Application of discontinuous Galerkin method to mechanical 2D problem with arbitrary polygonal and very high-order finite elements

Jan Jaśkowiec

Cracow University of Technology, Faculty of Civil Engineering,
Institute for Computational Civil Engineering
j.jaskowiec@L5.pk.edu.pl

Abstract

Discontinuous Galerkin with finite difference rules (DGFD) is applied to mechanical plane stress state problem. The considered domain is discretized by polygonal mesh. The polygonal elements can be for example hexagon, pentagon or just quadrangle or triangle. They do not have to be convex and fish-mesh in which the elements have fish shapes is used. When the elements are rectangular then the orthogonality of Chebyshev basis functions can be utilized. In such a case very high-order approximate solution can be obtained. In this work the approximation order exceeds 10 and reaches 60, which means in that case, 3660 numbers of degrees of freedom in a single element. The paper is illustrated by a benchmark example in which the exact solution is obtained by DGFD method for various meshes. In another example the stress concentration is easily recovered by very high-order version of DGFD method.

Key words: Discontinuous Galerkin, polygonal finite elements, plane stress state, very high-order method

1 Introduction

The discontinuous Galerkin (DG) method with finite difference rules (DGFD) method has been presented in [16] and further exploited in [19, 15, 18]. In this paper the DGFD method is applied to two-dimensional (2D) mechanical problem of a structure in the plane stress state (PSS). In the finite element mesh the arbitrary concave as well as non-concave polygonal finite elements are used. Furthermore, the mesh can be non-conforming (e.g. a vertex of one element can be located in the middle of edge of neighbouring element) [17, 19]. In the DGFD method a very high approximation order can be achieved, where the approximation order p can reach even 60 or higher [18]. The very high-order DGFD method is applied in this work in order to recover high levels of stress concentrations.

The discontinuous Galerkin (DG) method was proposed in the seventies of the previous century in [32] and constant development of the method has been observed since the time, see [9, 10, 13, 34]. This paper analyses the DG method for polygonal meshes. The application of DG method on the polygonal meshes is currently quite a new topic and there are a few papers tackling this problem, e.g. [37, 28, 29, 39, 7, 12, 1]. The author of the paper considered the polygonal meshes in DG method in [19] where the finite element cells can be non-simply-connected (i.e. a cell with holes inside) or even non-connected (i.e. a cell may consist of two or more parts which are completely separated from each other). However, in this work only the compact polygonal finite elements are considered for 2D vector elliptic problem.

In scientific literature the discontinuous Galerkin methods or the finite element method are referred to as high-order when polynomial order p is greater than 2, but on the other hand the approximation order p does not exceeds 10 e.g. [22, 3, 21, 33, 16, 17, 25] and [20, 11, 8, 6, 31, 36, 26, 38]. This paper uses the version of DGFD developed in [18] where the polynomial order in finite elements exceeds $p = 20$ and reaches $p = 101$ in two-dimensional space (2D) and even $p = 1001$ in one-dimensional space (1D). This work transfers the idea of a very high-order DGFD method to 2D mechanical problem in PSS.

In DG method, contrary to finite element method (FEM), the approximation functions are not continuous when passing from one finite element to another. Thus, the integration along the mesh skeleton (inter-element borders) is needed. In DG method the finite element mesh has to be divided into a set of cells, mesh skeleton and outer boundary, as shown in Fig. 1. In DGFD method, non-Lagrangian approximation functions are used. It means that there are no nodes nor shape functions in the method. Instead, basis functions are linearly combined and the degrees of freedom are just the coefficients of the linear combination. Abandoning the regime of approximation continuation it provides new opportunities because the neighbouring finite elements can be of different order or the finite element cells do not have to conform.

Figure 1: Polygonal mesh consisting of a set of cells, mesh skeleton and outer boundary

When solving the mechanical PSS problem by DGFD method the displacement gradients on the mesh skeleton have to be calculated. The displacement gradients are expressed in the skeleton local coordinates and then the derivatives in the skeleton normal direction are expressed by the fourth order finite difference rule [16]. On the other hand, the tangent derivatives are expressed by mid-values of such derivatives on both sides of the skeleton. Quite a similar approach is applied to the Dirichlet outer boundary where the conditions are imposed also by the fourth order finite difference rule. In consequence, well-conditioned linear set of equations is obtained, that takes into account both the compatibility conditions on the mesh skeleton and Dirichlet conditions on the outer boundary. In this paper the DGFD method is applied to polygonal finite elements cells as well as to very high-order element cells. The 2D very high-order cells have to be rectangular, or at least quadrilateral so that the transformation to square reference element is possible. It is due to the fact that the orthogonality of the Chebyshev (or Legendre) basis functions is fully exploited just for a square cell.

The following Section defines the mathematical problem of a 2D structure in plane stress state, with the inclusion of the discontinuity on the mesh skeleton. Section 3 focuses on the boundary conditions, where the special attention is put on the Dirichlet conditions. The details concerning the approximation are presented in Section 4. In the mechanical analysis the most important results are connected with stresses in the analysed structure. In FEM the raw stress fields are discontinuous, so in the DGFD method. Thus, the smoothing procedure has to be employed for continuous stress fields. Section 5 presents such a smoothing procedure which is based on Zienkiewicz-Zhu method but modified in order to take into account the discontinuity of the approximation field. The method is illustrated by examples presented in Section 6. In Section 6.1 the benchmark example with known exact solution is solved by a set of polygonal meshes including fish-mesh. Very high-order DGFD method is illustrated in Section 6.2 where stress concentrations are recovered by the method. Some final conclusions are presented in the last Section.

2 Mathematical model

The analysis of the mechanical problem starts with the standard equilibrium equation (momentum balance) on the domain V and outer boundary S with appropriate boundary conditions:

$$\begin{aligned} \operatorname{div} \boldsymbol{\sigma} + \mathbf{b} &= \mathbf{0} & \text{in } V \\ \boldsymbol{\sigma} \mathbf{n} &= \hat{\mathbf{t}} & \text{on } S^\sigma, \quad \mathbf{u} = \hat{\mathbf{u}} & \text{on } S^u \end{aligned} \quad (1)$$

where $\boldsymbol{\sigma}$ the stress tensor, $\mathbf{b}$ is the body force vector, $\hat{\mathbf{t}}$ is the prescribed traction forces vector, $\mathbf{u}$ is the relative displacement vector and $\hat{\mathbf{u}}$ is the prescribed displacement vector, S^σ and S^u are the part of the outer boundary on which the both types of the boundary conditions are defined.

The equation (1) is supplemented by the Hooke's laws

$$\boldsymbol{\sigma} = \mathbf{E} : \boldsymbol{\varepsilon} \quad (2)$$

where $\boldsymbol{\varepsilon}$ is the strain tensor, $\mathbf{E}$ is the fourth order Hooke's tensor which for two-dimensional case is defined as

$$\begin{aligned} (\mathbf{E})_{1111} &= (\mathbf{E})_{2222} = \frac{E}{1 - \nu^2} \\ (\mathbf{E})_{1122} &= (\mathbf{E})_{2211} = \nu \\ (\mathbf{E})_{1212} &= (\mathbf{E})_{1221} = (\mathbf{E})_{2112} = (\mathbf{E})_{2121} = \frac{E(1 - \nu)}{2(1 - \nu^2)} \end{aligned} \quad (3)$$

where E is the Young's modulus and ν is the Poisson's ratio.

In this work the small strains are assumed, that is why the Cauchy strain tensor is used

$$\boldsymbol{\varepsilon} = \nabla_s \mathbf{u} = \frac{1}{2} \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) \quad \text{in } V \quad (4)$$

Figure 2: Mesh skeleton with a set of skeleton local coordinates: normal and tangent

The problem formulation has to deal with approximation discontinuity along S^d . For future calculations the local coordinates associated with the mesh skeleton S^d are specified. They are based on two unit vectors: $(\mathbf{n}^d, \mathbf{s}^d)$, where $\mathbf{n}^d$ is normal to S^d and $\mathbf{s}^d$ is unit vectors tangent to S^d . In Fig. 2 the skeleton local coordinates are illustrated.

