/j.camwa.2018.12.029
3. Bayona V, Flyer N, Fornberg B (2019) On the role of polynomials in RBF-FD approximations: III. Behavior near domain boundaries. J Comput Phys 380:378–399. https://doi.org/10.1016/j.jcp.2018.12.013
4. Bayona V, Flyer N, Fornberg B, Barnett GA (2017) On the role of polynomials in RBF-FD approximations: II. Numerical solution of elliptic PDEs. J Comput Phys 332:257–273. https://doi.org/10.1016/j.jcp.2016.12.008
5. Bayona V, Moscoso M, Carretero M, Kindelan M (2010) RBF-FD formulas and convergence properties. J Comput Phys 229(22):8281–8295. https://doi.org/10.1016/j.jcp.2010.07.008
6. Chandhini G, Sanyasiraju YVSS (2007) Local RBF-FD solutions for steady convection–diffusion problems. Int J Numer Methods Eng 72(3):352–378. https://doi.org/10.1002/nme.2024
7. Cheng AD (2012) Multiquadric and its shape parameter—a numerical investigation of error estimate, condition number, and round-off error by arbitrary precision computation. Eng Anal Bound Elem 36(2):220–239. https://doi.org/10.1016/j.enganabound.2011.07.008
8. Chiappa A, Salvini P, Brutti C, Biancolini ME (2019) Upscaling 2D finite element analysis stress results using radial basis functions. Comput Struct 220:131–143. https://doi.org/10.1016/j.compstruc.2019.05.002
9. Dauge M, Düster A, Rank E (2015) Theoretical and numerical investigation of the finite cell method. J Sci Comput 65:1039–1064. https://doi.org/10.1007/s10915-015-9997-3
10. Davydov O, Oanh DT (2011) On the optimal shape parameter for Gaussian radial basis function finite difference approximation

of the Poisson equation. Comput Math Appl 62(5):2143–2161. https://doi.org/10.1016/j.camwa.2011.06.037

11. Demkowicz L (2007) Computing with hp-adaptive finite elements. One- and two-dimensional elliptic and maxwell problems. In: Applied mathematics and nonlinear science, vol I. Chapman & Hall/CRC Press, Taylor & Francis Group

12. Fasshauer GE, Zhang JG (2007) On choosing optimal shape parameters for RBF approximation. Numer Algorithms 45:1. https://doi.org/10.1007/s11075-007-9072-8

13. Flyer N, Barnett GA, Wicker LJ (2016) Enhancing finite differences with radial basis functions: experiments on the Navier–Stokes equations. J Comput Phys 316:39–62. https://doi.org/10.1016/j.jcp.2016.02.078

14. Flyer N, Fornberg B, Bayona V, Barnett GA (2016) On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy. J Comput Phys 321:21–38. https://doi.org/10.1016/j.jcp.2016.05.026

15. Fornberg B, Flyer N (2015) Fast generation of 2-D node distributions for mesh-free PDE discretizations. Comput Math Appl 69(7):531–544. https://doi.org/10.1016/j.camwa.2015.01.009

16. Fornberg B, Flyer N (2015) A primer on radial basis functions with applications to the geosciences. Society for Industrial and Applied Mathematics (SIAM), Philadelphia. https://doi.org/10.1137/1.9781611974041

17. Fornberg B, Flyer N (2015) Solving PDEs with radial basis functions. Acta Numer 24:215–258. https://doi.org/10.1017/S0962492914000130

18. Fornberg B, Larsson E, Flyer N (2011) Stable computations with Gaussian radial basis functions. SIAM J Sci Comput 33(2):869–892. https://doi.org/10.1137/09076756X

19. Fornberg B, Lehto E (2011) Stabilization of RBF-generated finite difference methods for convective PDEs. J Comput Phys 230(6):2270–2285. https://doi.org/10.1016/j.jcp.2010.12.014

20. Fornberg B, Lehto E, Powell C (2013) Stable calculation of Gaussian-based RBF-FD stencils. Comput Math Appl 65(4):627–637. https://doi.org/10.1016/j.camwa.2012.11.006

21. Hamfeldt BF, Salvador T (2018) Higher-order adaptive finite difference methods for fully nonlinear elliptic equations. J Sci Comput 75:1282–1306. https://doi.org/10.1007/s10915-017-0586-5

22. Huang CS, Lee CF, Cheng AD (2007) Error estimate, optimal shape factor, and high precision computation of multiquadric collocation method. Eng Anal Bound Elem 31(7):614–623. https://doi.org/10.1016/j.enganabound.2006.11.011

