



# Stabilization-free virtual element method for 3D hyperelastic problems

Bing-Bing Xu<sup>1</sup> · Fan Peng<sup>2</sup> · Peter Wriggers<sup>1</sup>

Received: 1 March 2024 / Accepted: 14 May 2024  
© The Author(s) 2024

## Abstract

In this work, we present a first-order stabilization-free virtual element method (SFVEM) for three-dimensional hyperelastic problems. Different from the conventional virtual element method, which necessitates additional stabilization terms in the bilinear formulation, the method developed in this work operates without the need for any stabilization. Consequently, it proves highly suitable for the computation of nonlinear problems. The stabilization-free virtual element method has been applied in two-dimensional hyperelasticity and three-dimensional elasticity problems. In this work, the method will be applied to three-dimensional hyperelasticity problems for the first time. Similar to the techniques used in the two-dimensional stabilization-free virtual element method, the new virtual element space is modified to allow the computation of the higher-order  $L_2$  projection of the gradient. This paper reviews the calculation process of the traditional  $\mathcal{H}_1$  projection operator; and describes in detail how to calculate the high-order  $L_2$  projection operator for three-dimensional problems. Based on this high-order  $L_2$  projection operator, this paper extends the method to more complex three-dimensional nonlinear problems. Some benchmark problems illustrate the capability of the stabilization-free VEM for three-dimensional hyperelastic problems.

**Keywords** Virtual element method · Stabilization free · Elastoplasticity · Nonlinear problems

## 1 Introduction

For a wide range of engineering and scientific problems governed by partial differential equations, numerical methods are an effective tool for simulating and solving such problems. The Virtual Element Method (VEM), as introduced in [1, 2], has received more and more attention due to its greater flexibility in the fields of the element shape and the definition of the degrees of freedom. As a generalization of the Finite Element Method (FEM), the virtual element method allows very general arbitrarily shaped elements, which makes the method suitable for calculating some special problems such

as crack propagation, hanging nodes, and adaptive meshes. In the framework of structural mechanics, the method has been applied to different fields including linear elastic problems [3–7], hyperelastic materials at finite deformations [8–11], contact problems [12–15], phase field fracture [16, 17], elastodynamics problems [18–21], and finite elastoplastic deformations [22–24]. Specific work of VEM in engineering can be found in the latest published VEM book [25]. At the same time, there are also other numerical methods that use polygonal or polyhedral meshes for calculations, such as Polygonal Finite Element Method (PFEM) [26, 27], Discontinuous Galerkin Methods (DG) [28, 29], Polygonal Smoothed Finite Element Method (S-FEM) [30, 31]. However, the above-mentioned methods have general difficulties when dealing with three-dimensional problems. For example, the above polygonal finite element methods have great difficulties when solving nonconvex polyhedrons or dealing with high-order elements. Generally speaking, VEM has higher advantages when dealing with complex-shaped elements and high-order elements.

Given the flexibility of VEM, it is interesting to develop a new virtual element format for three-dimensional hyperelastic solids. According to previous relevant descriptions [1, 2],

---

✉ Bing-Bing Xu  
bingbing.xu@ikm.uni-hannover.de  
Fan Peng  
pengfan33@chd.edu.cn  
Peter Wriggers  
wriggers@ikm.uni-hannover.de

<sup>1</sup> Institute of Continuum Mechanics, Leibniz University Hannover, 30823 Hannover, Niedersachsen, Germany

<sup>2</sup> School of Science, Chang'an University, Xi'an 710064, Shaanxi, China

the main idea of VEM is a split of the variable  $u_h$  into a projection part  $\Pi u$  and a remainder, that is  $u_h = \Pi u + (u_h - \Pi u)$ . This formulation requires additional stabilization terms in VEM to ensure that the element stiffness matrix has the correct rank (for elements with more than 4 nodes). The selection of stabilization terms, especially for nonlinear problems, is a complicated issue. Different formulations can be found in [3, 7], which are extended to energy stabilization in [9, 25, 32]. However, a virtual element that does not require any stabilization will be more flexible and reliable for the solution of nonlinear problems.

With this aim, we are going to construct a three-dimensional stabilization-free virtual element method and use it to solve hyperelastic problems. In conventional VEM, the stabilization term is needed because the strain order is too low to accurately describe the energy within the element. By increasing the order of the strain polynomial, the stabilization-free virtual element method (SFVEM) [33] has been recently proposed and has been successfully used in different fields [34–36]. However, these applications are usually limited to two-dimensional linear problems. Recently, the author developed SFVEM for hyperelastic problems [37], but it is still for two-dimensional problems. Similar to the two-dimensional problem, by using the high-order  $L_2$  projection operator  $\Pi_{l,E}^0 \nabla$  of the gradient of function in  $\mathcal{H}^1(E)$ , we can construct a stabilization-free virtual element format for three-dimensional problems. The first work on stabilization-free 3D virtual elements can be found in [38], but it is mainly aimed at heat transfer problems. In this work, the virtual element space is modified to allow the computation of a higher-order  $L_2$  projection operator. Based on that the SFVEM for three-dimensional problems will be developed and applied to hyperelasticity problems for the first time. In addition, this work will also use the arc length method to solve unstable structures.

Based on the above description, we can divide the paper into the following parts. In Sect. 2, the governing equations for hyperelastic problems are reviewed. Then, the three-dimensional stabilization-free virtual element spaces will be given in Sect. 3 for arbitrary order  $k$ . Based on the above definitions, the  $\mathcal{H}_1$  projection operator and higher-order  $L_2$  projection operator are developed in Sect. 4. The virtual element discretization will be described for 3D hyperelastic problems in Sect. 5. Numerical examples are presented and discussed in Sect. 6. The paper closes with some concluding remarks in Sect. 7.

## 2 Governing equations for finite elasticity

Consider a three-dimensional elastic solid that is deforming from the initial configuration occupying a volume  $\mathcal{B}_0 \in \mathbb{R}^3$  with boundary  $\Gamma_0 = \partial \mathcal{B}_0$  into the current configuration  $\mathcal{B}_t$

with boundary  $\partial \mathcal{B}_t$  (see Fig. 1). The position  $\mathbf{x}$  of a material point  $\mathcal{P}$  initially at  $\mathbf{X}$  is given by the motion

$$\mathbf{x} = \boldsymbol{\varphi}(\mathbf{X}, t) = \mathbf{X} + \mathbf{u}(\mathbf{X}, t), \tag{1}$$

where  $\mathbf{u}$  is the displacement field. We denote by  $\mathbf{F}$  the deformation gradient defined by

$$\mathbf{F} = \text{Grad} \boldsymbol{\varphi} = \mathbf{1} + \nabla \mathbf{u}, \tag{2}$$

where  $\mathbf{1}$  represents the second-order unity tensor and  $\nabla \mathbf{u}$  is the displacement gradient with respect to the initial coordinates. Based on  $\mathbf{F}$ , the right Cauchy-Green deformation tensor is given by

$$\mathbf{C} = \mathbf{F}^T \cdot \mathbf{F}. \tag{3}$$

Further, we can define the Green-Lagrange strain tensor

$$\mathbf{E} = \frac{1}{2} (\mathbf{C} - \mathbf{1}). \tag{4}$$

For the static case, the equilibrium requires

$$-\text{Div} \mathbf{P} = \mathbf{f}, \tag{5}$$

where  $\mathbf{P}$  is the first Piola–Kirchhoff stress and  $\mathbf{f}$  is the body force. The Dirichlet and Neumann boundary conditions are

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on} \quad \Gamma_D, \tag{6}$$

$$\mathbf{P} \cdot \mathbf{N} = \bar{\mathbf{t}} \quad \text{on} \quad \Gamma_N, \tag{7}$$

where  $\mathbf{N}$  is the outward normal vector of  $\partial \mathcal{B}_0$ .

The second Piola–Kirchhoff stress is often used in constitutive modeling since it is work conjugated to the Green-Lagrange strain tensor. The second Piola–Kirchhoff stress can be obtained from  $\mathbf{P}$  as

$$\mathbf{S} = \mathbf{F}^{-1} \cdot \mathbf{P}, \tag{8}$$

where  $\boldsymbol{\sigma}$  is the Cauchy stress tensor (real stress),  $J$  is the determinant of the deformation gradient tensor

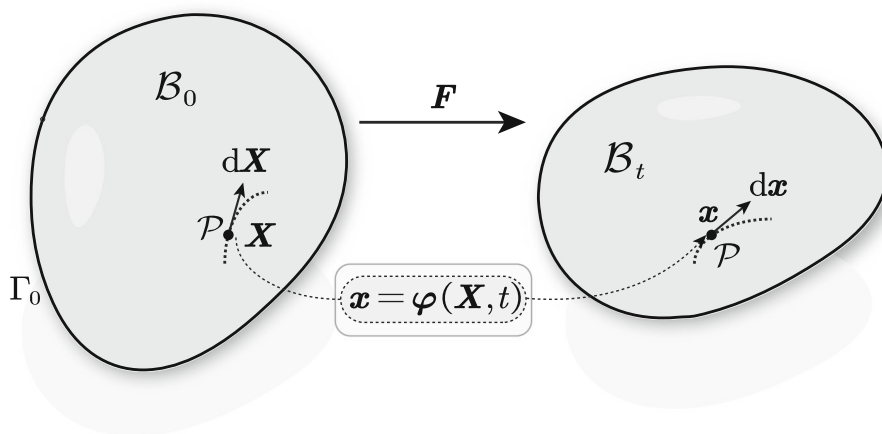
$$J = \det(\mathbf{F}). \tag{9}$$

The work conjugate relationship can be written as

$$\mathcal{W} = \int_{V_0} \delta \mathbf{E} : \mathbf{S} dV_0 = \int_{V_0} \delta \mathbf{F} : \mathbf{P} dV_0, \tag{10}$$

The virtual element formulation for a hyperelastic material can start from the potential energy function directly. By

**Fig. 1** Motion of a body from the initial configuration  $\mathcal{B}_0$  to the current configuration  $\mathcal{B}_t$



introducing a strain energy function  $\Psi(\mathbf{u})$  for elastic problems, the second Piola–Kirchhoff stresses follow

$$\mathbf{S} = 2 \frac{\partial \Psi(\mathbf{u})}{\partial \mathbf{C}}. \tag{11}$$

For a homogeneous compressible isotropic hyperelastic material, the strain energy function  $\Psi(\mathbf{u})$  for the Neo-Hookean hyperelastic model can be adopted as

$$\Psi = \frac{\mu}{2} (I_C - 3) - \mu \ln J + \frac{\lambda}{2} (\ln J)^2, \tag{12}$$

where  $I_C$  is the first invariant of the right Cauchy–Green tensor in Eq. (3),  $J$  is the determinant of  $\mathbf{F}$ ,  $\lambda$  and  $\mu$  are the two Lamé parameters.