The value of the discontinuity jump is described by the discontinuity operators, which are defined with the help of a normal vector $\mathbf{n}^d$ to the mesh skeleton. The discontinuity operators for an arbitrary function g are defined as follows

$$\begin{aligned} \llbracket g \rrbracket_\epsilon(\mathbf{x}) &= g(\mathbf{x} + \epsilon \mathbf{n}^d) - g(\mathbf{x} - \epsilon \mathbf{n}^d) \\ \llbracket g \rrbracket(\mathbf{x}) &= \lim_{\epsilon \rightarrow 0} \llbracket g \rrbracket_\epsilon(\mathbf{x}) \end{aligned} \quad \text{on } S^d \quad (5)$$

The mean values of the discontinuity for the same function g are defined as

$$\begin{aligned} \langle g \rangle_\epsilon(\mathbf{x}) &= \frac{1}{2} (g(\mathbf{x} + \epsilon \mathbf{n}^d) + g(\mathbf{x} - \epsilon \mathbf{n}^d)) \\ \langle g \rangle(\mathbf{x}) &= \lim_{\epsilon \rightarrow 0} \langle g \rangle_\epsilon(\mathbf{x}) \end{aligned} \quad \text{on } S^d \quad (6)$$

The weak form of the mechanical problem from eq. (1) is constructed with the vector test function $\mathbf{v}$

$$\int_V \mathbf{v} \cdot \text{div } \boldsymbol{\sigma} \, dV + \int_V \mathbf{v} \cdot \mathbf{b} \, dV = 0 \quad \forall \mathbf{v} \quad (7)$$

The integration by parts of the first term in equation (7) results in the following equation

$$\int_V \text{div}(\boldsymbol{\sigma} \mathbf{v}) \, dV - \int_V \boldsymbol{\nabla} \mathbf{v} : \boldsymbol{\sigma} \, dV + \int_V \mathbf{v} \cdot \mathbf{b} \, dV = 0 \quad (8)$$

In this work it is more convenient to use the gradient of the displacement vector instead of the total strain tensor. It can be easily noticed that using the definition of the Hooke's tensor from eq. (3) the following equality is true

$$\mathbf{E} : \boldsymbol{\varepsilon} = \mathbf{E} : \boldsymbol{\nabla} \mathbf{u} \quad (9)$$

Then eq. (8) can be rewritten to the form

$$\int_V \text{div}(\boldsymbol{\sigma} \mathbf{v}) \, dV - \int_V \boldsymbol{\nabla} \mathbf{v} : \mathbf{E} : \boldsymbol{\nabla} \mathbf{u} \, dV + \int_V \mathbf{v} \cdot \mathbf{b} \, dV = 0 \quad (10)$$

Due to the discontinuities on S^d the first integral in eq. (10) is equivalent to the integral along the outer boundary and the mesh skeleton, i.e.

$$\int_V \text{div}(\boldsymbol{\sigma} \mathbf{v}) \, dV = \int_S \mathbf{v} \cdot \boldsymbol{\sigma} \mathbf{n} \, dS - \int_{S^d} \llbracket \boldsymbol{\sigma} \mathbf{v} \rrbracket \cdot \mathbf{n}^d \, dS \quad (11)$$

It is a well known property in which the jump of the product of two functions can be expressed as

$$[[\boldsymbol{\sigma}\mathbf{v}]] = \langle \boldsymbol{\sigma} \rangle [[\mathbf{v}]] + [[\boldsymbol{\sigma}]] \langle \mathbf{v} \rangle \quad (12)$$

We want to softly enforce the continuity of the stresses in the $\mathbf{n}^d$ direction, that is why

$$[[\boldsymbol{\sigma}]]\mathbf{n}^d = \mathbf{0} \quad \Rightarrow \quad \langle \boldsymbol{\sigma} \rangle \mathbf{n}^d = \boldsymbol{\sigma}\mathbf{n}^d \quad \text{on } S^d \quad (13)$$

Eq. (10) changes now in the following way

$$\int_S \mathbf{v} \cdot \boldsymbol{\sigma}\mathbf{n} dS - \int_{S^d} [[\mathbf{v}]] \cdot \boldsymbol{\sigma}\mathbf{n}^d dS - \int_V \boldsymbol{\nabla}\mathbf{v} : \boldsymbol{\sigma} dV + \int_V \mathbf{v} \cdot \mathbf{b} dV = 0 \quad (14)$$

In order to calculate the second integral in eq. (14) the scalar value in the integral ($[[\mathbf{v}]] \cdot \boldsymbol{\sigma}\mathbf{n}^d$) has to be evaluated. The scalar value can be calculated in the global coordinates as well as in the S^d local coordinates. And in this case, it is much easier to perform the calculations in the local coordinates associated with the two unit vectors ($\mathbf{n}^d, \mathbf{s}^d$). It can be noticed that every vector $\mathbf{V}$ can be expressed by the global or local coefficients

$$\mathbf{V} = \begin{bmatrix} V_x \\ V_y \end{bmatrix} = \mathbf{n}^d V_n + \mathbf{s}^d V_s = \mathbf{Q}^d \begin{bmatrix} V_n \\ V_s \end{bmatrix} = \mathbf{Q}^d \mathbf{V}^d \quad \Rightarrow \quad \mathbf{V}^d = \mathbf{Q}^{dT} \mathbf{V} \quad (15)$$

where $\mathbf{Q}^d$ is the transformation matrix between the skeleton local coordinates and the global coordinates.

Now the inner product from the second integral in eq. (14) can be expressed with the help of the local coordinates and the transformation matrix as follows

$$[[\mathbf{v}]] \cdot \boldsymbol{\sigma}\mathbf{n}^d = [[\mathbf{v}]] \cdot \mathbf{Q}^d \mathbf{Q}^{dT} \boldsymbol{\sigma} \mathbf{Q}^d \mathbf{i} = [[\mathbf{v}]] \cdot \mathbf{Q}^d \boldsymbol{\sigma}^d \mathbf{i} = [[\mathbf{v}]] \cdot \mathbf{Q}^d \mathbf{t}^d \quad (16)$$

where $\mathbf{i}$ is the basis unit vector, $\boldsymbol{\sigma}^d$ is the stress tensor described in the S^d local coordinates and $\mathbf{t}^d$ is the tractions vector described in the S^d local coordinates, i.e.:

$$\mathbf{i} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \boldsymbol{\sigma}^d = \begin{bmatrix} \sigma_{nn}^d & \sigma_{ns}^d \\ \sigma_{sn}^d & \sigma_{ss}^d \end{bmatrix}, \quad \mathbf{t}^d = \begin{bmatrix} t_n^d \\ t_s^d \end{bmatrix} = \begin{bmatrix} t_n^d \\ t_s^d \end{bmatrix} \quad (17)$$

In this paper the material of the regarded domain is isotropic therefore, the Hooke's law from eqs. (2)₂ and (3) can directly be used also in the skeleton local coordinates, that is

$$\begin{aligned} t_n^d &= \frac{E}{1-\nu^2} \varepsilon_{nn} + \frac{E\nu}{1-\nu^2} \varepsilon_{ss} - \frac{E(1+\nu)}{1-\nu^2} \alpha \Theta = \frac{E}{1-\nu^2} \frac{\partial u_n}{\partial n} + \frac{E\nu}{1-\nu^2} \frac{\partial u_s}{\partial s} \\ t_s^d &= \frac{E(1-\nu)}{1-\nu^2} \varepsilon_{sn} = \frac{E(1-\nu)}{1-\nu^2} \frac{\partial u_s}{\partial n} + \frac{E(1-\nu)}{1-\nu^2} \frac{\partial u_n}{\partial s} \end{aligned} \quad (18)$$

where $\frac{\partial}{\partial n}, \frac{\partial}{\partial s}$ are the partial derivatives into $\mathbf{n}^d$ and $\mathbf{s}^d$ directions, respectively. The displacement coefficients in the local coordinates are related to the displacement vector in the global coordinates in the following way

$$u_n = \mathbf{n}^d \cdot \mathbf{u}, \quad u_s = \mathbf{s}^d \cdot \mathbf{u} \quad (19)$$

In consequence, the traction coefficients from eq. (18) come to the forms

$$\begin{aligned} t_n^d &= \frac{E}{1-\nu^2} \mathbf{n}^d \cdot \frac{\partial \mathbf{u}}{\partial n} + \frac{E\nu}{1-\nu^2} \mathbf{s}^d \cdot \frac{\partial \mathbf{u}}{\partial s} \\ t_s^d &= \frac{E(1-\nu)}{1-\nu^2} \mathbf{s}^d \cdot \frac{\partial \mathbf{u}}{\partial n} + \frac{E(1-\nu)}{1-\nu^2} \mathbf{n}^d \cdot \frac{\partial \mathbf{u}}{\partial s} \end{aligned} \quad (20)$$