23. Kaufmann T, Engström C, Fumeaux C (2010) Residual-based adaptive refinement for meshless eigenvalue solvers. In: 2010 International conference on electromagnetics in advanced applications, pp 244–247. https://doi.org/10.1109/ICEAA.2010.5653604

24. Kee BB, Liu G, Zhang G, Lu C (2008) A residual based error estimator using radial basis functions. Finite Elem Anal Des 44(9):631–645. https://doi.org/10.1016/j.finel.2008.02.002

25. Kindelan M, Álvarez D, Gonzalez-Rodriguez P (2018) Frequency optimized RBF-FD for wave equations. J Comput Phys 371:564–580. https://doi.org/10.1016/j.jcp.2018.06.006

26. Larsson E, Lehto E, Heryudono A, Fornberg B (2013) Stable computation of differentiation matrices and scattered node stencils based on Gaussian radial basis functions. SIAM J Sci Comput 35(4):A2096–A2119. https://doi.org/10.1137/120899108

27. Martin B, Fornberg B (2017) Using radial basis function-generated finite differences (RBF-FD) to solve heat transfer equilibrium problems in domains with interfaces. Eng Anal Bound Elem 79:38–48. https://doi.org/10.1016/j.enganabound.2017.03.005

28. Mishra PK (2019) NodeLab: a MATLAB package for meshfree node-generation and adaptive refinement. J Open Source Softw 4(40):1173. https://doi.org/10.21105/joss.01173

29. Mishra PK, Fasshauer GE, Sen MK, Ling L (2019) A stabilized radial basis-finite difference (RBF-FD) method with hybrid kernels. Comput Math Appl 77(9):2354–2368. https://doi.org/10.1016/j.camwa.2018.12.027

30. Mishra PK, Ling L, Liu X, Sen MK (2020) Adaptive radial basis function generated finite-difference (RBF-FD) on non-uniform nodes using p-refinement. arXiv:2004.06319

31. Oanh DT, Davydov O, Phu HX (2017) Adaptive RBF-FD method for elliptic problems with point singularities in 2D. Appl Math Comput 313:474–497. https://doi.org/10.1016/j.amc.2017.06.006

32. Pardo D, Demkowicz L (2006) Integration of hp-adaptivity and a two-grid solver for elliptic problems. Comput Methods Appl Mech Eng 195:674–710. https://doi.org/10.1016/j.cma.2005.02.018

33. Persson PO, Strang G (2004) A simple mesh generator in Matlab. SIAM Rev 46(2):329–345

34. Petras A, Ling L, Ruuth S (2018) An RBF-FD closest point method for solving PDEs on surfaces. J Comput Phys 370:43–57. https://doi.org/10.1016/j.jcp.2018.05.022

35. Ramm E, Rank E, Rannacher R, Schweizerhof K, Stein E, Wendland W, Wittum G, Wriggers P, Wunderlich W (2003) Error-controlled adaptive finite elements in solid mechanics. Wiley, Chichester

36. Santos LGC, Manzanares-Filho N, Menon GJ, Abreu E (2018) Comparing RBF-FD approximations based on stabilized Gaussians and on polyharmonic splines with polynomials. Int J Numer Methods Eng 115(4):462–500. https://doi.org/10.1002/nme.5813

37. Shankar V (2017) The overlapped radial basis function-finite difference (RBF-FD) method: a generalization of RBF-FD. J Comput Phys 342:211–228. https://doi.org/10.1016/j.jcp.2017.04.037

38. Shankar V, Wright G, Kirby R, Fogelson A (2015) A radial basis function (RBF)-finite difference (FD) method for diffusion and reaction–diffusion equations on surfaces. J Sci Comput 63(3):745–768. https://doi.org/10.1007/s10915-014-9914-1

39. Szabó B, Babuška I (2011) Introduction to finite element analysis: formulation, verification and validation. Wiley, New York

40. Timoshenko S, Goodier JN (1970) Theory of elasticity, 3rd edn. McGraw-Hill, New York

41. Tolstykh AI, Shirobokov DA (2003) On using radial basis functions in a finite difference mode with applications to elasticity problems. Comput Mech 33:68–79. https://doi.org/10.1007/s00466-003-0501-9

42. Wright GB (2003) Radial basis function interpolation: numerical and analytical developments. PhD thesis, University of Colorado at Boulder

43. Wright GB, Fornberg B (2006) Scattered node compact finite difference-type formulas generated from radial basis functions. J Comput Phys 212(1):99–123. https://doi.org/10.1016/j.jcp.2005.05.030