

Efficient virtual element formulations for compressible and incompressible finite deformations

P. Wriggers¹ · B. D. Reddy² · W. Rust¹ · B. Hudobivnik¹

Received: 9 February 2017 / Accepted: 27 March 2017 / Published online: 6 April 2017
© Springer-Verlag Berlin Heidelberg 2017

Abstract The virtual element method has been developed over the last decade and applied to problems in elasticity and other areas. The successful application of the method to linear problems leads naturally to the question of its effectiveness in the nonlinear regime. This work is concerned with extensions of the virtual element method to problems of compressible and incompressible nonlinear elasticity. Low-order formulations for problems in two dimensions, with elements being arbitrary polygons, are considered: for these, the ansatz functions are linear along element edges. The various formulations considered are based on minimization of energy, with a novel construction of the stabilization energy. The formulations are investigated through a series of numerical examples, which demonstrate their efficiency, convergence properties, and for the case of nearly incompressible and incompressible materials, locking-free behaviour.

Keywords Virtual element method · VEM · Mixed methods · Stabilization · Nonlinear elasticity

1 Introduction

The many different approaches to the approximate solution of problems involving partial differential equations include finite difference schemes, finite elements, finite volume techniques, boundary elements, and particle methods. Within the finite element method there have been various significant developments, including for example isogeometric schemes. Research continues to be motivated by the goal of developing stable, efficient and robust discretization schemes for finite deformation applications in solid mechanics.

While the finite element method is well established as an approach to handling problems in nonlinear solid mechanics, (see for example the texts by Bathe [1], Belytschko et al. [3] and Wriggers [38]) it is nevertheless of interest to explore new methods that have potential advantages such as flexibility with regard to mesh generation and choice of element shapes. The Discontinuous Galerkin method provides flexibility in the sense that meshing does not have to take into account hanging nodes: this can be an advantage in adaptive mesh procedures (see the works by ten Eyck and Lew [37] and Noels and Radovitzky [26] for applications to nonlinear elasticity).

There are situations in which it is advantageous not to be restricted in the choice of element shapes, and in this regard there have been interesting developments in the use of polygonal or polyhedral elements. Some representative works in this regard are those by Sukumar [34] and Sukumar and Malsch [35]; for finite deformation problems including contact, see Biabanaki and Khoei [5], Biabanaki et al. [6] and Chi et al. [10].

The virtual element method (VEM), an extension of the classical Galerkin finite element method, is a relatively recent development. The method permits the use of polygonal element for problems in two dimensions and polyhedral elements in three. Furthermore, there is no need for a restriction

✉ P. Wriggers
wriggers@ikm.uni-hannover.de

B. D. Reddy
daya.reddy@uct.ac.za

W. Rust
rust@ikm.uni-hannover.de

B. Hudobivnik
hudobivnik@ikm.uni-hannover.de

¹ Institute for Continuum Mechanics, Leibniz Universität Hannover, Appelstr.11, 30167 Hannover, Germany

² Centre for Research in Computational and Applied Mechanics, University of Cape Town, Cape Town, South Africa

to convex elements, nor is it necessary to avoid degeneracies such as element sides having an interior angle close to π radians. Thus the method permits the direct use of Voronoi meshers, and as another example crystals in a polycrystalline material can be represented by single elements. Key examples of the method are the works by Beirão da Veiga et al. [11, 13] and, for problems in elasticity [12, 16]. Stabilization procedures for the virtual element method that are using hourglass stabilization, well known from the work of Belytschko et al. [4], are described in Cangiani et al. [8] for linear Poisson problems. The use of VEM for nonlinear problems has been discussed recently in Beirão Da Veiga et al. [14] and Chi et al. [9] for inelastic materials at small strains and hyperelastic materials at finite deformations.

The structure of the VEM typically comprises a term in the weak formulation or energy functional in which the quantity φ_h being sought is replaced by its projection $\Pi\varphi_h$ onto a polynomial space. This results in a rank-deficient structure, so that it is necessary to add a stabilization term to the formulation. The stabilization term is generally a function of the difference $\varphi_h - \Pi\varphi_h$ between the original variable and its projection. In order to adhere to a fundamental aspect of VEM in which all integrations take place on element boundaries, the stabilization term proposed, for example, in Beirão da Veiga et al. [11, 12] takes the form of a sum of a function of nodal variables. This is the approach adopted in the recent nonlinear investigation by Chi et al. [9], with the scalar stabilization parameter of the linear case being replaced by one that depends on the fourth-order elasticity tensor (that is, the derivative of the relevant stress measure with respect to the deformation). This leads to a formulation that works very well when smaller load-steps are applied since the stabilization parameter is always related to the deformation of the last converged load step. In cases of problems that can run efficiently with few or even with only one load step, a different stabilization has to be employed that enables a consistent linearization at every deformation state, to yield a Newton scheme with quadratic convergence properties independent of load steps.