In eq. (20) the derivatives into the $\mathbf{n}^d$ are evaluated, by the fourth order finite difference rule [16]

$$\frac{\partial \mathbf{u}}{\partial n} = \frac{3}{2} \frac{[[\mathbf{u}]]_z}{2z} - \frac{1}{2} \left\langle \frac{\partial \mathbf{u}}{\partial n} \right\rangle_z \quad \text{on } S^d \quad (21)$$

The derivative into the tangent direction is evaluated by mid-value operators

$$\frac{\partial \mathbf{u}}{\partial s} = \left\langle \frac{\partial \mathbf{u}}{\partial s} \right\rangle \quad \text{on } S^d \quad (22)$$

The derivatives into the local coordinates directions in the mid-value operators are subsequently written with the help of gradients in the global coordinates

$$\begin{aligned}\left\langle \frac{\partial \mathbf{u}}{\partial n} \right\rangle_z &= \langle \nabla \mathbf{u} \cdot \mathbf{n}^d \rangle_z = \langle \nabla \mathbf{u} \rangle_z \cdot \mathbf{n}^d \\ \left\langle \frac{\partial \mathbf{u}}{\partial s} \right\rangle &= \langle \nabla \mathbf{u} \cdot \mathbf{s}^d \rangle = \langle \nabla \mathbf{u} \rangle \cdot \mathbf{s}^d\end{aligned}\quad (23)$$

The tractions in the local coordinates can now be rewritten

$$\begin{aligned}t_n^d &= \frac{E}{1-\nu^2} \mathbf{n}^d \cdot \left(\frac{3}{2} \frac{[\mathbf{u}]_z}{2z} - \frac{1}{2} \langle \nabla \mathbf{u} \rangle_z \cdot \mathbf{n}^d \right) + \frac{E\nu}{1-\nu^2} \mathbf{s}^d \cdot \langle \nabla \mathbf{u} \rangle \cdot \mathbf{s}^d \\ t_s^d &= \frac{E(1-\nu)}{1-\nu^2} \mathbf{s}^d \cdot \left(\frac{3}{2} \frac{[\mathbf{u}]_z}{2z} - \frac{1}{2} \langle \nabla \mathbf{u} \rangle_z \cdot \mathbf{n}^d \right) + \frac{E(1-\nu)}{1-\nu^2} \mathbf{n}^d \cdot \langle \nabla \mathbf{u} \rangle \cdot \mathbf{s}^d\end{aligned}\quad (24)$$

and subsequently

$$\begin{aligned}t_n^d &= \frac{E}{1-\nu^2} \mathbf{n}^d \cdot \frac{3}{2} \frac{[\mathbf{u}]_z}{2z} - \frac{E}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{n}^d) : \frac{1}{2} \langle \nabla \mathbf{u} \rangle_z + \frac{E\nu}{1-\nu^2} (\mathbf{s}^d \otimes \mathbf{s}^d) : \langle \nabla \mathbf{u} \rangle \\ t_s^d &= \frac{E(1-\nu)}{1-\nu^2} \mathbf{s}^d \cdot \frac{3}{2} \frac{[\mathbf{u}]_z}{2z} - \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s}^d \otimes \mathbf{n}^d) : \frac{1}{2} \langle \nabla \mathbf{u} \rangle_z + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{s}^d) : \langle \nabla \mathbf{u} \rangle\end{aligned}\quad (25)$$

Bearing in mind that in 2D

$$\mathbf{s}^d \otimes \mathbf{s}^d = \mathbf{I} - \mathbf{n}^d \otimes \mathbf{n}^d \quad (26)$$

the tractions from eq. (25) change into

$$\begin{aligned}t_n^d &= \frac{E}{1-\nu^2} \mathbf{n}^d \cdot \frac{3}{2} \frac{[\mathbf{u}]_z}{2z} - \frac{E}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{n}^d) : \frac{1}{2} \langle \nabla \mathbf{u} \rangle_z + \frac{E\nu}{1-\nu^2} (\mathbf{I} - \mathbf{n}^d \otimes \mathbf{n}^d) : \langle \nabla \mathbf{u} \rangle \\ t_s^d &= \frac{E(1-\nu)}{1-\nu^2} \mathbf{s}^d \cdot \frac{3}{2} \frac{[\mathbf{u}]_z}{2z} - \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s}^d \otimes \mathbf{n}^d) : \frac{1}{2} \langle \nabla \mathbf{u} \rangle_z + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{s}^d) : \langle \nabla \mathbf{u} \rangle\end{aligned}\quad (27)$$

The traction vector in the global coordinates $\mathbf{t}$ on S^d is provided by transformation of $\mathbf{t}^d$

$$\mathbf{t} = \mathbf{Q}^d \mathbf{t}^d = \mathbf{n}^d t_n^d + \mathbf{s}^d t_s^d \quad (28)$$

The relations from eq. (27) are incorporated into eq. (28)

$$\begin{aligned}\mathbf{t} &= \frac{E}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{n}^d) \frac{3}{2} \frac{[\mathbf{u}]_z}{2z} - \frac{E}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{n}^d \otimes \mathbf{n}^d) : \frac{1}{2} \langle \nabla \mathbf{u} \rangle_z + \frac{E\nu}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{I} - \mathbf{n}^d \otimes \mathbf{n}^d \otimes \mathbf{n}^d) : \langle \nabla \mathbf{u} \rangle \\ &+ \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s}^d \otimes \mathbf{s}^d) \frac{3}{2} \frac{[\mathbf{u}]_z}{2z} - \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s}^d \otimes \mathbf{s}^d \otimes \mathbf{n}^d) : \frac{1}{2} \langle \nabla \mathbf{u} \rangle_z + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s}^d \otimes \mathbf{n}^d \otimes \mathbf{s}^d) : \langle \nabla \mathbf{u} \rangle\end{aligned}\quad (29)$$

Eq. (29) can be rewritten in order to simplify it

$$\begin{aligned}\mathbf{t} &= \frac{3}{2} \left(\frac{E}{1-\nu^2} \mathbf{I} - \frac{E\nu}{1-\nu^2} \mathbf{s}^d \otimes \mathbf{s}^d \right) \frac{[\mathbf{u}]_z}{2z} \\ &- \frac{1}{2} \left(\frac{E}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{n}^d \otimes \mathbf{n}^d) + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s}^d \otimes \mathbf{s}^d \otimes \mathbf{n}^d) \right) : \langle \nabla \mathbf{u} \rangle_z \\ &+ \left(\frac{E\nu}{1-\nu^2} (\mathbf{n}^d \otimes \mathbf{I} - \mathbf{n}^d \otimes \mathbf{n}^d \otimes \mathbf{n}^d) + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s}^d \otimes \mathbf{n}^d \otimes \mathbf{s}^d) \right) : \langle \nabla \mathbf{u} \rangle\end{aligned}\quad (30)$$

And finally, the vector of tractions can be expressed in the following way

$$\mathbf{t} = \mathbf{E}_1^d [\mathbf{u}]_z - \mathbf{E}_2^d : \langle \nabla \mathbf{u} \rangle_z + \mathbf{E}_3^d : \langle \nabla \mathbf{u} \rangle \quad (31)$$

where $\mathbf{E}_1^d$, $\mathbf{E}_2^d$ and $\mathbf{E}_3^d$ are the appropriate tensors of the second and third order.

Now eq. (14) is transformed as follows

$$\begin{aligned}&\int_S \mathbf{v} \cdot \boldsymbol{\sigma} \mathbf{n} dS - \int_{S^d} [\mathbf{v}] \cdot \mathbf{E}_1^d [\mathbf{u}]_z - [\mathbf{v}] \cdot \mathbf{E}_2^d : \langle \nabla \mathbf{u} \rangle_z + [\mathbf{v}] \cdot \mathbf{E}_3^d : \langle \nabla \mathbf{u} \rangle dS \\ &- \int_V \nabla \mathbf{v} : \boldsymbol{\sigma} dV + \int_V \mathbf{v} \cdot \mathbf{b} dV = 0\end{aligned}\quad (32)$$

3 Boundary conditions

Domain boundary is decomposed in accordance with essential (Dirichlet) and natural (Neumann) boundary conditions. In most cases, Neumann boundary conditions are provided directly to the weak formulations. On the other hand, Dirichlet boundary conditions are applied by setting the boundary values to the degrees of freedom in the final aggregated form of the algebraic equations system. In the version of DG method that is applied in this paper, there are no nodes, so the Dirichlet conditions are applied by using the technique presented in [16], i.e with the help of the fourth order finite difference rules. As a result, the boundary conditions are consistently incorporated into the weak form of the considered problem.