Considering  $\frac{\partial J}{\partial \mathbf{C}} = \frac{1}{2} J \mathbf{C}^{-1}$ , the second Piola–Kirchhoff stress tensor  $\mathbf{S}$  can be obtained for the Neo-Hookean hyperelastic model (Eq. (12)) as

$$\begin{aligned} \mathbf{S} &= 2 \frac{\partial \Psi}{\partial \mathbf{C}} = \mu \mathbf{1} - 2\mu \frac{1}{J} \frac{\partial J}{\partial \mathbf{C}} + 2\lambda (\ln J) \frac{1}{J} \frac{\partial J}{\partial \mathbf{C}} \\ &= \mu (\mathbf{1} - \mathbf{C}^{-1}) + \lambda (\ln J) \mathbf{C}^{-1}. \end{aligned} \tag{13}$$

By differentiating the second Piola–Kirchhoff stress  $\mathbf{S}$ , the constitutive tensor  $\mathcal{D}$  can be deduced as

$$\mathcal{D} = 2 \frac{\partial \mathbf{S}}{\partial \mathbf{C}} = \lambda \mathbf{C}^{-1} \otimes \mathbf{C}^{-1} + 2(\mu - \lambda \ln J) \mathcal{L}, \tag{14}$$

with the components of  $\mathcal{L}$

$$\begin{aligned} \mathcal{L}_{IJKL} &= -\frac{\partial (\mathbf{C}^{-1})_{IJ}}{\partial C_{KL}} \\ &= \frac{1}{2} [(\mathbf{C}^{-1})_{IK} (\mathbf{C}^{-1})_{JL} + (\mathbf{C}^{-1})_{IL} (\mathbf{C}^{-1})_{JK}]. \end{aligned} \tag{15}$$

### 3 3D stabilization-free virtual element spaces

In this part, we describe the basic ideas of the three-dimensional stabilization-free virtual element method (SFVEM).

#### 3.1 Notation

For three-dimensional problems, a decomposition  $\mathcal{T}_h = \{\Omega_E\}_h$  is introduced where  $\Omega_E$  is a partition of the computational domain  $\Omega$  into non-overlapping polyhedron elements. For each polyhedron element  $E$  with boundary  $\partial E$  as shown in Fig. 2, the parameters are given as: volume  $|E|$ , barycenter  $\mathbf{x}_E = (x_E, y_E, z_E)^T$ , and diameter  $h_E$ . Besides, an element face  $F \in \partial E$  is a planar and a two-dimensional subset of  $\mathbb{R}^3$ . We denote the set of polygon faces by  $\mathcal{F}_h$  as shown in Fig. 2. By defining a local coordinate system  $(\xi, \eta)$ , the parameters for the local polygon face are given as area  $|F|$ , barycenter  $\xi_F = (\xi_F, \eta_F)^T$ , and diameter  $h_F = \sup_{\mathbf{x}, \mathbf{y} \in F} |\mathbf{x} - \mathbf{y}|$ . Specific requirements for 3D virtual element mesh can be found in [39].

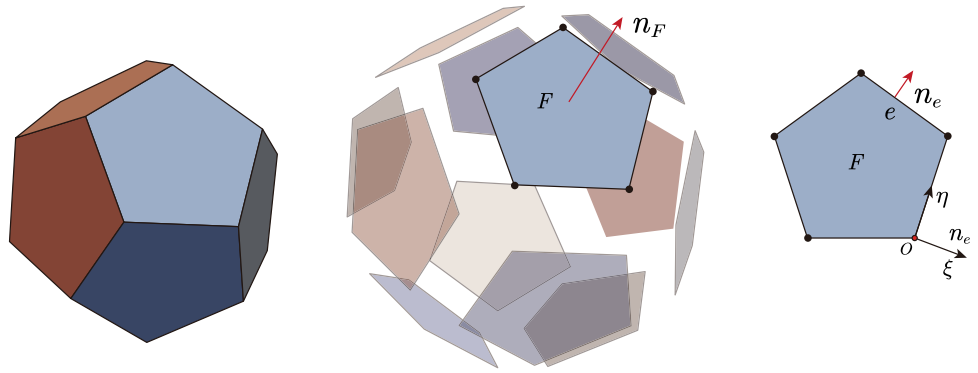
For three-dimensional problems ( $d = 3$ ), we introduce scaled monomials  $\mathcal{M}_k(E)$  as

$$\mathcal{M}_k(E) := \left\{ m_\alpha = \left( \frac{\mathbf{x} - \mathbf{x}_E}{h_E} \right)^\alpha \text{ for } \alpha \in \mathbb{N}^d \text{ with } |\alpha| \leq k \right\}, \tag{16}$$

where  $\alpha = (\alpha_1, \alpha_2, \alpha_3)$  is a multiindex and  $\mathbf{x}^\alpha := x_1^{\alpha_1} \dots x_d^{\alpha_d}$ . The dimension of the given polygonal function space (Eq. (16)) is

$$N_k := \dim(\mathbb{P}_k(E)) = \frac{(k+1)(k+2)(k+3)}{6} \tag{17}$$

**Fig. 2** A polyhedral element and its polygonal surfaces



Using local coordinates, the scaled monomials  $\mathcal{M}_k^f(F)$  for the polygon face  $F$  of a polyhedron can be defined as

$$\mathcal{M}_k^f(F) := \left\{ m_\alpha^f = \left( \frac{\xi - \xi_F}{h_F} \right)^\alpha \text{ for } \alpha \in \mathbb{N}^{d-1} \text{ with } |\alpha| \leq k \right\}, \quad (18)$$

with the dimension

$$N_k^f := \dim(\mathbb{P}_k(F)) = \frac{(k+1)(k+2)}{2} \quad (19)$$

### 3.2 Virtual element space

For a given three-dimensional polyhedral element  $E$  with boundary  $\partial E$  and  $F \in \partial E$  a polygonal face of  $E$ , we start by defining the local lifting spaces on each face  $F$

$$\tilde{\mathcal{V}}_k(F) := \left\{ u_h \in \mathcal{H}^1(F) \cap C^0(F) : \Delta_F u_h \in \mathbb{P}_k(F), u_h|_e \in \mathbb{P}_k(e), e \in \partial F \right\}, \quad (20)$$

where  $\Delta_F$  is the 2D Laplacian over the face  $F$ . In this work, a low-order Ansatz is adopted for the construction of the stabilization-free virtual element formulation ( $k = 1$ ) and the scaled monomials  $m_\alpha^f \in \mathcal{M}_k^f(F)$  (see Eq.(18)) have the explicit form as

$$m_1^f = 1, \quad m_2^f = \frac{\xi - \xi_F}{h_F}, \quad m_3^f = \frac{\eta - \eta_F}{h_F}. \quad (21)$$

The degrees of freedom can be selected as the value  $u_h(v)$  for the vertex  $F$ . We can define the  $\mathcal{H}_1$  projection operator  $\Pi_{k,F}^\nabla : \tilde{\mathcal{V}}_k(F) \rightarrow \mathbb{P}_k(F)$  using

$$\int_F \nabla_F \Pi_{k,F}^\nabla u_h \cdot \nabla_F p \, d\Omega_F = \int_F \nabla_F u_h \cdot \nabla_F p \, d\Omega_F, \quad \forall p \in \mathbb{P}_k(F), \quad (22)$$

$$P_0 \left( \Pi_{k,F}^\nabla u_h - u_h \right)_F = 0. \quad (23)$$

Integration by parts allows the evaluation of projection operator  $\Pi_{k,F}^\nabla$ , details can be found in [37, 40]. The higher-order moments in Eq. (20) can be approximated so the dimension of the function space does not change [41]. Then we can define the two-dimensional virtual element space

$$\begin{aligned} \mathcal{V}_k^h(F) &:= \left\{ u_h \in \tilde{\mathcal{V}}_k(F) : \int_F u_h p \, d\Omega_F \right. \\ &= \left. \int_F \left( \Pi_{k,F}^\nabla u_h p \right) \, d\Omega_F, \quad \forall p \in \mathbb{P}_k(E) \right\}. \end{aligned} \quad (24)$$

For the 3D polyhedral element  $E$ , we consider the preliminary virtual space

$$\begin{aligned} \tilde{\mathcal{V}}_k(E) &:= \left\{ u_h \in \mathcal{H}^1(E) \cap C^0(E) : \Delta u_h \right. \\ &= \left. \mathbb{P}_k(E), u_h|_e \in \tilde{\mathcal{V}}_k(F), F \in \partial E \right\}. \end{aligned} \quad (25)$$

Next, we can compute the  $\mathcal{H}_1$  projection operator  $\Pi_{k,E}^\nabla : \tilde{\mathcal{V}}_k(E) \rightarrow \mathbb{P}_k(E)$  using

$$\int_E \nabla \Pi_{k,E}^\nabla u_h \cdot \nabla p \, d\Omega = \int_E \nabla u_h \cdot \nabla p \, d\Omega, \quad \forall p \in \mathbb{P}_k(E), \quad (26)$$

$$P_0 \left( \Pi_{k,E}^\nabla u_h - u_h \right) = 0. \quad (27)$$

Then we can define the local virtual element space

$$\begin{aligned} \mathcal{V}_k^h(E) &:= \left\{ u_h \in \tilde{\mathcal{V}}_k(E) : \int_E u_h p \, d\Omega \right. \\ &= \left. \int_E \left( \Pi_{k,E}^\nabla u_h \right) p \, d\Omega, \quad \forall p \in \mathbb{P}_k(E) \right\}. \end{aligned} \quad (28)$$

The projection operator  $\Pi_{1,E}^\nabla$  depends only on its vertex values for  $k = 1$ . Thus the function space can be uniquely characterized by the vertex values (named degrees of freedom). According to the above space (28), we can construct a discrete form of the classic three-dimensional VEM. This consistency part requires additional stabilization for a correct rank of the element stiffness matrix.