We present here a formulation of VEM that uses a different stabilization technique, first described in Nadler and Rubin [25], generalized in Boerner et al. [7] and simplified in Krysl [19] in the context of stabilization procedures for the mean-strain hexahedron. The essence of the method is the addition to the positive semidefinite mean strain energy Ψ a positive-definite energy $\hat{\Psi}$ which is evaluated using full quadrature, and for consistency subtraction of a term involving $\hat{\Psi}$ as a function of the mean strain. This approach is exploited here in the context of a VEM for nonlinear elasticity involving polygonal elements in two dimensions, and the use of low-order elements for which the projected quantities are linear polynomials. The strain energy is then the sum of $\Psi(\Pi\mathbf{u}_h)$, the original energy as a function of the projected displacement, and the term $\hat{\Psi}(\Pi\mathbf{u}_h)$ to which are respectively added

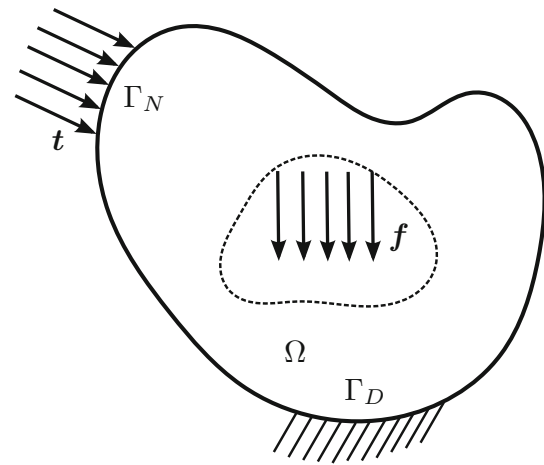


Fig. 1 Solid with boundary conditions

and subtracted a positive definite stabilization energy as a function of the displacement and its projection. The second feature of the formulation presented here is in the manner in which quadrature is carried out for the term involving $\hat{\Psi}(\mathbf{u}_h)$: this is achieved by constructing a mesh of triangles in the element with nodal points those for the original element, and by approximating the displacement as a linear function on each triangle. The method is efficient and leads to an explicit form for the integral.

The method presented here is developed for both compressible and incompressible materials, using the neo-Hookean strain energy as a model. For the compressible case behaviour is shown to locking-free for near-incompressibility, while for the incompressible case, which is based on a mixed formulation involving the pressure as a Lagrange multiplier, locking-free behaviour is also observed. Furthermore, both formulations perform extremely well in a series of benchmark tests involving regular, distorted and Voronoi meshes.

The structure of the rest of this work is as follows. The governing equations for nonlinear elasticity are presented in Sect. 2. Section 3 is devoted to a presentation of the VEM, including the formulation for incompressible materials. Section 3.3 discusses existing approaches to stabilization and presents the new stabilization term. A number of numerical examples are presented and discussed in Sect. 4, and in Sect. 5 some concluding remarks are presented.

2 Governing equations for finite elasticity

Consider an elastic body that occupies the bounded domain $\Omega \subset \mathbb{R}^2$. The body Ω has a boundary Γ which comprises non-overlapping sections Γ_D and Γ_N such that $\Gamma_D \cup \Gamma_N = \Gamma$ (Fig. 1).

The position \mathbf{x} of a material point initially at \mathbf{X} is given by the motion

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

where \mathbf{u} is the displacement. We also define the deformation gradient \mathbf{F} by

$$\mathbf{F} = \text{Grad } \boldsymbol{\varphi}, \tag{2}$$

the gradient being evaluated with respect to \mathbf{X} .

The body satisfy, on Ω the equation of equilibrium

$$- \text{Div } \mathbf{P} = \mathbf{f}, \tag{3}$$

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

$$\mathbf{u} = \bar{\mathbf{u}} \quad \text{on } \Gamma_D, \tag{4}$$

$$\mathbf{P}\mathbf{N} = \bar{\mathbf{t}} \quad \text{on } \Gamma_N. \tag{5}$$

with \mathbf{N} the outward unit normal vector, $\bar{\mathbf{u}}$ the prescribed displacement, and $\bar{\mathbf{t}}$ the surface traction on Γ_N .