The boundary conditions are presented by the first integral in eq. (14). This integral may be written by the sum of integrals along S^σ and S^u parts of the outer boundary

$$I_1 = \int_S \mathbf{v} \cdot \boldsymbol{\sigma} \mathbf{n} dS = \int_S \mathbf{v} \cdot \mathbf{t} dS = \int_{S^\sigma} \mathbf{v} \cdot \hat{\mathbf{t}} dS + \int_{S^u} \mathbf{v} \cdot \mathbf{t} dS \quad (33)$$

The traction vector $\mathbf{t}$ on S^u is described in the global coordinates. However, it can be also written in the local coordinates, defined on the outer boundary by the set of two perpendicular unit vectors $(\mathbf{n}, \mathbf{s})$, where $\mathbf{s}$ is tangent to the outer boundary. The relation between the traction vector in the global and local coordinates is based on the transformation matrix $\mathbf{Q}$

$$\mathbf{t} = \mathbf{Q} \mathbf{t}^b \quad \text{where} \quad \mathbf{Q} = [\mathbf{n} \quad \mathbf{s}] \quad \text{on } S^u \quad (34)$$

It can be easily shown that the vector $\mathbf{t}^b$ has the same form as shown in eq. (20)

$$\begin{aligned} t_n^b &= \frac{E}{1-\nu^2} \mathbf{n} \cdot \frac{\partial \mathbf{u}}{\partial n} + \frac{E\nu}{1-\nu^2} \mathbf{s} \cdot \frac{\partial \mathbf{u}}{\partial s} \\ t_s^b &= \frac{E(1-\nu)}{1-\nu^2} \mathbf{s} \cdot \frac{\partial \mathbf{u}}{\partial n} + \frac{E(1-\nu)}{1-\nu^2} \mathbf{n} \cdot \frac{\partial \mathbf{u}}{\partial s} \end{aligned} \quad \text{on } S^u \quad (35)$$

In eq. (35) the derivative into the tangent direction can be expressed by the gradient tensor in similar way as in eq. (23)

$$\frac{\partial \mathbf{u}}{\partial s} = \boldsymbol{\nabla} \mathbf{u} \cdot \mathbf{s} \quad \text{on } S^u \quad (36)$$

The derivative into the normal direction is approximated by the fourth order approximation adequately to [16]

$$\frac{\partial \mathbf{u}}{\partial n} = 6 \frac{\hat{\mathbf{u}} - \mathbf{u}(\mathbf{x}_{2z})}{2z} - 4 \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_z) \cdot \mathbf{n} - \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_{2z}) \cdot \mathbf{n} \quad (37)$$

The relations in eqs. (36) and (37) are subsequently substituted in the eq. (35)

$$\begin{aligned} t_n^b &= \frac{3}{z} \frac{E}{1-\nu^2} \mathbf{n} \cdot \hat{\mathbf{u}} - \frac{3}{z} \frac{E}{1-\nu^2} \mathbf{n} \cdot \mathbf{u} - 4 \frac{E}{1-\nu^2} \mathbf{n} \cdot \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_z) \cdot \mathbf{n} \\ &\quad - \frac{E}{1-\nu^2} \mathbf{n} \cdot \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_{2z}) \cdot \mathbf{n} + \frac{E\nu}{1-\nu^2} \mathbf{s} \cdot \boldsymbol{\nabla} \mathbf{u} \cdot \mathbf{s} \\ t_s^b &= \frac{3}{z} \frac{E(1-\nu)}{1-\nu^2} \mathbf{s} \cdot \hat{\mathbf{u}} - \frac{3}{z} \frac{E(1-\nu)}{1-\nu^2} \mathbf{s} \cdot \mathbf{u} - 4 \frac{E(1-\nu)}{1-\nu^2} \mathbf{s} \cdot \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_z) \cdot \mathbf{n} \\ &\quad - \frac{E(1-\nu)}{1-\nu^2} \mathbf{s} \cdot \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_{2z}) \cdot \mathbf{n} + \frac{E(1-\nu)}{1-\nu^2} \mathbf{n} \cdot \boldsymbol{\nabla} \mathbf{u} \cdot \mathbf{s} \end{aligned} \quad \text{on } S^u \quad (38)$$

The definition of tractions in eq. (38) can be reorganized in the following way

$$\begin{aligned} t_n^b &= \frac{3}{z} \frac{E}{1-\nu^2} \mathbf{n} \cdot \hat{\mathbf{u}} - \frac{3}{z} \frac{E}{1-\nu^2} \mathbf{n} \cdot \mathbf{u} - 4 \frac{E}{1-\nu^2} (\mathbf{n} \otimes \mathbf{n}) : \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_z) \\ &\quad - \frac{E}{1-\nu^2} (\mathbf{n} \otimes \mathbf{n}) : \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_{2z}) + \frac{E\nu}{1-\nu^2} (\mathbf{s} \otimes \mathbf{s}) : \boldsymbol{\nabla} \mathbf{u} + \frac{E\nu}{1-\nu^2} (\mathbf{r} \otimes \mathbf{r}) : \boldsymbol{\nabla} \mathbf{u} \\ t_s^b &= \frac{3}{z} \frac{E(1-\nu)}{1-\nu^2} \mathbf{s} \cdot \hat{\mathbf{u}} - \frac{3}{z} \frac{E(1-\nu)}{1-\nu^2} \mathbf{s} \cdot \mathbf{u} - 4 \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{n}) : \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_z) \\ &\quad - \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{n}) : \boldsymbol{\nabla} \mathbf{u}(\mathbf{x}_{2z}) + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{n} \otimes \mathbf{s}) : \boldsymbol{\nabla} \mathbf{u} \end{aligned} \quad \text{on } S^u \quad (39)$$

The traction vector in the global coordinates on the S^u is finally expressed as

$$\begin{aligned}
\mathbf{t} &= \mathbf{n} t_n^b + \mathbf{s} t_s^b \\
&= \frac{3}{z} \frac{E}{1-\nu^2} (\mathbf{n} \otimes \mathbf{n}) \hat{\mathbf{u}} - \frac{3}{z} \frac{E}{1-\nu^2} (\mathbf{n} \otimes \mathbf{n}) \mathbf{u} - 4 \frac{E}{1-\nu^2} (\mathbf{n} \otimes \mathbf{n} \otimes \mathbf{n}) : \nabla \mathbf{u}(\mathbf{x}_z) \\
&\quad - \frac{E}{1-\nu^2} (\mathbf{n} \otimes \mathbf{n} \otimes \mathbf{n}) : \nabla \mathbf{u}(\mathbf{x}_{2z}) + \frac{E\nu}{1-\nu^2} (\mathbf{n} \otimes \mathbf{s} \otimes \mathbf{s}) : \nabla \mathbf{u} + \frac{E\nu}{1-\nu^2} (\mathbf{n} \otimes \mathbf{r} \otimes \mathbf{r}) : \nabla \mathbf{u} \\
&\quad + \frac{3}{z} \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{s}) \hat{\mathbf{u}} - \frac{3}{z} \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{s}) \mathbf{u} - 4 \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{s} \otimes \mathbf{n}) : \nabla \mathbf{u}(\mathbf{x}_z) \\
&\quad - \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{s} \otimes \mathbf{n}) : \nabla \mathbf{u}(\mathbf{x}_{2z}) + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{n} \otimes \mathbf{s}) : \nabla \mathbf{u}
\end{aligned} \tag{40}$$

The traction vector $\mathbf{t}$ on S^u can be organized in the following form

$$\begin{aligned}
\mathbf{t} &= \frac{3}{z} \left(\frac{E}{1-\nu^2} \mathbf{I} - \frac{E\nu}{1-\nu^2} \mathbf{s} \otimes \mathbf{s} \right) \hat{\mathbf{u}} - \frac{3}{z} \left(\frac{E}{1-\nu^2} \mathbf{I} - \frac{E\nu}{1-\nu^2} \mathbf{s} \otimes \mathbf{s} \right) \mathbf{u} \\
&\quad + \left(\frac{E\nu}{1-\nu^2} (\mathbf{n} \otimes \mathbf{s} \otimes \mathbf{s}) + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{n} \otimes \mathbf{s}) \right) : \nabla \mathbf{u} \\
&\quad - 4 \left(\frac{E}{1-\nu^2} (\mathbf{n} \otimes \mathbf{n} \otimes \mathbf{n}) + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{s} \otimes \mathbf{n}) \right) : \nabla \mathbf{u}(\mathbf{x}_z) \\
&\quad - \left(\frac{E}{1-\nu^2} (\mathbf{n} \otimes \mathbf{n} \otimes \mathbf{n}) + \frac{E(1-\nu)}{1-\nu^2} (\mathbf{s} \otimes \mathbf{s} \otimes \mathbf{n}) \right) : \nabla \mathbf{u}(\mathbf{x}_{2z})
\end{aligned} \tag{41}$$

Finally, the traction vector on S^u can be written as follows

$$\mathbf{t} = \mathbf{E}_1^b \hat{\mathbf{u}} - \mathbf{E}_1^b \mathbf{u} + \mathbf{E}_2^b : \nabla \mathbf{u} - \mathbf{E}_3^b : \nabla \mathbf{u}(\mathbf{x}_z) - \mathbf{E}_4^b : \nabla \mathbf{u}(\mathbf{x}_{2z}) \tag{42}$$

where $\mathbf{E}_1^b$, $\mathbf{E}_2^b$, $\mathbf{E}_3^b$ and $\mathbf{E}_4^b$ are appropriate tensors of the second and third order.