To construct a virtual element formulation without any stabilization for three-dimensional problems, we need to introduce an additional enhancement space for  $k = 1$ . Inspired by Ref. [42], we define a local enlarged enhancement virtual element space based on the higher-order polynomial projection

$$\mathcal{V}_{1,l}(E) := \left\{ u_h \in \mathcal{H}^1(E) : \Delta u_h \in \mathbb{P}_{l+1}(E), \int_E u_h p \, d\Omega = \int_E \Pi_{1,E}^\nabla u_h p_{l+1} \, d\Omega, \forall p_{l+1} \in \mathbb{P}_{l+1}(E) \right\} \quad (29)$$

Again the degrees of freedom can be selected as the values at  $N_E^V$  vertices in  $E$  (same as the d.o.f used in  $\mathcal{V}_k^h(E)$ ). The order of the polynomial ( $l$  in Eq. (29)) is related to the number of vertices  $N_E^V$ . For two-dimensional problems, the selection of  $l$  to ensure well-posedness can be found in [33, 35, 43, 44]. For three-dimensional problems, the selection of  $l$  has been based on an eigenvalue analysis, and the relationship between the order  $l$  and the number of vertices  $N_E^V$  can be found in our previous work [45]. As discussed in [45], the order  $l$  can be selected as  $l = 1 \sim 2$  for most of the elements.

### 4 Projection operators

The  $\mathcal{H}_1$  projection operator  $\Pi_{1,F}^\nabla : \tilde{\mathcal{V}}_1(E) \rightarrow \mathbb{P}_1(E)$  was introduced in the previous section. This projection operator is known from conventional VEM where a stabilization term was to be introduced to ensure the correct rank of the element stiffness matrix. In this work, we construct a virtual element formulation without any stabilization for 3D problems. As shown in Eq. (29), the  $\mathcal{H}^1$  projection operator is necessary in the local enlarged enhancement virtual element space for the approximation of higher order moments. So in this part, we need to review how to calculate the  $\mathcal{H}_1$  projection operator  $\Pi_{1,E}^\nabla$  for three-dimensional problems, and further explain how to calculate the  $L_2$  projection operator  $\Pi_{l,E}^0 \nabla$ .

#### 4.1 $\mathcal{H}_1$ projection operator on polyhedral elements

In the present work, only the first-order element ( $k = 1$ ) is considered. The scaled monomials  $m_\alpha \in \mathcal{M}_1(E)$  have the form

$$m_1 = 1, \quad m_2 = \frac{x - x_E}{h_E}, \quad m_3 = \frac{y - y_E}{h_E}, \quad m_4 = \frac{z - z_E}{h_E}. \quad (30)$$

According to the previous definition, the projector  $\Pi_{1,E}^\nabla : \mathcal{V}_1(E) \rightarrow \mathbb{P}_1(E)$  can be calculated by

$$\int_E \nabla \Pi_{1,E}^\nabla u_h \cdot \nabla p \, d\Omega = \int_E \nabla u_h \cdot \nabla p \, d\Omega, \forall p \in \mathbb{P}_1(E) \quad (31)$$

with the additional condition

$$\sum_{i=1}^{n_E} \left( \Pi_{k,E}^\nabla u_h - u_h \right) = 0, \quad (32)$$

where  $n_E$  is the number of vertices. Considering the Green formula for Eq. (31), we have

$$\int_E \nabla \Pi_{1,E}^\nabla u_h \cdot \nabla p \, d\Omega = - \int_E u_h \cdot \Delta p \, d\Omega + \int_{\partial E} u_h \cdot \frac{\partial p}{\partial \mathbf{n}} \, d\Gamma, \quad (33)$$

where the projection operator  $\Pi_{1,E}^\nabla$  can be expanded in a matrix form

$$\Pi_{1,E}^\nabla \boldsymbol{\phi}^T = \mathbf{m}^T \mathbf{\Pi}_{1*,E}^\nabla = \boldsymbol{\phi}^T \mathbf{\Pi}_{1,E}^\nabla, \quad (34)$$

where  $\boldsymbol{\phi}$  is the unknown basis function vector,  $\mathbf{\Pi}_{1*,E}^\nabla$  is matrix formulation of the Ritz projection operator. Substituting Eq. (34) into Eq. (33), yields

$$\int_E \nabla \mathbf{m} \cdot \nabla \mathbf{m}^T \, d\Omega \mathbf{\Pi}_{1*,E}^\nabla = - \int_E \Delta \mathbf{m} \cdot \boldsymbol{\phi}^T \, d\Omega + \sum_{F \subset \partial E} \int_F (\nabla \mathbf{m} \cdot \mathbf{n}_F) \boldsymbol{\phi}^T \, d\Gamma, \quad (35)$$

which leads to an equation system for  $\mathbf{\Pi}_{1*,E}^\nabla$

$$\mathbf{G}^\nabla \mathbf{\Pi}_{1*,E}^\nabla = \mathbf{B}^\nabla, \quad (36)$$

where

$$\mathbf{G}^\nabla := \int_E \nabla \mathbf{m} \cdot \nabla \mathbf{m}^T \, d\Omega, \quad (37)$$

$$\mathbf{B}^\nabla := \sum_{F \subset \partial E} \int_F (\nabla \mathbf{m} \cdot \mathbf{n}_F) \boldsymbol{\phi}^T \, d\Gamma. \quad (38)$$

Since the surface  $F$  of a polyhedral element  $E$  is a polygonal element, see Fig. 2, the shape function  $\boldsymbol{\phi}$  on the surface is still unknown. In this aspect, the three-dimensional problem differs from the two-dimensional problem. Based on the definition of the 3D virtual space (Eq. (25)), the unknown shape function  $\boldsymbol{\phi}$  in  $\mathbf{B}^\nabla$  can be calculated by

$$\begin{aligned} \mathbf{B}^\nabla &= \sum_{F \subset \partial E} \int_F (\nabla \mathbf{m} \cdot \mathbf{n}_F) \boldsymbol{\phi}^T d\Gamma \\ &= \sum_{F \subset \partial E} \left[ (\nabla \mathbf{m} \cdot \mathbf{n}_F) \int_F (\mathbf{m}^f)^T d\Gamma \cdot \boldsymbol{\Pi}_{1^*, F}^\nabla \right], \end{aligned} \quad (39)$$

where  $\mathbf{m}^f$  and  $\boldsymbol{\Pi}_{1^*, F}^\nabla$  are the scaled monomial vector and Ritz projection matrix with respect to local coordinates  $(\xi, \eta)$ . For  $k = 1$ , we can directly calculate the matrix  $\mathbf{G}^\nabla$  through the volume of the unit  $|E|$ , and the matrix  $\mathbf{B}^\nabla$  can be simplified as

$$\mathbf{B}^\nabla = \sum_{F \subset \partial E} \left[ (\nabla \mathbf{m} \cdot \mathbf{n}_F) \cdot [|F|, 0, 0] \cdot \boldsymbol{\Pi}_{1^*, F}^\nabla \right], \quad (40)$$

where  $|F|$  is the area of  $F$ . Considering the consistency condition in Eq. (32), the Ritz projection matrix is obtained by solving Eq. (36).

## 4.2 $L_2$ projection operator on polyhedral elements

Based on the  $L_2(E)$  projection  $\boldsymbol{\Pi}_{l, E}^0 \nabla$  of the gradient of variables in  $\mathcal{V}_{1, l}(E)$  (see Eq. (29)) to  $[\mathbb{P}_l(E)]^3$ , this projection can be explicitly expressed as

$$\int_E \mathbf{p}_l^T \boldsymbol{\Pi}_{l, E}^0 \nabla u_h d\Omega = \int_E \mathbf{p}_l^T \nabla u_h d\Omega. \quad (41)$$

Integrating by parts of the right-hand side of Eq. (41), we can obtain

$$\int_E \mathbf{p}_l^T \nabla u_h d\Omega = \int_{\partial E} (\mathbf{p}_l^T \cdot \mathbf{n}_F) u_h d\Gamma - \int_E (\operatorname{div} \mathbf{p}_l) u_h d\Omega. \quad (42)$$

The higher-order moments can be considered in the function space  $\mathcal{V}_{1, l}(E)$  [41], which also means that we need to calculate the  $\mathcal{H}_1$  projection operator.

The key idea for the stabilization-free formulation is to use higher-order polynomials for the gradient, which results in a mixed-form method. For the gradient  $\nabla u_h$  and the polynomial  $\mathbf{p}_l$ , we follow the construction in [35, 36], the variable field  $u_h$  and the gradient can be expanded as

$$\boldsymbol{\Pi}_{l, E}^0 \nabla u_h = (\mathbf{N}^p)^T \boldsymbol{\Pi}^m \tilde{\mathbf{u}}, \quad \mathbf{p}_l = (\mathbf{N}^p)^T \tilde{\mathbf{p}}_l, \quad (43)$$

where  $\boldsymbol{\Pi}^m$  is a matrix form of the projector  $\boldsymbol{\Pi}_{l, E}^0 \nabla$  and  $\mathbf{N}^p$  is a matrix of complete polynomial of order  $l$

$$(\mathbf{N}^p)^T := \begin{bmatrix} \mathbf{m}_l^T & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{m}_l^T & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{m}_l^T \end{bmatrix}. \quad (44)$$

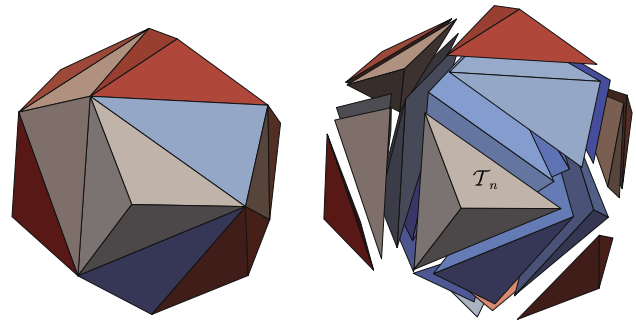


Fig. 3 Polyhedral triangulation and numerical integration

Substituting Eq. (43) into Eq. (41), considering Eq. (42) and  $u_h = \boldsymbol{\phi}^T \tilde{\mathbf{u}}$ , yields

$$\begin{aligned} \tilde{\mathbf{p}}^T \int_E \mathbf{N}^p (\mathbf{N}^p)^T d\Omega \boldsymbol{\Pi}^m \tilde{\mathbf{u}} \\ = \tilde{\mathbf{p}}^T \int_{\partial E} (\mathbf{N}^p \cdot \mathbf{n}_F) \boldsymbol{\phi}^T d\Gamma \tilde{\mathbf{u}} - \tilde{\mathbf{p}}^T \int_E (\operatorname{div} \mathbf{N}^p) \boldsymbol{\phi}^T d\Omega \tilde{\mathbf{u}}, \end{aligned} \quad (45)$$

which can be rewritten as

$$\begin{aligned} \int_E \mathbf{N}^p (\mathbf{N}^p)^T d\Omega \boldsymbol{\Pi}^m \\ = \int_{\partial E} (\mathbf{N}^p \cdot \mathbf{n}_F) \boldsymbol{\phi}^T d\Gamma - \int_E (\operatorname{div} \mathbf{N}^p) \boldsymbol{\phi}^T d\Omega, \end{aligned} \quad (46)$$

since the form is true for all  $\tilde{\mathbf{u}}$  and  $\tilde{\mathbf{p}}$ . Similar to Eq. (35), Eq. (46) leads to the matrix form

$$\mathbf{G} \boldsymbol{\Pi}^m = \mathbf{B}, \quad (47)$$

where

$$\mathbf{G} := \int_E \mathbf{N}^p (\mathbf{N}^p)^T d\Omega, \quad (48)$$

$$\mathbf{B} := \int_{\partial E} (\mathbf{N}^p \cdot \mathbf{n}_F) \boldsymbol{\phi}^T d\Gamma - \int_E (\operatorname{div} \mathbf{N}^p) \boldsymbol{\phi}^T d\Omega. \quad (49)$$

The matrices  $\mathbf{B}$  and  $\mathbf{G}$  are very similar to the matrices used in the  $\mathcal{H}_1$  projection matrix, see Eqs. (37) and (38). Between the two matrices mentioned above, matrix  $\mathbf{G}$  is more straightforward to compute. It involves dividing the polyhedral element into multiple tetrahedral elements (as shown in Fig. 3) and utilizing Gaussian integration at each tetrahedron for the calculation. Since matrix  $\mathbf{G}$  is the integration of a complete polynomial, the application of the divergence theorem facilitates the conversion of the integral to the surface  $F$  and subsequently to the edge  $e$ . A detailed exposition of this process can be found in the VEM book [25].