By introducing a strain energy function $\Psi(\boldsymbol{\varphi})$ for elastic problems the first Piola–Kirchhoff stresses follow from

$$\mathbf{P}(\boldsymbol{\varphi}) = \frac{\partial \Psi(\boldsymbol{\varphi})}{\partial \mathbf{F}}. \tag{6}$$

For a homogeneous compressible isotropic hyperelastic material we adopt the neo-Hookean strain energy function for the two-dimensional case

$$\Psi(\mathbf{u}) = \frac{\lambda}{4} (J(\mathbf{u})^2 - 1 - 2 \ln J(\mathbf{u})) + \frac{\mu}{2} (\text{tr } \mathbf{C}(\mathbf{u}) - 2 - 2 \ln J(\mathbf{u})) \tag{7}$$

in which λ and μ are the Lamé constants. This strain energy is known as Neo-Hookean model. The right Cauchy-Green tensor $\mathbf{C}(\mathbf{u})$ is defined as $\mathbf{C} = \mathbf{F}^T \mathbf{F}$ and the Jacobian $J(\mathbf{u})$ of the deformation is given as $J = \det \mathbf{F}$.

For incompressible materials the strain energy can be based on the split of \mathbf{C} into an isochoric part $\bar{\mathbf{C}}$, defined by

$$\bar{\mathbf{C}} = \det \mathbf{C}^{-\frac{1}{2}} \mathbf{C} \tag{8}$$

and a part related to the incompressibility constraint. The strain energy (7) then reduces to

$$\Psi^i(\mathbf{u}) = \frac{\mu}{2} (\text{tr } \bar{\mathbf{C}}(\mathbf{u}) - 2). \tag{9}$$

The development of a hyperelastic virtual element can start from the potential energy function directly instead of

using the weak form. In that case the potential energy can be written with (7) as

$$U(\mathbf{u}) = \int_{\Omega} [\Psi(\mathbf{u}) - \mathbf{f} \cdot \mathbf{u}] \, d\Omega - \int_{\Gamma_N} \mathbf{t} \cdot \mathbf{u} \, d\Gamma \tag{10}$$

For incompressible deformation the potential energy is given by

$$\begin{aligned} U^i(\mathbf{u}) &= \int_{\Omega} [\Psi^i(\mathbf{u}) - \mathbf{f} \cdot \mathbf{u}] \, d\Omega \\ &+ \int_{\Omega} p [J(\mathbf{u}) - 1] - \int_{\Gamma_N} \mathbf{t} \cdot \mathbf{u} \, d\Gamma \end{aligned} \tag{11}$$

where the incompressibility constraint is added via a Lagrangian parameter p which is interpreted as the pressure.

3 Formulation of the virtual element method

The main idea of the virtual element method (VEM) is a Galerkin projection of the deformation onto a specific ansatz space. The domain Ω is partitioned into non-overlapping polygonal elements which need not be convex.

Here a low-order approach is adopted, using linear ansatz functions. Hence element nodes are placed only at the vertices of the polygonal elements. The discrete space of test functions on Ω is denoted by V_h , and for a conforming approach we require that $V_h \subset V$. This requirement is met by defining the shape or basis functions in V_h to comprise continuous functions whose restriction to an element Ω_e includes (but is larger than) linear functions. Furthermore, the restriction of the element shape functions to the element boundaries are linear functions (Fig. 2).

3.1 VEM ansatz functions

Generally the virtual element method relies on the split of the ansatz space into a linear part $\Pi \mathbf{u}_h$ and a remainder

$$\mathbf{u}_h = \Pi \mathbf{u}_h + (\mathbf{u}_h - \Pi \mathbf{u}_h) \tag{12}$$

The projection Π is defined at element level by

$$\Pi \mathbf{u}_h = \mathbf{H} \mathbf{a} = \begin{bmatrix} 1 & 0 & x & 0 & y & 0 \\ 0 & 1 & 0 & x & 0 & y \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \\ \dots \\ a_6 \end{Bmatrix} \tag{13}$$

The linear ansatz for the deformation along the element edge is given for a boundary segment k of the virtual element, defined by the local nodes (1)–(2) by, see right side of Fig. 3,

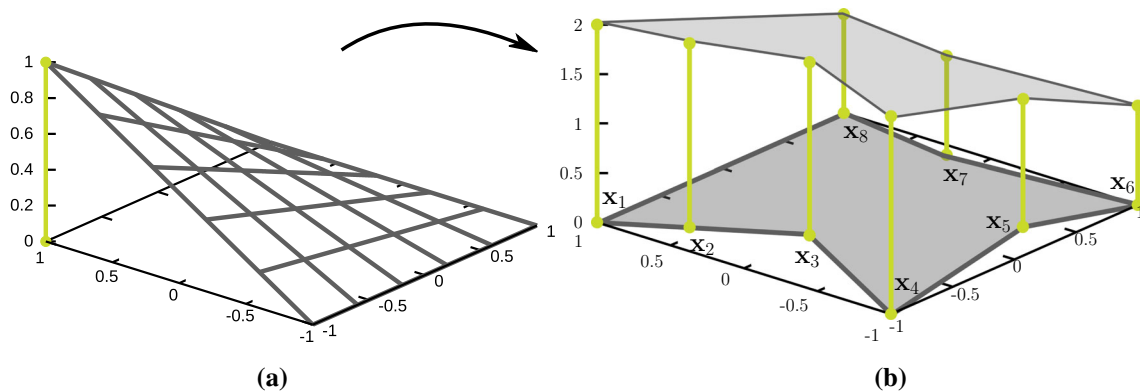


Fig. 2 Comparison of the ansatz functions for FEM and VEM formulations. **a** Finite element ansatz, **b** polynomial basis function for the virtual element ansatz with vertices \mathbf{x}_l