The weak form equation (14) can be written as the equation of bilinear and linear forms

$$A^m(\mathbf{u}, \Theta, \mathbf{v}) = F^m(\mathbf{v}), \quad \forall \mathbf{v} \tag{43}$$

where the definitions of the operators are as follows

$$\begin{aligned}
A^m(\mathbf{u}, \Theta, \mathbf{v}) &= \int_{S^u} \mathbf{v} \cdot \mathbf{E}_1^b \mathbf{u} dS - \int_{S^u} \mathbf{v} \cdot \mathbf{E}_2^b : \nabla \mathbf{u} dS + \int_{S^u} \mathbf{v} \cdot \mathbf{E}_3^b : \nabla \mathbf{u}(\mathbf{x}_z) dS \\
&\quad + \int_{S^u} \mathbf{v} \cdot \mathbf{E}_4^b : \nabla \mathbf{u}(\mathbf{x}_{2z}) dS \\
&\quad + \int_{S^d} \llbracket \mathbf{v} \rrbracket \cdot \mathbf{E}_1^d \llbracket \mathbf{u} \rrbracket_z dS - \int_{S^d} \llbracket \mathbf{v} \rrbracket \cdot \mathbf{E}_2^d : \langle \nabla \mathbf{u} \rangle_z dS + \int_{S^d} \llbracket \mathbf{v} \rrbracket \cdot \mathbf{E}_3^d : \langle \nabla \mathbf{u} \rangle dS \\
&\quad + \int_V \nabla \mathbf{v} : \mathbf{E} : \nabla \mathbf{u} dV
\end{aligned} \tag{44}$$

$$F^m(\mathbf{v}) = \int_V \mathbf{v} \cdot \mathbf{b} dV + \int_{S^\sigma} \mathbf{v} \cdot \hat{\mathbf{t}} dS + \int_{S^u} \mathbf{v} \cdot \mathbf{E}_1^b \hat{\mathbf{u}} dS \tag{45}$$

4 Approximation

In DG method arbitrary basis functions for the approximation in finite elements can be applied, see [16, 24, 40]. However, the polynomial approximation is most universal and can be successfully applied to every kind of a problem. In particular, the Chebyshev polynomials can be chosen as basis functions for approximation in finite elements. As it was shown in [18], the Chebyshev basis or the monomials basis functions give the same results for the approximation order p up to ten. For monomial basis with $p > 10$ the truncation errors affect strongly the final results. The influence of

truncation errors can be successfully avoided when the Chebyshev polynomials are applied. The orthogonality property of these functions can be utilized, providing that rectangular finite elements are present. In a case of non-rectangular finite elements the transformation to the square reference finite element can be applied, for more details see [18]. In this work the Chebyshev basis functions are applied to both arbitrary polygonal finite elements as well as to rectangular ones with the application of a very high approximation order.

Approximation in the e -th single finite element is constructed with the help of the element local basis functions and the local degrees of freedom

$$u_\alpha^e(\mathbf{x}) = \sum_{i=1}^{n_e} b_i^e(\mathbf{x}) (\tilde{\mathbf{u}}_\alpha^e)_i = \mathbf{b}^e \tilde{\mathbf{u}}_\alpha^e, \quad \text{for } \mathbf{x} \in V^e \text{ and } \alpha = x, y \quad (46)$$

where u_α^e is the displacement in the e -th element into the x or y direction, $\tilde{\mathbf{u}}_\alpha^e$ is the element degrees of freedom (edof) vector for the e -th element for α direction, $\mathbf{b}^e$ is the vector of the e -th element basis functions. The 'element degrees of freedom' indicates degrees of freedom connected with the single finite element and 'element basis functions' means that the basis functions are defined in the local coordinates connected with the element and their support is limited to the element. The choice of the local basis functions for each cell is quite arbitrary so in this analysis the Chebyshev basis functions are chosen. The basis functions are defined in the cell local coordinates scaled to the sizes of the cells, i.e.:

$$x^e = \frac{x - x_m^e}{0.5 h_x^e}, \quad y^e = \frac{y - y_m^e}{0.5 h_y^e} \quad (47)$$

where the point (x_m^e, y_m^e) is the centre of gravity of the e -th element cell and h_x^e, h_y^e are the characteristic lengths of the e -th cell in the x and y directions, respectively.

The definition of the Chebyshev functions are well known, but for the sake of clarity the recursive definition of Chebyshev functions is quoted

$$T_0(\xi) = 1, \quad T_1(\xi) = \xi, \quad T_{n+1}(\xi) = 2 \cdot \xi \cdot T_n(\xi) - T_{n-1}(\xi) \quad (48)$$

The vectors of the basis functions for the Chebyshev basis functions in 1D are then as follows

$$\begin{aligned} \text{for } p = 1 : \quad \mathbf{b}^e &= \mathbf{b}_1^e = [1 \quad T_1(x^e)] \\ \text{for } p = 2 : \quad \mathbf{b}^e &= \mathbf{b}_2^e = [\mathbf{b}_1^e \quad T_2(x^e)] \\ \text{for } p = 3 : \quad \mathbf{b}^e &= \mathbf{b}_3^e = [\mathbf{b}_2^e \quad T_3(x^e)] \\ \text{for } p = 4 : \quad \mathbf{b}^e &= \mathbf{b}_4^e = [\mathbf{b}_3^e \quad T_4(x^e)] \\ &\text{and so on ...} \end{aligned} \quad (49)$$

and in 2D

$$\begin{aligned} \text{for } p = 1 : \quad \mathbf{b}^e &= \mathbf{b}_1^e = [1 \quad T_1(x^e) \quad T_1(y^e)] \\ \text{for } p = 2 : \quad \mathbf{b}^e &= \mathbf{b}_2^e = [\mathbf{b}_1^e \quad T_1(x^e)T_1(y^e) \quad T_2(x^e) \quad T_2(y^e)] \\ \text{for } p = 3 : \quad \mathbf{b}^e &= \mathbf{b}_3^e = [\mathbf{b}_2^e \quad T_3(x^e) \quad T_2(x^e)T_1(y^e) \quad T_1(x^e)T_2(y^e) \quad T_3(y^e)] \\ \text{for } p = 4 : \quad \mathbf{b}^e &= \mathbf{b}_4^e = [\mathbf{b}_3^e \quad T_4(x^e) \quad T_3(x^e)T_1(y^e) \quad T_2(x^e)T_2(y^e) \quad T_1(x^e)T_3(y^e) \quad T_4(y^e)] \\ &\text{and so on ...} \end{aligned} \quad (50)$$

In 1D a single finite element is just a segment and due to local transformation in eq. (47) each segment is transformed to $[-1, 1]$ reference segment. And the same is for rectangular finite element in 2D, which is transformed to the square $[-1, 1] \times [-1, 1]$ reference element. Therefore, for rectangular finite elements very high-order approximation can be used. However, in 2D case finite element cell can be of quite arbitrary shape (e.g. triangle quadrilateral, pentagon) [19]. In this paper, the results are presented, for instant, for a fish-mesh, i.e. the finite elements cells are of fish shapes. But for non-rectangular finite elements the approximation order is limited to about $p = 10$ due to truncation errors.

Eq. (46) presents the approximation for one element cell. It can be rewritten for the displacement vector approximation in the whole domain

$$\mathbf{u} = \Phi \tilde{\mathbf{u}}, \quad \text{for } \mathbf{x} \in V \quad (51)$$

where $\tilde{\mathbf{u}}$ is the vector of displacement degrees of freedom for the whole mesh and Φ is the displacement approximation matrix also for the whole mesh.