The calculation of matrix  $\mathbf{B}$  seems to be very complicated because  $\mathbf{B}$  contains higher-order moments (see Eq. (49)) that

have not been defined before (another understanding is, that we do not know the basis function  $\phi$  within the element  $E$  and on the element boundary  $F$ ). In order not to increase the degree of freedom, we can change the first-order polynomial  $\mathbb{P}_1$  in the boundary element space (see Eqs. (20) and (24)) to an  $l$ -order polynomial  $\mathbb{P}_l$  and can take advantage of the properties related to the enlarged enhancement virtual element space (low-order mapping operators approximate high-order moments) [41]. Then the matrix  $\mathbf{B}$  can be written as

$$\begin{aligned} \mathbf{B} &= \sum_{F \subset \partial E} \int_e N^p \mathbf{n}_F \Pi_{1,f}^\nabla \phi^T d\Gamma - \int_E (\text{div} N^p) \phi^T \Pi_{1,E}^\nabla d\Omega \\ &= \sum_{F \subset \partial E} \int_e N^p \mathbf{n}_F (\mathbf{m}_1^f)^T d\Gamma \Pi_{1*,f}^\nabla \\ &\quad - \int_E (\text{div} N^p) \mathbf{m}_1^T d\Omega \Pi_{1*,E}^\nabla. \end{aligned} \tag{50}$$

The integrations in Eq. (50) can be calculated by partitioning  $F$  and  $E$  into triangles and tetrahedrons and adopting a Gauss quadrature rule.

Substituting matrices  $\mathbf{G}$  and  $\mathbf{B}$  into Eq. (47) yields the projection matrix  $\mathbf{\Pi}^m$

$$\mathbf{\Pi}^m = \mathbf{G}^{-1} \mathbf{B}, \tag{51}$$

and the gradient of the variable can be approximated by

$$\nabla u \approx \mathbf{\Pi}_{l,E}^0 \nabla u_h = (\mathbf{N}^p)^T \mathbf{\Pi}^m \tilde{\mathbf{u}}. \tag{52}$$

### 5 Stabilization-free virtual element method for 3D hyperelastic problems

According to the discussion in the previous section, we can directly use Eq. (52) to approximate the gradient of the variable. Under the premise of choosing an appropriate polynomial order  $l$ , the virtual element method format does not require additional stabilization terms. Therefore, when addressing nonlinear problems such as hyperelasticity, we can employ the same framework (calculation process) as in the finite element method. Subsequently, we will follow the basic ideas in continuum mechanics to construct a format for SFVEM to solve this nonlinear problem.

#### 5.1 Lagrange linearized internal virtual work

Recall from Eq. (10) that the internal virtual work can be expressed in a Lagrange form as

$$\mathcal{W} = \int_{V_0} \delta \mathbf{E} : \mathbf{S} dV_0 \tag{53}$$

Then, the linearization of the virtual work is obtained from

$$\delta \mathcal{W} = \int_{V_0} \delta \mathbf{E} : \Delta \mathbf{S} dV_0 + \int_{V_0} \Delta (\delta \mathbf{E}) : \mathbf{S} dV_0. \tag{54}$$

Considering the definition of the Green-Lagrange strain tensor  $\mathbf{E}$  (Eq. (4)), we have

$$\delta \mathbf{E} = \frac{1}{2} (\delta \mathbf{F}^T \cdot \mathbf{F} + \mathbf{F}^T \cdot \delta \mathbf{F}) \tag{55}$$

and

$$\begin{aligned} \Delta \delta \mathbf{E} &= \frac{1}{2} (\delta \mathbf{F}^T \cdot \Delta \mathbf{F} + \Delta \mathbf{F}^T \cdot \delta \mathbf{F}) \\ &= \frac{1}{2} (\delta \mathbf{D}^T \Delta \mathbf{D} + \Delta \mathbf{D}^T \delta \mathbf{D}) \end{aligned} \tag{56}$$

where

$$\mathbf{D} = \frac{\partial \mathbf{u}}{\partial \mathbf{X}}, \quad \mathbf{F} = \mathbf{1} + \mathbf{D}. \tag{57}$$

Substituting Eqs. (56) and (57) into Eq. (54) and considering  $\Delta \mathbf{S} = \mathcal{D} : \Delta \mathbf{E}$ , the linearization of the virtual work has the formulation as

$$\begin{aligned} \delta \mathcal{W} &= \int_{V_0} \delta \mathbf{E} : \Delta \mathbf{S} dV_0 + \int_{V_0} \mathbf{S} : (\delta \mathbf{D}^T \cdot \Delta \mathbf{D}) dV_0 \\ &= \int_{V_0} \delta \mathbf{E} : \mathcal{D} : \Delta \mathbf{E} dV_0 + \int_{V_0} \mathbf{S} : (\delta \mathbf{D}^T \cdot \Delta \mathbf{D}) dV_0, \end{aligned} \tag{58}$$

where  $\mathcal{D}$  is the constitutive tensor given in Eq. (14).

#### 5.2 Virtual element discretization

For three-dimensional problems, utilizing Voigt notation, the Green-Lagrange strain tensor can be written as

$$\hat{\mathbf{E}} = [E_{XX} \ E_{YY} \ E_{ZZ} \ 2E_{XY} \ 2E_{YZ} \ 2E_{ZX}]^T, \tag{59}$$

and the second Piola–Kirchhoff stress can be written as

$$\hat{\mathbf{S}} = [S_{XX} \ S_{YY} \ S_{ZZ} \ S_{XY} \ S_{YZ} \ S_{ZX}]^T. \tag{60}$$

Besides, the incremental constitutive relationship can be rewritten as

$$\Delta \hat{\mathbf{S}} = \hat{\mathcal{D}} \cdot \Delta \hat{\mathbf{E}}. \tag{61}$$

Considering Eq. (52), the gradient of displacement  $\mathbf{D}$  (defined in Eq. (57)) can be simulated as

$$\mathbf{D} = \begin{bmatrix} \frac{\partial u}{\partial X} & \frac{\partial u}{\partial Y} & \frac{\partial u}{\partial Z} \\ \frac{\partial v}{\partial X} & \frac{\partial v}{\partial Y} & \frac{\partial v}{\partial Z} \\ \frac{\partial w}{\partial X} & \frac{\partial w}{\partial Y} & \frac{\partial w}{\partial Z} \end{bmatrix} = [(\mathbf{N}^p)^T \mathbf{\Pi}^m [\tilde{\mathbf{u}} \ \tilde{\mathbf{v}} \ \tilde{\mathbf{w}}]]^T, \tag{62}$$

and the Green-Lagrange strain tensor can be calculated as

$$\begin{aligned}\Delta \hat{\mathbf{E}} &= \mathbf{A}_1 (\mathbf{N}^p)^T \boldsymbol{\Pi}^m \Delta \tilde{\mathbf{u}} + \mathbf{A}_2 (\mathbf{N}^p)^T \boldsymbol{\Pi}^m \Delta \tilde{\mathbf{v}} \\ &\quad + \mathbf{A}_3 (\mathbf{N}^p)^T \boldsymbol{\Pi}^m \Delta \tilde{\mathbf{w}} \\ &= \left[ \mathbf{A}_1 (\mathbf{N}^p)^T \boldsymbol{\Pi}^m \quad \mathbf{A}_2 (\mathbf{N}^p)^T \boldsymbol{\Pi}^m \quad \mathbf{A}_3 (\mathbf{N}^p)^T \boldsymbol{\Pi}^m \right] \Delta \tilde{\mathbf{U}} \\ &= \mathbf{A} \mathbf{N}_p^T \boldsymbol{\Pi}_m \Delta \tilde{\mathbf{U}},\end{aligned}\quad (63)$$

where

$$\mathbf{A}_1 = \begin{bmatrix} F_{11} & 0 & 0 \\ 0 & F_{12} & 0 \\ 0 & 0 & F_{13} \\ F_{12} & F_{11} & 0 \\ 0 & F_{13} & F_{12} \\ F_{13} & 0 & F_{11} \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} F_{21} & 0 & 0 \\ 0 & F_{22} & 0 \\ 0 & 0 & F_{23} \\ F_{22} & F_{21} & 0 \\ 0 & F_{23} & F_{22} \\ F_{23} & 0 & F_{21} \end{bmatrix}$$

$$\mathbf{A}_3 = \begin{bmatrix} F_{31} & 0 & 0 \\ 0 & F_{32} & 0 \\ 0 & 0 & F_{33} \\ F_{32} & F_{31} & 0 \\ 0 & F_{33} & F_{32} \\ F_{33} & 0 & F_{31} \end{bmatrix}, \quad (64)$$

$$\mathbf{A} = [\mathbf{A}_1 \quad \mathbf{A}_2 \quad \mathbf{A}_3], \quad (65)$$

$$\mathbf{N}_p^T = \begin{bmatrix} (\mathbf{N}^p)^T & & \\ & (\mathbf{N}^p)^T & \\ & & (\mathbf{N}^p)^T \end{bmatrix}, \quad \boldsymbol{\Pi}_m = \begin{bmatrix} \boldsymbol{\Pi}^m & & \\ & \boldsymbol{\Pi}^m & \\ & & \boldsymbol{\Pi}^m \end{bmatrix}, \quad (66)$$

and

$$\Delta \tilde{\mathbf{U}} = [\Delta \tilde{\mathbf{u}} \quad \Delta \tilde{\mathbf{v}} \quad \Delta \tilde{\mathbf{w}}]^T. \quad (67)$$