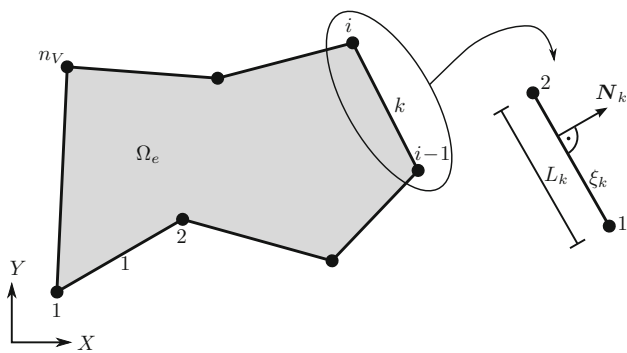


Fig. 3 Virtual element with n_V nodes and local boundary segment of the element

$$(\mathbf{u}_h)_k = (1 - \xi_k) \mathbf{u}_1 + \xi_k \mathbf{u}_2 = M_{k1} \mathbf{u}_1 + M_{k2} \mathbf{u}_2 \quad \text{with} \quad \xi_k = \frac{x_k}{L_k} \tag{14}$$

where, for example, M_{k1} is the ansatz function along a segment k related to node (1), ξ_k is the local dimensionless coordinate and \mathbf{u}_1 is the nodal value at that node, see Fig. 3.

The projection Π is defined such that it satisfies, see Beirão Da Veiga et al. [14],

$$\nabla \Pi \mathbf{u}_h|_e \stackrel{!}{=} \frac{1}{\Omega_e} \int_{\Omega_e} \text{Grad } \mathbf{u}_h \, d\Omega = \frac{1}{\Omega_e} \int_{\Gamma_e} \mathbf{u}_h \otimes \mathbf{N} \, d\Gamma \tag{15}$$

where \mathbf{N} is the normal at the boundary Γ_e of the domain Ω_e .

From (13), the gradient of the projection is thus given by

$$\nabla \Pi \mathbf{u}_h|_e = \begin{bmatrix} a_3 & a_5 \\ a_4 & a_6 \end{bmatrix} \tag{16}$$

which is constant at element level.

The right hand side of (15) yields with (14)

$$\begin{aligned} & \frac{1}{\Omega_e} \int_{\Gamma_e} \mathbf{u}_h \otimes \mathbf{N} \, d\Gamma \\ &= \frac{1}{\Omega_e} \sum_{k=1}^{n_V} \int_{\Gamma_k} \begin{bmatrix} u_x(\mathbf{x}_k) N_x & u_x(\mathbf{x}_k) N_y \\ u_y(\mathbf{x}_k) N_x & u_y(\mathbf{x}_k) N_y \end{bmatrix} L_k \, d\Gamma \end{aligned} \tag{17}$$

where we have used $\mathbf{N} = \{N_x, N_y\}^T$ and $\mathbf{u} = \{u_x, u_y\}^T$, and n_V are the number of segments of the element. Note that the normal vector \mathbf{N} changes from segment to segment. In the 2D case it can be computed for a segment k as

$$\mathbf{N}_k = \frac{1}{L_k} \begin{Bmatrix} N_x \\ N_y \end{Bmatrix}_k = \frac{1}{L_k} \begin{Bmatrix} -(Y_2 - Y_1) \\ X_2 - X_1 \end{Bmatrix}_k \tag{18}$$

All quantities are related to the undeformed or initial configuration. Furthermore we have used a form of \mathbf{N}_k that is not normalized since the length L_k cancels out when the integral in (17) is evaluated over the edges.

The integral in (17) can be evaluated for the ansatz functions (14) exactly by using the trapezoidal or Gauss–Lobatto rule. By selecting the vertices as the Gauss–Lobatto points it is sufficient to know the values $\hat{\mathbf{u}}$ at the n_V vertices V (see Beirão da Veiga et al. [13]). Since the ansatz function in (14) fullfills the property $M_I(x_J) = \delta_{IJ}$ the actual form of the function \mathbf{N} does not enter the evaluation of the boundary integrals.