The jumps and mid-values are approximated using the same vector of degrees of freedom, i.e.

$$[\![\mathbf{u}]\!] = [\![\Phi]\!]\tilde{\mathbf{u}}, \quad \langle \mathbf{u} \rangle = \langle \Phi \rangle \tilde{\mathbf{u}} \quad \text{on } S^d \quad (52)$$

In the same way the approximation of the jump and mean values at distance ϵ from S^d are defined

$$[\![\mathbf{u}]\!]_\epsilon = [\![\Phi]\!]_\epsilon \tilde{\mathbf{u}}, \quad \langle \mathbf{u} \rangle_\epsilon = \langle \Phi \rangle_\epsilon \tilde{\mathbf{u}} \quad (53)$$

This paper considers the Galerkin formulation, therefore the same approximations as in the equations (51), (52) and (53) are applied to the test function $\mathbf{v}$.

When the approximations from eqs (51), to (53) are substituted into the weak form equation (43), the linear system of equations is obtained

$$\mathbf{K}\tilde{\mathbf{u}} = \mathbf{F} \quad (54)$$

$$\begin{aligned} \mathbf{K} = & \int_V \mathbf{B}^T : \mathbf{E} : \mathbf{B} dV + \int_{S^u} \Phi^T \mathbf{E}_1^b \Phi dS - \int_{S^u} \Phi^T \mathbf{E}_2^b : \mathbf{B} dS + \int_{S^u} \Phi^T \mathbf{E}_3^b : \mathbf{B}(\mathbf{x}_z) dS + \int_{S^u} \Phi^T \mathbf{E}_4^b : \mathbf{B}(\mathbf{x}_{2z}) dS \\ & + \int_{S^d} [\![\Phi]\!]^T \mathbf{E}_1^d [\![\Phi]\!]_z dS - \int_{S^d} [\![\Phi]\!]^T \mathbf{E}_2^d : \langle \mathbf{B} \rangle_z dS + \int_{S^d} [\![\Phi]\!]^T \mathbf{E}_3^d : \langle \mathbf{B} \rangle dS \end{aligned} \quad (55)$$

$$\mathbf{F} = \int_V \Phi^T \mathbf{b} dV + \int_{S^\sigma} \Phi^T \hat{\mathbf{t}} dS + \int_{S^u} \Phi^T \mathbf{E}_1^b \hat{\mathbf{u}} dS \quad (56)$$

where $\mathbf{B} = \nabla \Phi$.

5 Postprocessing

When the system of equations in eq. (54) is solved, the displacement field is found. In spite of the fact that, the approximation upon the mesh is not continuous, the displacement fields are continuous anyway (or the discontinuity levels are negligibly small). On the other hand, the stresses calculated by the eq. (2) are not continuous, like in the finite element method. In order to obtain the continuous stresses, the smoothing procedures have to be performed. For the purpose of this analysis, the procedure based on the Zienkiewicz-Zhu [41] patch recovery is chosen. In the smoothing procedure the fact that the approximation is discontinuous needs to be taken into account, so that the final stresses are continuous.

The smooth stress component σ_{ij}^* is to be found by minimising the error measure

$$\epsilon_{ij} = \frac{1}{|V|} \int_V (\sigma_{ij}^* - \sigma_{ij})^2 dV + \frac{1}{|S^d|} \int_{S^d} [\![\sigma_{ij}^*]\!]^2 dS \quad \text{for } i, j = x, y \quad (57)$$

where σ_{ij} is the stress component obtained by the Hook's law based on the discrete solution from eq. (54) and σ_{ij}^* is the stress component that is continuous in the domain and has to be found. The second integral in eq. (57) is to enforce the continuity of σ_{ij}^* . Each of the smoothed stress components is discretized on the same mesh as the displacement and with the same approximation basis, even though other approximation order can be used

$$\sigma_{ij}^* = \Psi^\sigma \tilde{\sigma}_{ij} \quad \text{and} \quad [\![\sigma_{ij}^*]\!] = [\![\Psi^\sigma]\!] \tilde{\sigma}_{ij} \quad (58)$$

where $\tilde{\sigma}_{ij}$ is the vector of dof for ij stress component.

Each stress component σ_{ij} can be expressed by the displacement gradient

$$\begin{aligned} \sigma_{xx} &= \frac{E}{1-\nu^2} \begin{bmatrix} 1 & 0 \\ 0 & \nu \end{bmatrix} : \nabla \mathbf{u} = \mathbf{E}_{xx} : \mathbf{B}\tilde{\mathbf{u}} \\ \sigma_{yy} &= \frac{E}{1-\nu^2} \begin{bmatrix} \nu & 0 \\ 0 & 1 \end{bmatrix} : \nabla \mathbf{u} = \mathbf{E}_{yy} : \mathbf{B}\tilde{\mathbf{u}} \\ \sigma_{xy} &= \frac{E}{1-\nu^2} \begin{bmatrix} 0 & \frac{1-\nu}{2} \\ \frac{1-\nu}{2} & 0 \end{bmatrix} : \nabla \mathbf{u} = \mathbf{E}_{xy} : \mathbf{B}\tilde{\mathbf{u}} \end{aligned} \quad (59)$$

When the approximation is incorporated into eq. (57) and minimized in respect to $\tilde{\sigma}_{ij}$ a set of linear equations is obtained

$$\mathbf{M}^\sigma \tilde{\sigma}_{ij} = \mathbf{F}_{ij}^\sigma \quad (60)$$

where

$$\mathbf{M}^\sigma = \frac{1}{|V|} \int_V \boldsymbol{\Psi}^{\sigma T} \boldsymbol{\Psi}^\sigma dV + \frac{1}{|S^d|} \int_{S^d} [\![\boldsymbol{\Psi}^\sigma]\!]^T [\![\boldsymbol{\Psi}^\sigma]\!] dS \quad (61)$$

$$\mathbf{F}_{ij}^\sigma = \int_V \boldsymbol{\Psi}^{\sigma T} \mathbf{E}_{ij} : \mathbf{B} dV \tilde{\mathbf{u}} \quad (62)$$

6 Examples

This Section considers two examples that show the correctness and effectiveness of the approach put forward. The first example, Section 6.1, deals with a benchmark problem for which the exact solution is known. The problem is solved using different kinds of meshes, and one of the mesh is un-typical, since it consists of fish-shape cells. In the second example, Section 6.2, the domain with a rectangular hole is analysed. The very high-order approximations are tested here to recover the stress concentrations. In this example the approximation order p reaches up to 60.

6.1 Benchmark example

The following example is motivated by paper [2], where the cantilever beam has been analysed and the results were compared to the analytical solution taken from [35]. The solution has been applied by many researchers to convergence, error analysis and for adaptation purposes, e.g. [41, 30, 5, 27, 4, 23, 14]. The analysed cantilever is of depth D , length L and thickness T . The cantilever, presented in Fig. 3, is supported on the left side, whereas on the right it is subjected to load distributed as parabolic tangent traction. The exact stress solution is [2]

$$\sigma_x = -\frac{P(L-x)y}{I} \quad (63)$$

$$\sigma_y = 0 \quad (64)$$

$$\sigma_{xy} = \frac{P}{2I} \left(\frac{D^2}{4} - y^2 \right) \quad (65)$$

while the exact solution for displacements is

$$u_x(x, y) = -\frac{Py}{6EI} \left(x \cdot (6L - 3x) + (2 + \nu) \left(y^2 - \frac{D^2}{4} \right) \right) \quad (66)$$

$$u_y(x, y) = \frac{P}{6EI} \left(3\nu y^2 \cdot (L - x) + (4 + 5\nu) \frac{D^2 x}{4} + (3L - x) x^2 \right) \quad (67)$$

where E is Young's modulus, ν is Poissons's ratio, I is the cross-section moment of inertia, P is the load multiplier. As it can be noticed the cantilever beam is not strictly fixed on the left side, so a non-homogeneous boundary conditions need to be incorporated

$$\hat{u}_x(y) = -\frac{P(2 + \nu)}{6EI} \left(y^2 - \frac{D^2}{4} \right) y \quad (68)$$

$$\hat{u}_y(y) = \frac{PL\nu}{2EI} y^2 \quad (69)$$

In calculations performed in this work, the cantilever dimensions and other parameters were set the same as in [2], namely: $L = 4$ m, $D = 1$ m, $E = 20$ Gpa, $P = 10$ N, and $\nu = 0.16$. The cantilever beam thickness was set as $T = 0.1$ m so $I = 0.0083$ m⁴. The stress functions σ_{xx} and σ_{xy} on the deformed beam are shown in Fig. 4.