For the last term in Eq. (58), we can define vector  $\boldsymbol{\theta}$  as

$$\boldsymbol{\theta} = \left[ \frac{\partial u}{\partial X} \quad \frac{\partial u}{\partial Y} \quad \frac{\partial u}{\partial Z} \quad \frac{\partial v}{\partial X} \quad \frac{\partial v}{\partial Y} \quad \frac{\partial v}{\partial Z} \quad \frac{\partial w}{\partial X} \quad \frac{\partial w}{\partial Y} \quad \frac{\partial w}{\partial Z} \right]^T, \quad (68)$$

and the term  $\delta \boldsymbol{\theta}$  can be calculated as

$$\Delta \boldsymbol{\theta} = \mathbf{N}_p^T \cdot \boldsymbol{\Pi}_m \cdot \Delta \tilde{\mathbf{U}}. \quad (69)$$

### 5.3 Element stiffness matrix and internal force

Based on the definition of  $\boldsymbol{\theta}$  given in Eq. (68), the last term in Eq. (58) can be written as

$$\begin{aligned}\mathbf{S} : (\delta \mathbf{D}^T \cdot \Delta \mathbf{D}) &= \delta \boldsymbol{\theta}^T \cdot \boldsymbol{\mathcal{I}} \cdot \Delta \boldsymbol{\theta} \\ &= \delta \tilde{\mathbf{U}}^T \cdot \left( \boldsymbol{\Pi}_m^T \cdot \mathbf{N}_p \right) \cdot \boldsymbol{\mathcal{I}} \\ &\quad \cdot \left( \mathbf{N}_p^T \cdot \boldsymbol{\Pi}_m \right) \cdot \Delta \tilde{\mathbf{U}},\end{aligned}\quad (70)$$

where

$$\boldsymbol{\mathcal{I}} = \begin{bmatrix} \mathbf{S} & & \\ & \mathbf{S} & \\ & & \mathbf{S} \end{bmatrix}. \quad (71)$$

Substituting Eq. (70) into Eq. (58), the linearized variation of energy can be written as

$$\delta \mathcal{W} = \int_{V_0} \delta \hat{\mathbf{E}}^T \cdot \hat{\mathbf{D}} \cdot \Delta \hat{\mathbf{E}} dV_0 + \int_{V_0} \delta \boldsymbol{\theta}^T \cdot \boldsymbol{\mathcal{I}} \cdot \Delta \boldsymbol{\theta} dV_0. \quad (72)$$

Consider Eqs. (63) and (69), we have

$$\begin{aligned}\delta \mathcal{W} &= \delta \tilde{\mathbf{U}}^T \cdot \int_{V_0} \left( \boldsymbol{\Pi}_m^T \cdot \mathbf{N}_p \cdot \mathbf{A}^T \right) \\ &\quad \cdot \hat{\mathbf{D}} \cdot \left( \mathbf{A} \cdot \mathbf{N}_p^T \cdot \boldsymbol{\Pi}_m \right) dV_0 \cdot \Delta \tilde{\mathbf{U}} \\ &\quad + \delta \tilde{\mathbf{U}}^T \cdot \int_{V_0} \left( \boldsymbol{\Pi}_m^T \cdot \mathbf{N}_p \right) \cdot \boldsymbol{\mathcal{I}} \cdot \left( \mathbf{N}_p^T \cdot \boldsymbol{\Pi}_m \right) dV_0 \cdot \Delta \tilde{\mathbf{U}}.\end{aligned}\quad (73)$$

Then the element tangent stiffness matrix can be obtained by

$$\mathbf{K}_{K2}^t = \mathbf{K}_{K2}^{t1} + \mathbf{K}_{K2}^{t\sigma}, \quad (74)$$

where

$$\mathbf{K}_{K2}^{t1} = \boldsymbol{\Pi}_m^T \cdot \int_{V_0} \mathbf{N}_p \cdot \mathbf{A}^T \cdot \hat{\mathbf{D}} \cdot \mathbf{A} \cdot \mathbf{N}_p^T dV_0 \cdot \boldsymbol{\Pi}_m, \quad (75)$$

$$\mathbf{K}_{K2}^{t\sigma} = \boldsymbol{\Pi}_m^T \cdot \int_{V_0} \mathbf{N}_p \cdot \boldsymbol{\mathcal{I}} \cdot \mathbf{N}_p^T dV_0 \cdot \boldsymbol{\Pi}_m. \quad (76)$$

The  $\boldsymbol{\Pi}_m$  matrix is calculated and stored before the iteration. In addition, in each nonlinear iteration step, the integral term in the tangent stiffness matrix needs to be calculated based on the current stress state. Similar to before, the integral can be calculated by dividing the polyhedron into tetrahedrons and adopting a Gauss quadrature rule. The internal force is obtained from Eq. (10) as

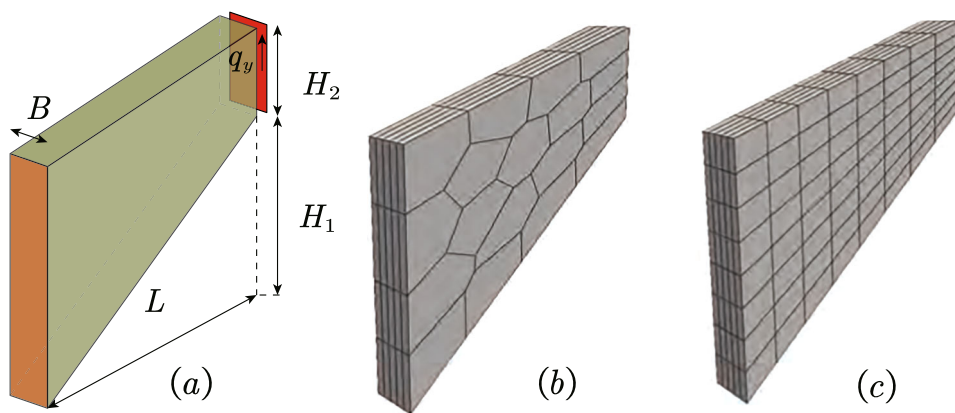
$$\mathbf{F}_{\text{int}} = \boldsymbol{\Pi}_m^T \cdot \int_{V_0} \mathbf{N}_p \cdot \mathbf{A}^T \cdot \hat{\mathbf{S}} dV_0. \quad (77)$$

At each step, the second Piola–Kirchhoff stress  $\hat{\mathbf{S}}$  at each interpolation point is calculated and the Cauchy stress  $\boldsymbol{\sigma}$  can be calculated as

$$\boldsymbol{\sigma} = \frac{1}{J} \mathbf{F} \cdot \mathbf{S} \cdot \mathbf{F}^T. \quad (78)$$



**Fig. 4** 3D Cook’s membrane problem, **a** Geometry and boundary condition; **b** example of a polyhedral swept mesh; **c** example of a regular mesh



## 6 Numerical examples

### 6.1 3D Cook’s membrane problem

As a first example, we consider the standard Cook’s membrane bending test with the geometric model illustrated in Fig. 4. The relevant dimensions are  $L = 48$ ,  $H_1 = 44$ ,  $H_2 = 16$ ,  $B = 10$ . The left side of the tapered beam is fixed and the right side is applied a constant distributed vertical load  $q_y$ . The material parameters are selected as  $\lambda = 100$  and  $\mu = 40$ . The distributed vertical load is given as  $q_y = 4$ .

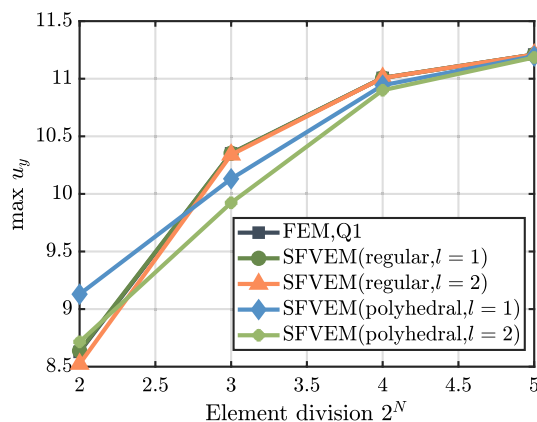
For SFVEM, it is necessary to ensure that the order of the basis function of the strain mode meets the requirements (the rank of the resulting stiffness matrix is correct). Using an eigenvalue analysis of different polyhedral elements, see [45], a relationship between the number of vertices and the polynomial order  $l$  was obtained. Furthermore, we note that while an individual element is rank deficient, the rank of the global stiffness matrix can be correct. In such cases, a low-order polynomial ( $l = 1$ ) can be used to avoid the high computational cost caused by the domain integration. The results obtained from polynomials of different orders will be compared in this example.

To compare the performance of SFVEM for 3D hyperelastic problems, the first-order finite element method (Q1) is selected. The convergence is studied here by adopting various meshes using uniform refinement. Four different meshes defined by the parameter  $N$  which corresponds to the number of divisions  $2^N \times 2^N$  in  $L \times H$  are selected. As shown in Fig. 4, two different types of meshes including a polyhedral swept mesh and a hexahedral swept mesh are used in this example. For different meshes, the maximum vertical displacement  $u_y$  for different divisions  $2^N$  are solved by different methods and compared in Table 1.