Hence by comparing (16) and (17) the unknowns a_3 to a_6 are obtained by inspection where all contributions related to the segments have to be added

$$\begin{aligned} a_3 &= \frac{1}{2\Omega_e} \left[N_{x1}(u_{x1} + u_{x2}) + \dots + N_{xV-1}(u_{x_{nV-1}} + u_{x_{nV}}) \right. \\ & \quad \left. + N_{xV}(u_{x_{nV}} + u_{x1}) \right] \\ a_4 &= \frac{1}{2\Omega_e} \left[N_{x1}(u_{y1} + u_{y2}) + \dots + N_{xV-1}(u_{y_{nV-1}} + u_{y_{nV}}) \right. \end{aligned}$$

$$\begin{aligned}
 &+ N_{xV}(u_{ynV} + u_{y1})] \\
 a_5 &= \frac{1}{2\Omega_e} [N_{y1}(u_{x1} + u_{x2}) + \dots + N_{yV-1}(u_{xnV-1} + u_{xnV}) \\
 &+ N_{yV}(u_{xnV} + u_{x1})] \\
 a_6 &= \frac{1}{2\Omega_e} [N_{y1}(u_{y1} + u_{y2}) + \dots + N_{yV-1}(u_{ynV-1} + u_{ynV}) \\
 &+ N_{yV}(u_{ynV} + u_{y1})] \tag{19}
 \end{aligned}$$

This projection determines the ansatz $\Pi \mathbf{u}_h$ in (13) within an element only up to a constant strain field and has to be supplemented by a further condition to ensure uniqueness. For this purpose we adopt the condition (see for example Beirão Da Veiga et al. [12]) that the sum of the nodal values of \mathbf{u}_h and of its projection $\Pi \mathbf{u}_h$ are equal. This yields for each element Ω_e

$$\frac{1}{n_V} \sum_{I=1}^{n_V} \Pi \mathbf{u}_h(\mathbf{x}_I) = \frac{1}{n_V} \sum_{I=1}^{n_V} \mathbf{u}_h(\mathbf{x}_I), \tag{20}$$

where \mathbf{x}_I are the coordinates of the nodal point I and the sum includes all boundary nodes.

From (13) and (14) we have, substituting in (20)

$$\underbrace{\frac{1}{n_V} \sum_I \mathbf{H}(\mathbf{x}_I) \mathbf{a}}_{\bar{\mathbf{H}}_T} = \frac{1}{n_V} \sum_I \mathbf{N}(\mathbf{x}_I) \mathbf{u}_I = \underbrace{\frac{1}{n_V} \begin{bmatrix} 1 & 0 & \dots & 1 & 0 \\ 0 & 1 & \dots & 0 & 1 \end{bmatrix}}_{\bar{\mathbf{N}}_T} \begin{Bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \dots \\ \mathbf{u}_{nV} \end{Bmatrix} \tag{21}$$

These two equations can now be used to determine the two unknowns a_1 and a_2 . Using the already computed unknowns a_i , see (19) we obtain

$$\begin{aligned}
 a_1 &= \sum_{I=1}^{n_V} [u_{xI} - a_3 x_I - a_5 y_I] / n_V \\
 a_2 &= \sum_{I=1}^{n_V} [u_{yI} - a_4 x_I - a_6 y_I] / n_V \tag{22}
 \end{aligned}$$

With this result the ansatz function $\Pi \mathbf{u}_h$ of the virtual element is completely defined.

Note that the constant gradient $\nabla \Pi \mathbf{u}_h|_e$ can be computed directly using (16) and (17). Thus for the computation of the strain energy it is not necessary to evaluate Eq. (20).

3.2 Construction of the virtual element

Within the virtual element method the gradient of the displacement field is approximated by a constant part that stems

from the projection that has been discussed in the last section. A construction of a virtual element which is based only on this projection would lead to a rank deficient element once the number of vertices is greater than 3. Thus the formulation has to be stabilized, as in the case of the classical one-point integrated elements developed by Flanagan and Belytschko [15], Belytschko and Bindeman [2], Reese et al. [29], Reese and Wriggers [30], Reese [28], Mueller-Hoeppe et al. [24], Korelc et al. [17], Krysl [20], to mention some key contributions.

In the following we will first discuss the formulation of the element part that stems from the projection, see last section, for the compressible and incompressible cases. Furthermore, different possibilities for stabilizing the virtual element method will be considered.

3.2.1 Constant part due to projection

The simplest possible formulation for a finite deformation virtual element is a split into a constant part of the deformation gradient and an associated stabilization term. This was performed for the linear case in Beirão Da Veiga et al. [12] and also in Wriggers et al. [39] where the focus was on contact mechanics. The same approach can be found in the work of Beirão Da Veiga et al. [14] and Chi et al. [9] for the nonlinear case.