The problem has been solved using four polygonal meshes: with 8 elements, with 30 element, with non-conforming rectangular elements and with fish-like elements, see Fig. 5. It should be noted that for the last mesh (fish-mesh) at the fish tails the finite elements vertices are very sharp. In this example the approximation order was set to $p = 3$ for all the meshes. In each case the exact solution has been obtained. It shows that the proposed version of DG method for the elasticity problem is correct, since even for very un-typical finite elements the method gives the exact solution.

Figure 3: The example cantilever

Figure 4: The exact σ_{xx} and σ_{xy} stresses on the deformed cantilever beam in the benchmark example

(a) Polygonal mesh with 8 finite elements

(b) Polygonal mesh with 30 finite elements

(c) Mesh with non-conforming rectangular elements

(d) Polygonal mesh with fish-like finite elements

Figure 5: Four polygonal meshes for the benchmark example

Figure 6: Cantilever beam with rectangular hole: the geometry, boundary conditions and discretisation by four non-conforming elements. Dimensions in [m]

(a) Mesh with hole made by 4 rectangular elements

(b) Mesh with hole made by 3014 triangular elements

6.2 The cantilever beam with hole

In this example the cantilever beam in plane stress state with the rectangular hole inside is considered. The geometry and boundary conditions are shown in Fig. 6. The material parameters are inherited from the previous example in Section 6.1. In this example the stresses are expected to concentrate on the rectangular hole vertices. Although, usually in such situation the finite element mesh is refined around the points with high stresses but in this work a mesh with only four rectangular finite elements is applied. However, for comparison, a triangular dense mesh (3014 elements) is used in this example. Both of the meshes are shown in Fig. 6.

The first mesh consists of only four non-conforming finite elements and some elements are elongated. The choice of such a mesh was motivated by showing the effectiveness of the proposed method, that even for such a mesh high gradients of stresses located at the hole vertices can be successfully recovered. Due to fact that in the mesh the cells are rectangles, the very high-order approximation can be applied in these elements. For the mesh, the example is solved by approximation orders $p = 4, 10, 20, 40, 60$. By applying the very high-order finite elements, the stress concentrations on the hole vertices can be recovered, even for very rough mesh. In a case of the triangular dense mesh the approximation order in finite elements is set $p = 2$.

The number of degrees of freedom in single elements ($\#edof$) is connected with approximation order p by the following relation

$$\#edof = (p + 1)^2 + (p + 1) \quad (70)$$

As a result the numbers of degrees of freedom ($\#dof$) for the rectangular mesh are: $\#dof = 120, 528, 1848, 6888, 15128$. And for the triangular mesh $\#dof = 36168$.

The outcomes are shown in a form of maps of stress components depicted on the deformed structure for various approximation orders $p = 4, 10, 20, 40, 60$ and for the triangular mesh, see Figs. 8 to 11. The results illustrate the high effectiveness of the DGFD method with very high-order finite elements. The numerical calculations are stable even for elements with $p = 60$. High levels of stresses are located at very near neighbourhood hole vertices, which means extremely high stress gradients. In spite of the fact that, there are only four finite elements in the mesh, the very high-order DGFD method can cope with such great stress gradients. It can be noticed that the results obtained on the rectangular mesh with $p = 40$ and $p = 60$ are comparable to the results on dense triangular mesh.

(a) Stress σ_{xx} for $p = 4$, rectangular mesh

(b) Stress σ_{xx} for $p = 10$, rectangular mesh

(c) Stress σ_{xx} for $p = 20$, rectangular mesh

(d) Stress σ_{xx} for $p = 40$, rectangular mesh

(e) Stress σ_{xx} for $p = 60$, rectangular mesh

(f) Stress σ_{xx} for $p = 2$, triangular mesh

Figure 8: Stress σ_{xx} in [kPa] on the deformed domain obtained on rectangular mesh with $p = 4, 10, 20, 40, 60$ and triangular mesh with $p = 2$

(a) Stress σ_{yy} for $p = 4$, rectangular mesh

(b) Stress σ_{yy} for $p = 10$, rectangular mesh

(c) Stress σ_{yy} for $p = 20$, rectangular mesh

(d) Stress σ_{yy} for $p = 40$, rectangular mesh

(e) Stress σ_{yy} for $p = 60$, rectangular mesh

(f) Stress σ_{yy} for $p = 2$, triangular mesh

Figure 9: Stress σ_{yy} in [kPa] on the deformed domain obtained on rectangular mesh with $p = 4, 10, 20, 40, 60$ and triangular mesh with $p = 2$

(a) Stress σ_{xy} for $p = 4$, rectangular mesh

(b) Stress σ_{xy} for $p = 10$, rectangular mesh

(c) Stress σ_{xy} for $p = 20$, rectangular mesh

(d) Stress σ_{xy} for $p = 40$, rectangular mesh

(e) Stress σ_{xy} for $p = 60$, rectangular mesh

(f) Stress σ_{xy} for $p = 2$, triangular mesh

Figure 10: Stress σ_{xy} in [kPa] on the deformed domain obtained on rectangular mesh with $p = 4, 10, 20, 40, 60$ and triangular mesh with $p = 2$

(a) Von Mises stress for $p = 4$, rectangular mesh

(b) Von Mises stress for $p = 10$, rectangular mesh

(c) Von Mises stress for $p = 20$, rectangular mesh

(d) Von Mises stress for $p = 40$, rectangular mesh

(e) Von Mises stress for $p = 60$, rectangular mesh

(f) Von Mises stress for $p = 2$, triangular mesh

Figure 11: Von Mises stress in [kPa] on the deformed domain obtained on rectangular mesh with $p = 4, 10, 20, 40, 60$ and triangular mesh with $p = 2$

7 Conclusions

This paper presents the application of discontinuous Galerkin with finite difference (DGFD) method to the mechanical problem in plane stress state. The version of DG method utilizes the high order finite difference rules for compatibility conditions on the mesh skeleton. The same kind of finite difference approach is applied to enforce the Dirichlet boundary conditions. The calculations on the mesh skeleton and Dirichlet outer boundary are performed on the local coordinates associated with the normal and tangent directions to mesh skeleton or boundary. There is a great flexibility in choosing basis functions for approximations in DGFD method, so the Chebyshev or Legendre polynomials can be used as a basis functions. However, in this study the Chebyshev polynomials were selected, though the Legendre polynomials give similar results. If the finite element cells are not rectangular, a typical approximation order p up to 8, can be used. But, the very high-order approximation is possible only for rectangular finite elements. In this paper the results are presented for approximation order p up to 60.

The DGFD method can be applied on the mesh with finite elements of quite arbitrary shapes. In the benchmark example four different meshes are used. First two were based on polygonal (pentagonal or hexagonal) finite elements. In the third mesh the cells are non-conforming. The cells in the fourth mesh have the shape of fish. For all of the meshes the exact solution of the benchmark examples was recovered. In the benchmark example the fish-mesh consists of fish-like cells and the cells are elongated into x direction, the cells are not concave and some vertices are very sharp. In spite of such extraordinary shapes of cells, the correct final results were obtained.

The second example regards the domain with a rectangular hole in which the stress concentrations on the hole vertices were observed. This case presents very high-ordered finite elements in which the approximation order is set to $p = 4, 10, 20, 40$ and 60 . It was possible to obtain such high value of p by using the Chebyshev polynomials as basis functions. In this example the mesh consisted only of four non-conforming rectangular finite elements but high values of stresses and high values of stresses gradients were perfectly recovered. For comparative reasons, a typical dense triangular mesh with second order finite elements was also used. The results for the triangular mesh are fully comparable with the very high-order rectangular mesh.

The very high-order DGFD method gives new opportunities in the analysis of mechanical problems. The p -type refinement can achieve a new dimension, because the level p is practically unlimited. Moreover, in DGFD method the neighbouring finite elements can have different values of p and the solution remains continuous.