Table 1 illustrates that the results calculated by SFVEM for regular meshes are very close to those obtained by FEM. The results depict the same convergence characteristic as SFVEM. Polyhedral meshes do not change this behavior. The order of the polynomial  $l = 1$  yields already accurate

**Table 1** Maximum vertical displacement  $u_y$  for different element division  $N$  for polyhedral and regular meshes

$N$	FEM(regular)	SFVEM(regular)		SFVEM(polyhedral)	
		$l = 1$	$l = 2$	$l = 1$	$l = 2$
	Q1				
2	8.6285	8.6378	8.5233	9.1293	8.7152
3	10.3516	10.3530	10.3398	10.1310	9.9228
4	11.0071	11.0073	11.0064	10.9434	10.9002
5	11.2099	11.2099	11.2098	11.1964	11.1846



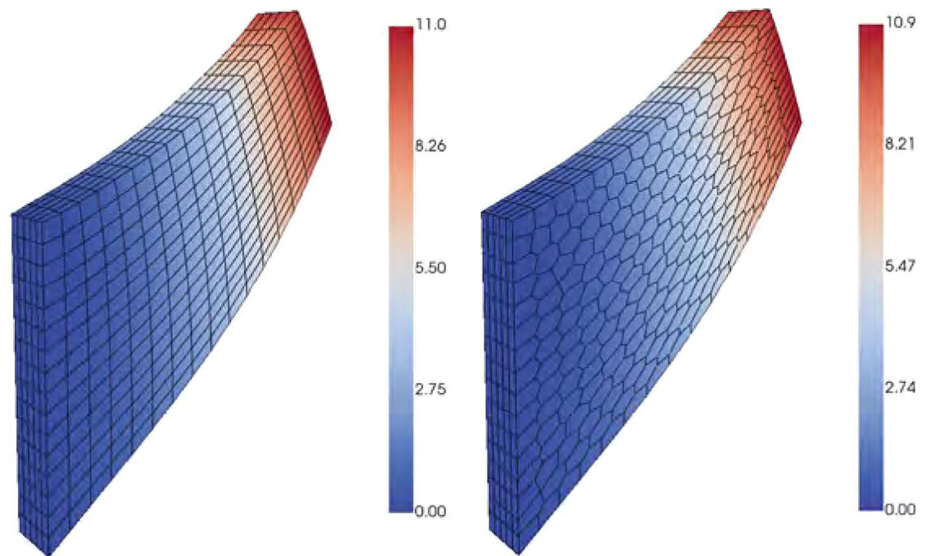
**Fig. 5** Maximum value of vertical displacement  $u_y$  for different element division  $2^N$

results. Higher-order polynomials are less efficient for the same accuracy. The convergence curves obtained by different methods and different parameters are shown in Fig. 5. Besides, for  $l = 1$  and  $N = 4$ , the contour plots of displacements  $u_y$  and stresses  $S_{yx}$  are given in Figs. 6 and 7.

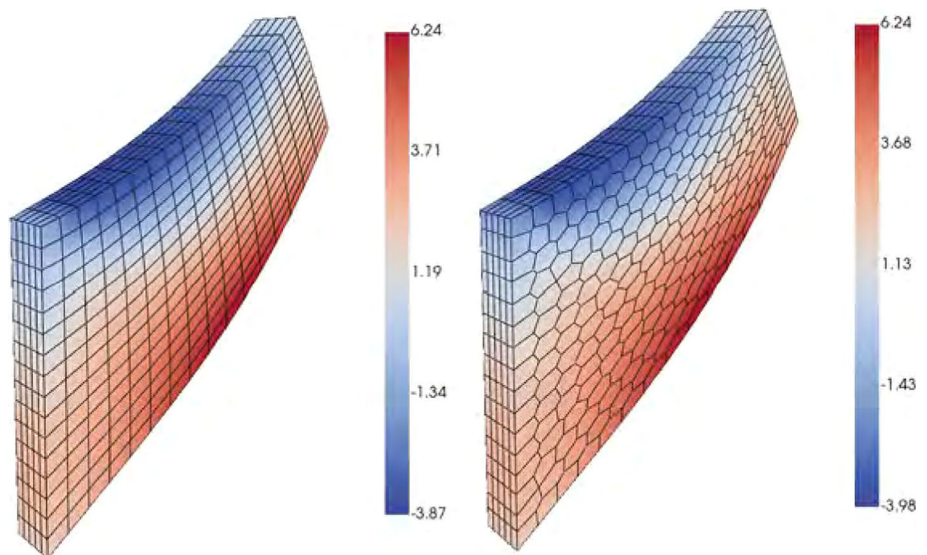
### 6.2 Punch problem

In this example, a punch problem is selected which is subjected to high compression loading. The geometry and relevant dimensions are given in Fig. 8 as  $L_1 = L_2 = 2$ ,  $H = 1$ . A surface load  $P = 300$  is applied on one-quarter of

**Fig. 6** Contour plots of displacement  $u_y$  obtained by SFVEM for different meshes,  $l = 1, N = 4$



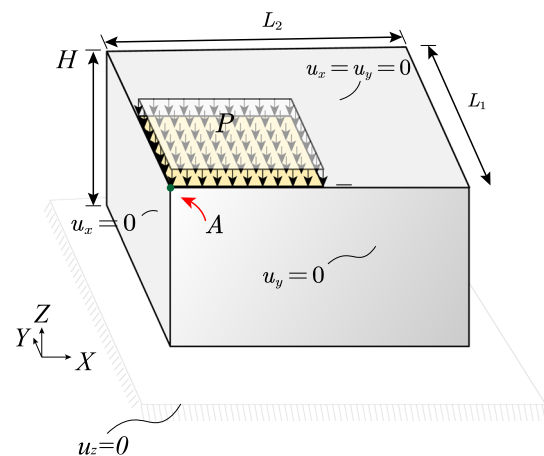
**Fig. 7** Contour plots of stresses  $S_{xy}$  obtained by SFVEM for different meshes,  $l = 1, N = 4$



the block and other boundary conditions are given in Fig. 8. The Neo-Hookean material model is used with the Lamé constants  $\lambda = 400.75$  and  $\mu = 92.5$ .

Similar to the first example, a numerical convergence study is presented for different meshes. The vertical displacement  $u_z$  of point  $A$  (as shown in Fig. 8) is reported for mesh refinement parameter  $N$  corresponding to a mesh of  $N \cdot 2N \cdot 2N$  for regular and polyhedral meshes. For the regular meshes, the order of the strain mode is selected as  $l = 1$ . For the polyhedral meshes, the order of the strain mode is selected as  $l = 1$  and  $l = 2$ . The first-order finite element method (FEM, Q1) is selected for comparison.

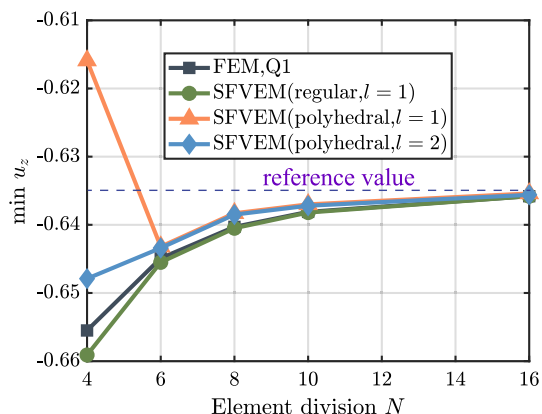
The values of the vertical displacements  $u_z$  of point  $A$  are calculated and given in Table 2 for regular meshes and polyhedral meshes. For the regular hexahedral mesh, the results obtained by SFVEM are very close to those obtained by FEM, even for coarse meshes. For polyhedral meshes, for



**Fig. 8** Punch problem and boundary boundary conditions

**Table 2** Maximum vertical displacement  $u_z$  for different element division  $N$  for polyhedral and regular meshes

$N$	FEM	SFVEM(regular)	SFVEM(poly, $l = 1$ )	SFVEM(poly, $l = 2$ )
4	-0.6555	-0.6591	-0.6159	-0.6479
6	-0.6449	-0.6455	-0.6432	-0.6434
8	-0.6403	-0.6405	-0.6383	-0.6385
10	-0.6381	-0.6382	-0.6370	-0.6372
16	-0.6358	-0.6358	-0.6354	-0.6356



**Fig. 9** Minimum value (corner A) of vertical displacement  $u_z$  for different element division  $N$

strain modes of different orders  $l$ , the displacement results obtained are different only in the case of the coarsest mesh ( $N = 4$ ). The reason for this difference may be that the mesh

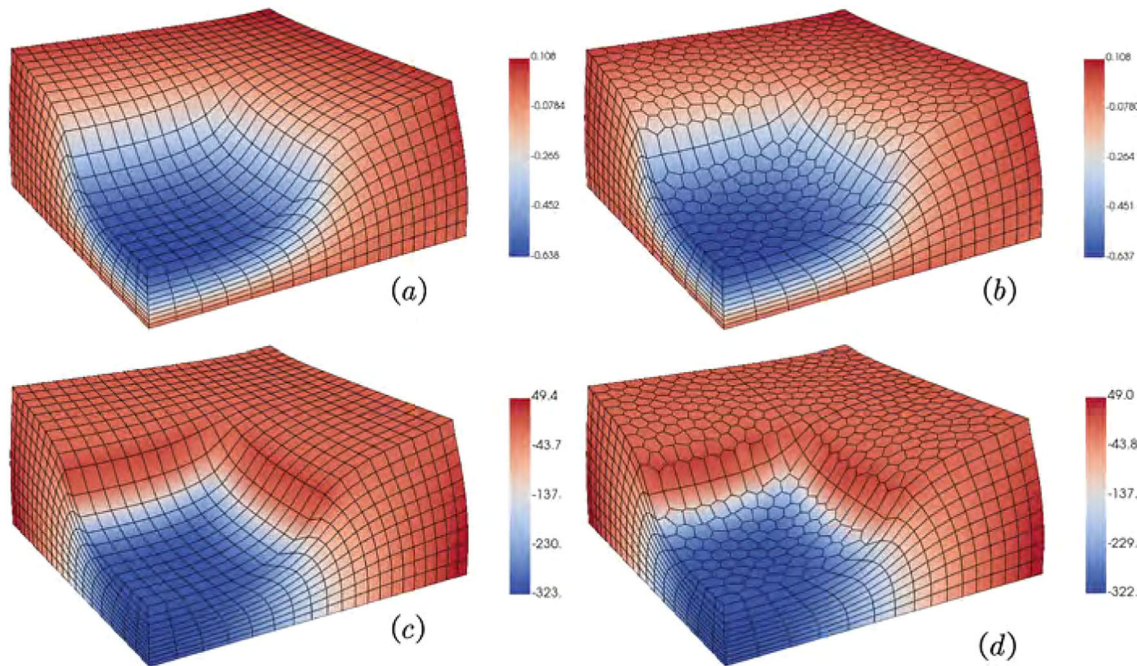
size is too large. For other meshes, the displacement results obtained by different orders  $l$  are basically the same.

The convergence of the displacement  $u_z$  of point A is demonstrated in Fig. 9. For the current material and load boundary conditions, we used a quadratic (Q2) element to calculate a reference solution, which led to  $u_z^{\text{ref}} = -0.6344$  and is illustrated in Fig. 9. It can be seen from the figure that although the results obtained by the hexahedral mesh are in better agreement with the finite element, the results obtained by the polyhedral mesh are closer to the reference solution.