Compressible case Here we employ the same approach, however not by starting from the weak form but from the hyperelastic potential function (10). Thus we have, by summing up all element contributions for the n_e virtual elements,

$$U(\mathbf{u}) = \sum_{e=1}^{n_e} [U_c(\Pi \mathbf{u}_h|_e) + U_{stab}(\mathbf{u}_h|_e - \Pi \mathbf{u}_h|_e)]. \tag{23}$$

The first part in this equation can be computed by using the results obtained in the last section. This yields

$$\begin{aligned}
 U_c(\Pi \mathbf{u}_h|_e) &= \int_{\Omega_e} [\Psi(\Pi \mathbf{u}_h|_e) - \mathbf{f} \cdot \Pi \mathbf{u}_h|_e] d\Omega \\
 &- \int_{\Gamma_e^\sigma} \mathbf{t} \cdot \Pi \mathbf{u}_h|_e d\Gamma. \tag{24}
 \end{aligned}$$

It is clear from the expression

$$\begin{aligned}
 \Psi(\Pi \mathbf{u}_h|_e) &= \frac{\lambda}{4} (J(\Pi \mathbf{u}_h|_e)^2 - 1 - 2 \ln J(\Pi \mathbf{u}_h|_e)) \\
 &+ \frac{\mu}{2} (\text{tr } \mathbf{C}(\Pi \mathbf{u}_h|_e) - 2 - 2 \ln J(\Pi \mathbf{u}_h|_e)) \tag{25}
 \end{aligned}$$

that for the compressible case the strain energy depends on functions that are constant in each element Ω_e since the deformation gradient, see (16) and (19), is constant:

$$\mathbf{F}_e = \mathbf{1} + \nabla \Pi \mathbf{u}_h|_e. \tag{26}$$

Thus

$$\int_{\Omega_e} [\Psi(\Pi \mathbf{u}_h|_e)] d\Omega = \Psi(\Pi \mathbf{u}_h|_e) \Omega_e \tag{27}$$

where Ω_e is the area of the virtual element. Note, however, that the strain energy $\Psi(\Pi \mathbf{u}_h|_e)$ is still a nonlinear function with respect to the displacement nodal degrees of freedom.

Now the strain energy function in (24) can be evaluated using the approximation at element level of the deformation gradient \mathbf{F}_e , see (16), which depends on the nodal degrees of freedom, see (19). Likewise, the Jacobian $J_e = \det \mathbf{F}_e$ and the right Cauchy–Green tensor $\mathbf{C}_e = \mathbf{F}_e^T \mathbf{F}_e$ can be evaluated simply. All derivations leading to the residual vector \mathbf{R}_e^c and the tangent matrix $\mathbf{K}_{T_e}^c$ were performed with the symbolic tool ACEGEN, see Korelc and Wriggers [18]. This yields for (24)

$$\mathbf{R}_e^c = \frac{\partial U_c(\Pi \mathbf{u}_h|_e)}{\partial \mathbf{u}_e} \quad \text{and} \quad \mathbf{K}_{T_e}^c = \frac{\partial \mathbf{R}_e^c(\mathbf{u}_e)}{\partial \mathbf{u}_e} \tag{28}$$

where \mathbf{u}_e are the nodal displacements of the virtual element Ω_e .

Incompressible case The same procedure can be used for the case of incompressible materials. The first part in Eq. (23) can be computed by using the results obtained in the last section and the functional (11). This yields

$$U_c^i(\Pi \mathbf{u}_h|_e) = \int_{\Omega_e} [\Psi^i(\Pi \mathbf{u}_h|_e) - \mathbf{f} \cdot \Pi \mathbf{u}_h|_e] d\Omega + \int_{\Omega_e} p [J(\mathbf{u}_e) - 1] d\Omega - \int_{\Gamma_e^g} \mathbf{t} \cdot \Pi \mathbf{u}_h|_e d\Gamma. \tag{29}$$

From the expression

$$\Psi^i(\Pi \mathbf{u}_h|_e) = \frac{\mu}{2} [\text{tr} \bar{\mathbf{C}}(\Pi \mathbf{u}_h|_e) - 2] \tag{30}$$

the strain energy depends on a quantity that is constant in each element. Thus we have

$$\int_{\Omega_e} [\Psi^i(\Pi \mathbf{u}_h|_e)] d\Omega = \Psi^i(\Pi \mathbf{u}_h|_e) |\Omega_e|. \tag{31}$$

However $\Psi^i(\Pi \mathbf{u}_h|_e)$ is still a nonlinear function with respect to the nodal degrees of freedom \mathbf{u}_I .