References

- [1] Paola F. Antonietti, Paul Houston, Marco Sarti, and Marco Verani. Multigrid algorithms for hp-version interior penalty discontinuous Galerkin methods on polygonal and polyhedral meshes. *arXiv:1412.0913 [math.NA]*, page 26 pages, 2014.
- [2] Charles E. Augarde and Andrew J. Deeks. The use of Timoshenko’s exact solution for a cantilever beam in adaptive analysis. *Finite Elem. Anal. Des.*, 44(9-10):595–601, June 2008.
- [3] F. Bassi, L. Botti, A. Colombo, A. Crivellini, A. Ghidoni, and F. Massa. On the development of an implicit high-order discontinuous Galerkin method for DNS and implicit LES of turbulent flows. *European Journal of Mechanics - B/Fluids*, 55, Part 2:367 – 379, 2016. Vortical Structures and Wall Turbulence.
- [4] T. Belytschko, Y. Y. Lu, and L. Gu. Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering*, 37(2):229–256, 1994.
- [5] M.B. Bergallo, C.E. Neuman, and V.E. Sonzogni. Composite mesh concept based FEM error estimation and solution improvement. *Computer Methods in Applied Mechanics and Engineering*, 188(4):755–774, 2000. IVth World Congress on Computational Mechanics. (II). Optimum.
- [6] Hadrien Bériot, Albert Prinn, and Gwénaél Gabard. Efficient implementation of high-order finite elements for Helmholtz problems. *International Journal for Numerical Methods in Engineering*, 10, 2016.
- [7] Andrea Cangiani, Emmanuil H. Georgoulis, and Paul Houston. hp-Version discontinuous Galerkin methods on polygonal and polyhedral meshes. *Mathematical Models and Methods in Applied Sciences*, 24(10):2009–2041, 2014.
- [8] W. Cecot and M. Oleksy. High order FEM for multigrid homogenization. *Computers & Mathematics with Applications*, 70(7):1391 – 1400, 2015. High-Order Finite Element and Isogeometric Methods.

- [9] B. Bernardo Cockburn, George Karniadakis, and Chi-Wang Shu, editors. *Discontinuous Galerkin methods : theory, computation, and applications*. Lecture notes in computational science and engineering. Springer, Berlin, New York, 2000.
- [10] D.A. Di Pietro and A. Ern. *Mathematical Aspects of Discontinuous Galerkin Methods*. Mathématiques et Applications. Springer Berlin Heidelberg, 2011.
- [11] Evan S. Gawlik and Adrian J. Lew. High-order finite element methods for moving boundary problems with prescribed boundary evolution. *Computer Methods in Applied Mechanics and Engineering*, 278:314 – 346, 2014.
- [12] Stefano Giani. Solving elliptic eigenvalue problems on polygonal meshes using discontinuous Galerkin composite finite element methods. *Applied Mathematics and Computation*, pages –, 2015.
- [13] Jan S. Hesthaven and Tim Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2007.
- [14] D.A. Hu, S.Y. Long, K.Y. Liu, and G.Y. Li. A modified meshless local Petrov–Galerkin method to elasticity problems in computer modeling and simulation. *Engineering Analysis with Boundary Elements*, 30(5):399–404, 2006.
- [15] J. Jaśkowiec. Discontinuous Galerkin method on reference domain. *Computer Assisted Methods in Engineering and Science*, page submitted in February, 2016.
- [16] J. Jaśkowiec. The discontinuous Galerkin method with higher degree finite difference compatibility conditions and arbitrary local and global basis functions. *Computer Assisted Methods in Engineering and Science*, page submitted in February, 2016.
- [17] J. Jaśkowiec. The hp nonconforming mesh refinement in discontinuous Galerkin finite element method based on Zienkiewicz-Zhu error estimation. *Computer Assisted Methods in Engineering and Science*, page submitted in February, 2016.
- [18] J. Jaśkowiec. Very-high order discontinuous Galerkin method. *Journal of Computational Physics*, page submitted in February, 2016.
- [19] J. Jaśkowiec, P. Pluciński, and A. Stankiewicz. Discontinuous Galerkin method with arbitrary polygonal finite elements. *Finite Elements in Analysis and Design*, page submitted in September, 2015.
- [20] Yingjun Jiang and Jingtang Ma. High-order finite element methods for time-fractional partial differential equations. *Journal of Computational and Applied Mathematics*, 235(11):3285 – 3290, 2011.
- [21] G. Kitzler and J. Schberl. A high order space–momentum discontinuous Galerkin method for the boltzmann equation. *Computers & Mathematics with Applications*, 70(7):1539 – 1554, 2015. High-Order Finite Element and Isogeometric Methods.
- [22] Moritz Kompenhans, Gonzalo Rubio, Esteban Ferrer, and Eusebio Valero. Adaptation strategies for high order discontinuous Galerkin methods based on tau-estimation. *Journal of Computational Physics*, 306:216 – 236, 2016.
- [23] G.R Liu and Z.H Tu. An adaptive procedure based on background cells for meshless methods. *Computer Methods in Applied Mechanics and Engineering*, 191(1718):1923–1943, 2002.
- [24] Hong Luo, Joseph D. Baum, and Rainald Lhner. A discontinuous Galerkin method based on a taylor basis for the compressible flows on arbitrary grids. *Journal of Computational Physics*, 227(20):8875 – 8893, 2008.
- [25] Éric Madaule, Marco Restelli, and Eric Sonnendrücker. Energy conserving discontinuous Galerkin spectral element method for the vlasovpoisson system. *Journal of Computational Physics*, 279:261 – 288, 2014.
- [26] G. Mengaldo, D. De Grazia, D. Moxey, P.E. Vincent, and S.J. Sherwin. Dealiasing techniques for high-order spectral element methods on regular and irregular grids. *Journal of Computational Physics*, 299:56 – 81, 2015.
- [27] Y. Miao, Y.H. Wang, and F. Yu. Development of hybrid boundary node method in two-dimensional elasticity. *Engineering Analysis with Boundary Elements*, 29(7):703–712, 2005.

- [28] Lin Mu, Junping Wang, Yanqiu Wang, and Xiu Ye. Interior penalty discontinuous Galerkin method on very general polygonal and polyhedral meshes. *Journal of Computational and Applied Mathematics*, 255:432 – 440, 2014.
- [29] Lin Mu, Xiaoshen Wang, and Yanqiu Wang. Shape regularity conditions for polygonal/polyhedral meshes, exemplified in a discontinuous Galerkin discretization. *Numerical Methods for Partial Differential Equations*, 31(1):308–325, 2015.
- [30] Hyung-Seok Oh and R.C. Batra. Application of Zienkiewicz–Zhu’s error estimate with superconvergent patch recovery to hierarchical p-refinement. *Finite Elements in Analysis and Design*, 31(4):273–280, 1999.
- [31] Marta Oleksy and Witold Cecot. Application of hp-adaptive finite element method to two-scale computation. *Archives of Computational Methods in Engineering*, (Vol. 22, Iss. 1):105–134, 2015.
- [32] W.H. Reed and T.R. Hill. *Triangular mesh methods for the neutron transport equation*. Oct 1973.
- [33] S. M. Renda, R. Hartmann, C. De Bartolo, and M. Wallraff. A high-order discontinuous Galerkin method for all-speed flows. *International Journal for Numerical Methods in Fluids*, 77(4):224–247, 2015.
- [34] B. Rivière. *Discontinuous Galerkin Methods for Solving Elliptic and Parabolic Equations*. Society for Industrial and Applied Mathematics, 2008.
- [35] S. Timoshenko. *Theory of elasticity*. Engineering societies monographs. McGraw-Hill, 1951.
- [36] Kevin P Vincent, Matthew J Gonzales, Andrew K Gillette, Christopher T Villongco, Simone Pezzuto, Jeffrey H Omens, Michael J Holst, and Andrew D McCulloch. High-order finite element methods for cardiac monodomain simulations. *Frontiers in Physiology*, 6(217), 2015.
- [37] Junping Wang and Xiu Ye. A weak Galerkin mixed finite element method for second order elliptic problems. *Mathematics of Computation*, 83:2101–2126, 2014.
- [38] Zhi Jian Wang. A perspective on high-order methods in computational fluid dynamics. *Science China Physics, Mechanics & Astronomy*, 59:614701–1 – 614701–6, January 2016.
- [39] D. Wirasaet, E.J. Kubatko, C.E. Michoski, S. Tanaka, J.J. Westerink, and C. Dawson. Discontinuous Galerkin methods with nodal and hybrid modal/nodal triangular, quadrilateral, and polygonal elements for nonlinear shallow water flow. *Computer Methods in Applied Mechanics and Engineering*, 270:113 – 149, 2014.
- [40] Ling Yuan and Chi-Wang Shu. Discontinuous Galerkin method based on non-polynomial approximation spaces. *Journal of Computational Physics*, 218(1):295–323, 2006.
- [41] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. part 1: The recovery technique. *International Journal for Numerical Methods in Engineering*, 33(7):1331–1364, 1992.