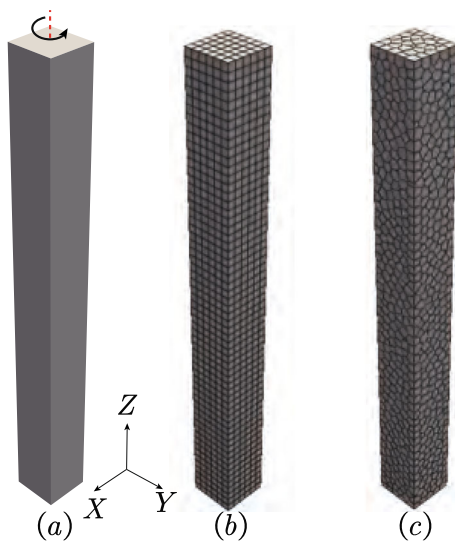
Finally, for  $N = 10$ , the contour plots of displacement  $u_z$  and stress  $\sigma_{zz}$  obtained by SFVEM on hexahedral mesh ( $l = 1$ ) and polyhedral mesh ( $l = 2$ ) are depicted in Fig. 10.

### 6.3 Twisting of a column

In this example, the twisting of a column is used to evaluate the performance of the proposed method under extremely large deformations. The geometry of the problem and the virtual element meshes (including hexahedral meshes and



**Fig. 10** Contour plots of displacements and stresses obtained by SFVEM with different meshes, **a, b** displacements  $u_z$ ; **c, d** stresses  $\sigma_{zz}$



**Fig. 11** Twisting column: geometry, hexahedral mesh and polyhedral mesh used for the analysis

polyhedral meshes) are given in Fig. 11. The bottom nodes are fixed and the top nodes rotate around the  $Z$ -axis as:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \quad (79)$$

where  $(X, Y, Z)$  represents the initial configuration,  $(x, y, z)$  is the deformed configuration. Besides,  $\theta$  represents the angle of rotation for the top surface around the  $Z$ -axis.

The material model for this example is assumed as Neo-Hookean hyperelastic with the Lamé parameters  $\lambda = 100$  and  $\mu = 40$ . The angle of rotation of the top surface is given

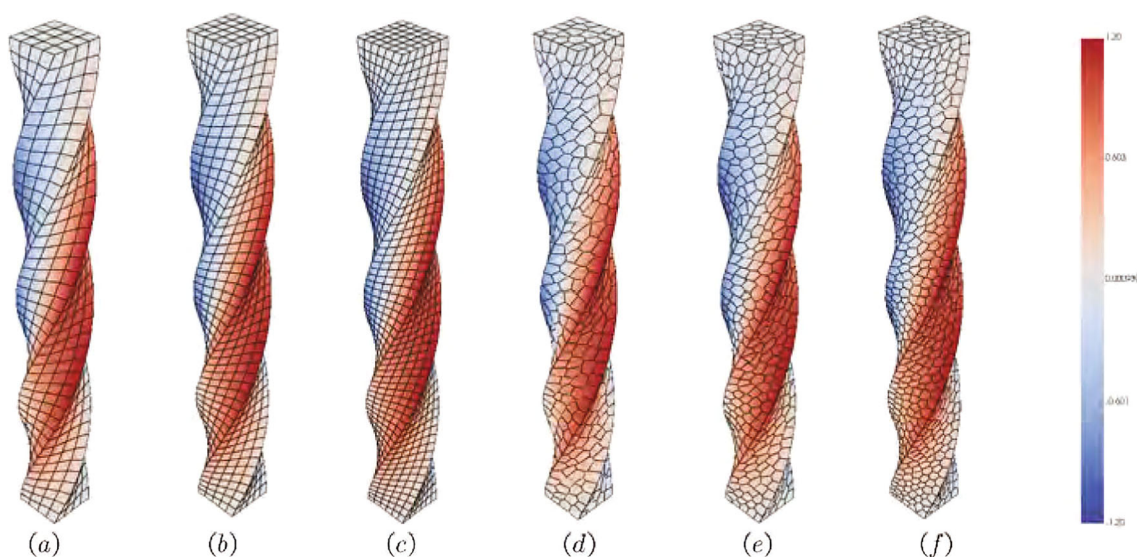
as  $\theta = \pi$ . The order of the polynomial is selected as  $l = 1$  to save computing time. For different meshes, the deformed shapes and contour plots of displacement  $u_y$  are provided in Fig. 12. Again, the regular and polyhedral meshes lead to the same responses.

## 6.4 Force deformation of round arch

The last example is an arch subjected to a concentrated load. The geometric model and polyhedral mesh are given in Fig. 13. The outer radius of the arch is 100, the inner radius is 95, and the thickness is 10. As shown in Fig. 13, two different boundary conditions are applied. For boundary condition 1, the left side of the model is fixed. For boundary condition 2, the left side of the model is hinged. For two different boundary conditions, the bottom right end of the model is fixed. Nodal forces are applied to the top of the model. The material parameters are assumed as  $E = 5$  and  $\nu = 0.35$ .

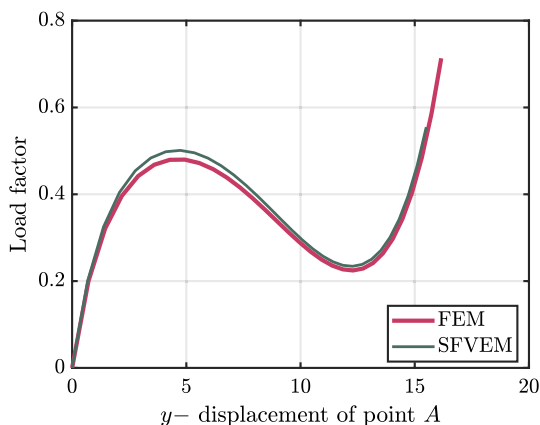
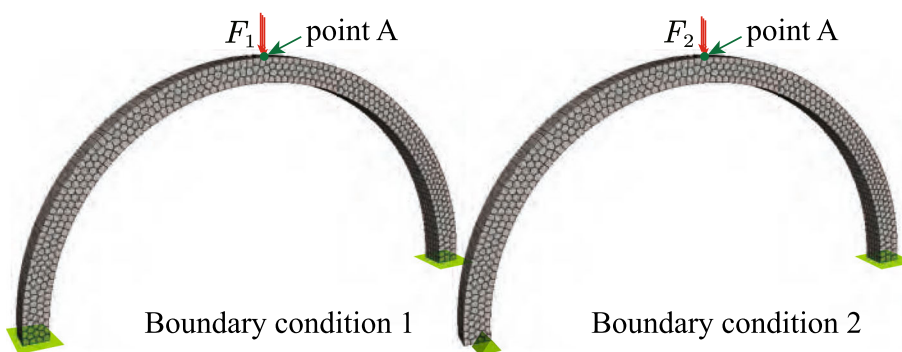
This type of structure is geometrically unstable and needs the path-following or arc-length solution method. The basic idea of the arc-length method is to add a constraint condition to the set of nonlinear equations, which is associated with the total load factor  $\lambda$ . Since the method proposed in this work does not require stabilization, the techniques employed in the nonlinear finite element method can be directly used. For details, see related books on nonlinear finite elements method [46].

For the first type of boundary condition, the initial load factor  $\lambda$  is selected as  $\lambda = 0.2$ . For the second type of boundary condition, the initial load factor  $\lambda$  is selected as  $\lambda = 0.02$ . The evolution of displacement  $u_y$  of point A (as shown in Fig. 13) with respect to the load factor is plotted in Fig. 14 and Fig. 15 for different boundary conditions, respectively.

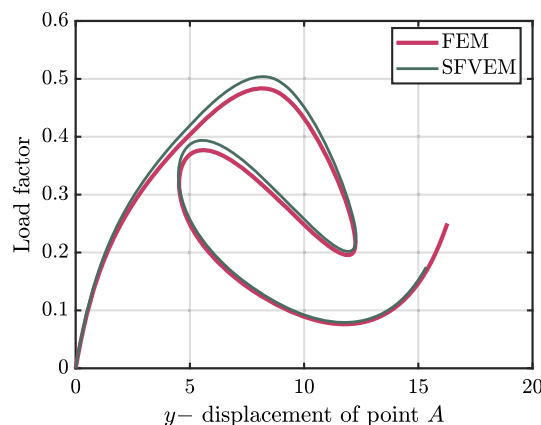


**Fig. 12** Contour plots of displacements  $u_y$  obtained by SFVEM with different meshes, **a–c** hexahedral mesh; **d–f** polyhedral mesh

**Fig. 13** Three-dimensional arch geometry and mesh. This problem considers two different boundary conditions: boundary condition 1: the left side of the model is fixed, boundary condition 2: the left side of the model is hinged



**Fig. 14** Load displacement curve of the round arch with boundary 1



**Fig. 15** Load displacement curve of the round arch with boundary 2

The load-deflection curves obtained with the SFVEM match very well with the solutions obtained by FEM with similar meshes. For the second type of boundary condition, Fig. 15 shows the complex nonlinear behavior (snap-back) of this simple structure. The contour plots obtained under different displacements and different boundary conditions are provided in Fig. 16.

### 7 Conclusion

This paper introduces a stabilization-free virtual element method for 3D hyperelastic problems. As an extension to the two-dimensional problem, a first-order three-dimensional virtual element method without any stabilization is presented. This paper discusses in detail the specific calculation process of the traditional first-order  $\mathcal{H}_1$  projection operator and

the high-order  $L_2$  projection operator for three-dimensional polyhedral elements. The  $L_2$  projection operators are computed relying on the components of the gradient, and the matrix for strain approximation is directly constructed. As no stability terms are required in SFVEM, hyperelastic problems can be directly solved within the framework following the finite element method. Benchmark examples are used to demonstrate the accuracy of the 3D SFVEM for hyperelastic problems. In addition, this paper also uses the arc-length method to track the complete structural deformation of unstable structures. It can be seen from the numerical examples that when using a hexahedral mesh, the results obtained are similar to FEM. At the same time, SFVEM yields more accurate results for polyhedral meshes. Since no stabilization term is required, the method can be applied to complex multiphysics problems, such as large-strain incompressible electromechanics or electromechanical growth problems in the future.