The incompressibility constraint is discretized by constant terms; however, the Jacobian J_e within an element is not computed as $J_e = \det \mathbf{F}_e$ but rather as was done in Chi et

al. [9], in a different context. For a constant approximation it follows for the two-dimensional case that $J = \omega_e / \Omega_e$ where ω_e is the area of the element in the deformed configuration and Ω_e the area with respect to the undeformed configuration. In the spirit of the virtual element method the deformed and undeformed areas are computed using an integral over the boundary:

$$\Omega_e = \frac{1}{2} \int_{\Gamma_e} \mathbf{X}_e \cdot \mathbf{N}_e d\Gamma, \quad \omega_e = \frac{1}{2} \int_{\gamma_e} \mathbf{x}_e \cdot \mathbf{n}_e d\gamma. \tag{32}$$

Here Γ_e denotes the undeformed element edge while γ_e is the deformed edge of the virtual element. The normal vector \mathbf{N}_e is computed using (18) for an edge of the virtual element. In the same way the deformed normal \mathbf{n}_e can be computed by inserting in (18) the deformed configuration $\mathbf{x}_e = \mathbf{X}_e + \mathbf{u}_e$ where \mathbf{X}_e is the vector to the boundary in the undeformed configuration. For the virtual element depicted in Fig. 3 one can directly evaluate the first term in (32) as

$$\Omega_e = \frac{1}{2} [N_{y1}(X_1 + X_2) + \dots + N_{yV-1}(X_{nV-1} + X_{nV}) + N_{yV}(X_{nV} + X_1)] - \frac{1}{2} [N_{x1}(Y_1 + Y_2) + \dots + N_{xV-1}(Y_{nV-1} + Y_{nV}) + N_{xV}(Y_{nV} + Y_1)]. \tag{33}$$

The evaluation of the second term in (32) is similar, with the normal vector components and coordinates being replaced by their values in the deformed configuration.

As a result, the incompressibility constraint (29) can be written

$$\int_{\Omega_e} p [J(\mathbf{u}_e) - 1] d\Omega = p_e (\omega_e - \Omega_e) \tag{34}$$

when the pressure p is approximated as constant on each element. This element fulfills incompressibility at element level. It will be identified by VEM-T1-I in the example section.

If nearly incompressible materials are considered the classical Q1-P0 formulation can be employed. In this formulation an additional unknown θ is introduced and the incompressibility constraint (29) is relaxed, leading to a Hu–Washizu formulation (see Simo et al. [33])

$$\int_{\Omega_e} \left(p [J(\mathbf{u}_e) - \theta] + \frac{K}{2} (\theta - 1)^2 \right) d\Omega \tag{35}$$

where the penalty parameter K can be identified as modulus of compression. By choosing a constant ansatz θ_e for θ the integrals can be evaluated as before leading to

$$\int_{\Omega_e} \left(p [J(\mathbf{u}_e) - \theta] + \frac{K}{2} (\theta - 1)^2 \right) d\Omega = p_e (\omega_e - \theta_e \Omega_e) + \frac{K}{2} (\theta_e - 1)^2 \Omega_e. \tag{36}$$

This formulation allows an elimination of the variables p_e and θ_e at element level and thus does not introduce further unknowns. It will be denoted in the following by VEM-T1-P0.

All derivations with respect to the unknown displacements leading to the residual vector \mathbf{R}_e^c and the tangent matrix \mathbf{K}_{Te}^c were performed with the symbolic tool ACEGEN, see Korelc and Wriggers [18]. This yields for (24)

$$\mathbf{R}_e^c = \frac{\partial U_c^i(\Pi \mathbf{u}_h|_e)}{\partial \mathbf{u}_e} \quad \text{and} \quad \mathbf{K}_{Te}^c = \frac{\partial \mathbf{R}_e^c(\mathbf{u}_e)}{\partial \mathbf{u}_e} \tag{37}$$

where \mathbf{u}_e are the nodal displacements of the virtual element Ω_e . The residual and tangent for the incompressible case are evaluated similarly.

3.3 Stabilization techniques for nonlinear virtual elements

Several approaches to stabilization can be followed, among them hour-glass stabilization techniques, enhanced and assumed strain methods, and other mixed technologies. In the literature on virtual element technologies so far stabilization techniques were discussed that work well for classical solid mechanics problems, see e.g. Beirão da Veiga et al. [11], Beirão Da Veiga et al. [14] and Chi et al. [9], but these stabilization methods eventually exhibit deficiencies once bending dominated response is present (like a simple beam bending problem where the trace or norm of the constitutive tensor is not related to the bending energy), or if they cannot be easily linearized in a consistent manner since it is almost impossible to linearize the norm or trace of the constitutive tensor consistently which depends in a complex manner on the displacement field.

A classical stabilization technique that is based on a mixed formulation is the enhanced assumed strain approach, see e.g. Simo and Rifai [32] and Simo and Armero [31]. However, as it was shown in Reddy and Simo [27] it is not possible to enhance displacement triangular elements since the enhanced modes are orthogonal to the deformation modes. This is also true for the virtual element formulations since the ansatz function in (13) does not differ from the ansatz function for triangular elements. Thus a different stabilization technique has to be designed on the basis that it should be independent of the problem to be solved and be able to yield a quadratically converging solution algorithm. Here it should be noted, that there exist possibilities to enhance tri-

angular elements when different functionals are introduced, see e.g. Taylor [36] and Lovadina and Auricchio [23].