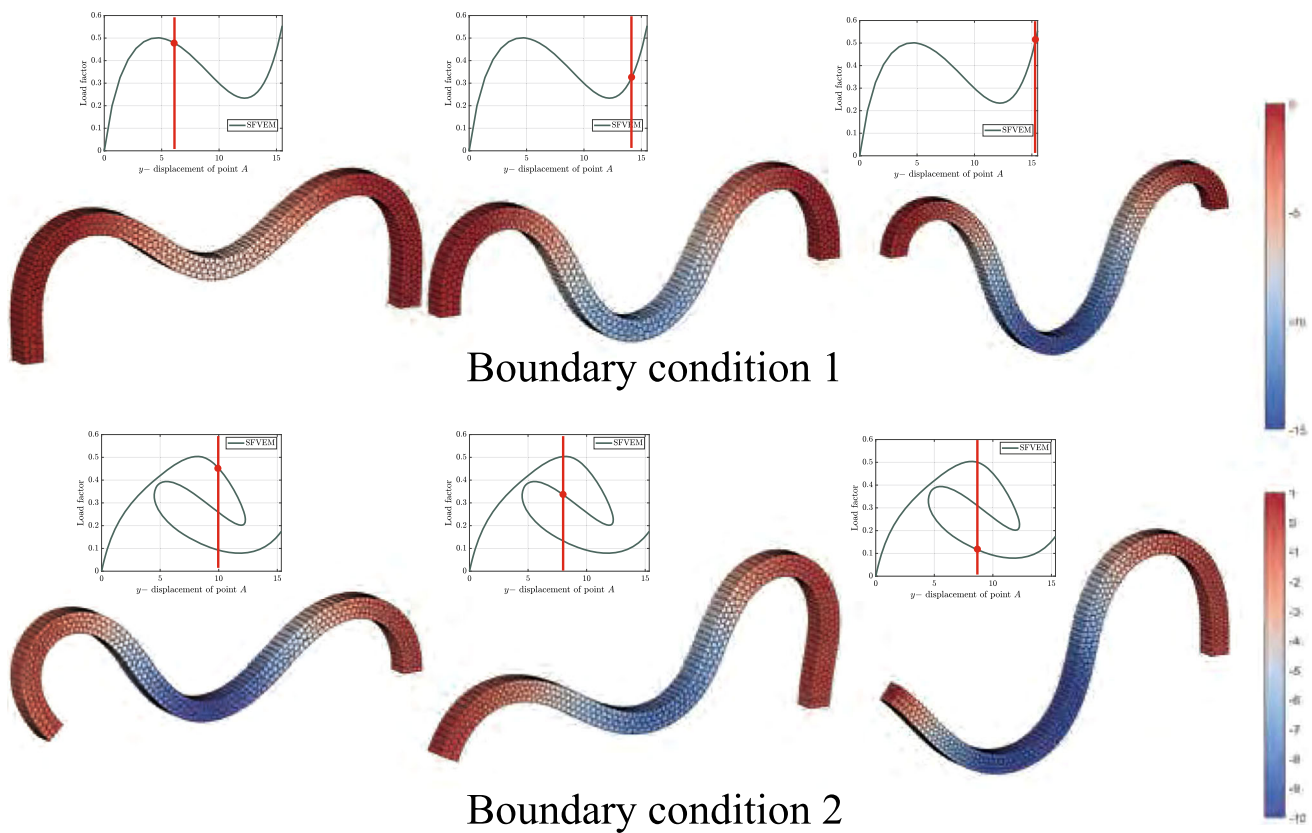


Fig. 16 Contour plots of displacement  $u_y$  at different times

**Acknowledgements** The first and last authors are grateful for the support provided by the Alexander von Humboldt Foundation, Germany.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Veiga L et al (2012) Basic principles of virtual element methods. *Math Models Methods Appl Sci* 23:199–214
2. Veiga L, Brezzi F, Marini L, Russo A (2014) Mixed Virtual Element Methods for general second order elliptic problems on polygonal meshes. *ESAIM Math Model Numer Anal* 26:727–747
3. Veiga L, Brezzi F (2013) Virtual elements for linear elasticity problems. *SIAM J Numer Anal* 51:794–812
4. Gain A, Talischi C, Paulino G (2013) On the virtual element method for three-dimensional elasticity problems on arbitrary polyhedral meshes. *Comput Methods Appl Mech Eng* 282:132–160
5. Artioli E, Veiga L, Lovadina C, Sacco E (2017) Arbitrary order 2D virtual elements for polygonal meshes: part I, elastic problem. *Comput Mech* 60:727–747
6. Dassi F, Lovadina C, Visinoni M (2020) A three-dimensional Hellinger–Reissner virtual element method for linear elasticity problems. *Comput Methods Appl Mech Eng* 364:112910
7. Mengolini M, Benedetto M, Aragón A (2019) An engineering perspective to the virtual element method and its interplay with the standard finite element method. *Comput Methods Appl Mech Eng* 350:995–1023
8. Chi H, Veiga L, Paulino G (2016) Some basic formulations of the virtual element method (VEM) for finite deformations. *Comput Methods Appl Mech Eng* 318:995–1023
9. Wriggers P, Reddy B, Rust W, Hudobivnik B (2017) Efficient virtual element formulations for compressible and incompressible finite deformations. *Comput Mech* 60:995–1023
10. van Huyssteen D, Reddy B (2020) A virtual element method for isotropic hyperelasticity. *Comput Methods Appl Mech Eng* 367:113134
11. de Bellis ML, Wriggers P, Hudobivnik B (2019) Serendipity virtual element formulation for nonlinear elasticity. *Comput Struct* 223:106094
12. Wriggers P, Rust W, Reddy B (2016) A virtual element method for contact. *Comput Mech* 58:995–1023

13. Aldakheel F, Hudobivnik B, Artioli E, Veiga L, Wriggers P (2020) Curvilinear virtual elements for contact mechanics. *Comput Methods Appl Mech Eng* 372:113394
14. Shen W, Ohsaki M, Zhang J (2022) A 2-dimensional contact analysis using second-order virtual element method. *Comput Mech* 70:995–1023
15. Cihan M, Hudobivnik B, Korelc J, Wriggers P (2022) A virtual element method for 3D contact problems with non-conforming meshes. *Comput Methods Appl Mech Eng* 402:115385
16. Liu T-R, Aldakheel F, Aliabadi M (2023) Virtual element method for phase field modeling of dynamic fracture. *Comput Methods Appl Mech Eng* 411:116050
17. Aldakheel F, Hudobivnik B, Wriggers P (2019) Virtual element formulation for phase-field modeling of ductile fracture. *Int J Multiscale Comput Eng* 17:181–200
18. Park K, Chi H, Paulino G (2019) On nonconvex meshes for elastodynamics using virtual element methods with explicit time integration. *Comput Methods Appl Mech Eng* 356:669–684
19. Park K, Chi H, Paulino G (2019) Numerical recipes for elastodynamic virtual element methods with explicit time integration. *Int J Numer Methods Eng* 121:1–31
20. Cihan M, Hudobivnik B, Aldakheel F, Wriggers P (2021) Virtual element formulation for finite strain elastodynamics. *Comput Model Eng Sci* 129:1151–1180
21. Sukumar N, Tupek M (2022) Virtual elements on agglomerated finite elements to increase the critical time step in elastodynamic simulations. *Int J Numer Meth Eng* 123:4702–4725
22. Wriggers P, Hudobivnik B (2017) A low order virtual element formulation for finite elasto-plastic deformations. *Comput Methods Appl Mech Eng* 327:4702–4725
23. Hudobivnik B, Aldakheel F, Wriggers P (2019) A low order 3d virtual element formulation for finite elasto-plastic deformations. *Comput Mech* 63:4702–4725
24. Cihan M, Hudobivnik B, Aldakheel F, Wriggers P (2021) 3d mixed virtual element formulation for dynamic elasto-plastic analysis. *Comput Mech* 68:1–18
25. Wriggers P, Aldakheel F, Hudobivnik B (2024) *Virtual element methods in engineering sciences*. Springer, Berlin
26. Sukumar N, Tabarraei A (2004) Conformal polygonal finite elements. *Int J Numer Methods Eng* 61:2045–2066
27. Nguyen-Xuan H (2017) A polygonal finite element method for plate analysis. *Comput Struct* 188:45–62
28. Di Pietro DA, Ern A (2012) *Mathematical aspects of discontinuous Galerkin methods*. Springer, Berlin
29. Hesthaven J, Warburton T (2007) *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*, vol 54 (2007)
30. Li S, Cui X (2020) N-sided polygonal smoothed finite element method (NSFEM) with non-matching meshes and their applications for brittle fracture problems. *Comput Methods Appl Mech Eng* 359:112672
31. Wu S-W et al (2023) Arbitrary polygon mesh for elastic and elasto-plastic analysis of solids using smoothed finite element method. *Comput Methods Appl Mech Eng* 405:115874
32. Wriggers P, Hudobivnik B, Aldakheel F (2021) Nurbs-based geometries: a mapping approach for virtual serendipity elements. *Comput Methods Appl Mech Eng* 378:113732
33. Berrone S, Borio A, Marcon F, Teora G (2023) A first-order stabilization-free virtual element method. *Appl Math Lett* 142:108641
34. Meng J, Wang X, Bu L, Mei L (2022) A lowest-order free-stabilization virtual element method for the Laplacian eigenvalue problem. *J Comput Appl Math* 410:114013
35. Chen A, Sukumar N (2023) Stabilization-free serendipity virtual element method for plane elasticity. *Comput Methods Appl Mech Eng* 404:115784
36. Chen A, Sukumar N (2023) Stabilization-free virtual element method for plane elasticity. *Comput Math Appl* 138:88–105
37. Xu B-B, Peng F, Wriggers P (2023) Stabilization-free virtual element method for finite strain applications. *Comput Methods Appl Mech Eng* 417:116555
38. Taylor R (2023) VEM Prototypes. Private communication
39. Sorgente T, Biasotti S, Manzini G, Spagnuolo M (2022) Polyhedral mesh quality indicator for the virtual element method. *Comput Math Appl* 114:151–160
40. Beirao da Veiga L, Brezzi F, Marini LD, Russo A (2014) The Hitchhiker's guide to the virtual element method. *Math Models Methods Appl Sci* 24:1541–1573
41. Ahmad B, Alsaedi A, Brezzi F, Marini L, Russo A (2013) Equivalent projectors for virtual element methods. *Comput Math Appl* 66:376–391
42. Berrone S, Borio A, Marcon F (2021) Lowest order stabilization free virtual element method for the Poisson equation. [arXiv:2103.16896](https://arxiv.org/abs/2103.16896) (2021)
43. D'Altri A, Miranda S, Patruno L, Sacco E (2021) An enhanced VEM formulation for plane elasticity. *Comput Methods Appl Mech Eng* 376:113663
44. Lamperti A, Cremonesi M, Perego U, Russo A, Lovadina C (2023) A Hu–Washizu variational approach to self-stabilized virtual elements: 2D linear elastostatics. *Comput Mech* 71:1–21
45. Xu B-B, Wriggers P (2024) 3D stabilization-free virtual element method for linear elastic analysis. *Comput Methods Appl Mech Eng* 421:116826
46. Wriggers P (2008) *Nonlinear finite element methods*. Springer, Berlin

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