3.3.1 Stabilization by a discrete bi-linear form

One way to formulate the stabilization part U_{stab} is to use the split in Eq. (12), see Beirão da Veiga et al. [11]. The stabilization term is then constructed as a sum over all nodes of a positive-definite function involving the difference: that is,

$$U_{stab} = \frac{\gamma}{2} \sum_{I=1}^{n_V} [\mathbf{u}_I - \bar{\mathbf{H}}(\mathbf{x}_I)\mathbf{a}] \cdot [\mathbf{u}_I - \bar{\mathbf{H}}(\mathbf{x}_I)\mathbf{a}] \tag{38}$$

where γ is a stabilization parameter. Here $\bar{\mathbf{H}}$ is an alternative way of writing \mathbf{H} in (13) and is given by

$$\bar{\mathbf{H}} = \begin{bmatrix} \underbrace{1 \ 0}_{\text{translation}} & \underbrace{\begin{matrix} \text{rotations} \\ -y^* & x^* \end{matrix}}_{\text{axial strain}} & \underbrace{\begin{matrix} \text{shear strain} \\ y^* & x^* \end{matrix}}_{\text{volumetric strain}} \end{bmatrix}. \tag{39}$$

This takes care of the different deformation modes within the element, see Gain et al. [16] and Wriggers et al. [39]. Since \mathbf{a} is known, see (19) and (22), all terms are now given as functions of the unknown nodal displacements and thus the element residual and tangent stiffness can be derived.

In Beirão Da Veiga et al. [14] this parameter is replaced by a term that depends on the norm of the constitutive tensor $\|\frac{\partial \mathbf{P}}{\partial \mathbf{F}}(\mathbf{F}_e)\|$, and hence on the deformation of the solid. An alternative stabilization parameter $\frac{1}{4} \text{tr}[\frac{\partial \mathbf{P}}{\partial \mathbf{F}}(\mathbf{F}_e)]$, referred to as a trace-based stabilization, was recently proposed in Chi et al. [9]. Both stabilization parameters yield a far better approximation than the constant parameter γ . However, such a term presents a challenge when linearizing in a Newton procedure, even with the use of the symbolic tool AGEGEN, see Korelc and Wriggers [18], which automatically provides the code for the residual vector and tangent stiffness matrix. Thus it has to be computed at the last converged load step, which leads to incremental load stepping, even if the physical problem allows the computation of the solution in one step.

3.3.2 Nonlinear stabilization

We present here a novel stabilization approach which is based on that proposed by Krysl [19]. The essence of the approach is to introduce a new, positive definite strain energy $\hat{\Psi}$ and to define the stabilization contribution to the strain energy by

$$U_{stab}(\mathbf{u}_h|_e - \Pi \mathbf{u}_h|_e) = \hat{U}(\mathbf{u}_h|_e) - \hat{U}(\Pi \mathbf{u}_h|_e). \tag{40}$$

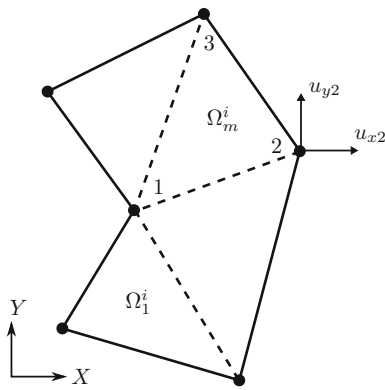


Fig. 4 Internal triangular mesh

The second term on the right side ensures the consistency of the total potential energy, which is now given by

$$U(\mathbf{u}_h) = U(\Pi\mathbf{u}_h) + \hat{U}(\mathbf{u}_h) - \hat{U}(\Pi\mathbf{u}_h), \tag{41}$$

and in which for example

$$\hat{U}(\mathbf{u}_h) = \sum_{e=1}^{n_e} \int_{\Omega_e} \hat{\Psi}(\mathbf{u}_h|_e) d\Omega. \tag{42}$$

The terms involving $\Pi\mathbf{u}_h$ can be integrated as (24). It remains to devise a procedure for computing the term involving the displacement $\mathbf{u}_h|_e$.

The idea is now to approximate the displacement field $\mathbf{u}_h|_e$ in (12) by an inscribed triangular finite element mesh, see Fig. 4. It consists of n_{int} linear three-noded triangles that are connected to the nodes of the virtual element. This mesh can then be used to compute the stabilization energy. This specific choice of the mesh does not introduce extra degrees of freedom.

Note that the subtraction of the piecewise-constant part $\hat{U}(\Pi\mathbf{u}_h)$ is necessary to ensure that the stabilization energy does not influence the converged result. Once the element size is very small, a constant strain will occur in each virtual element. In that case $\hat{U}(\mathbf{u}_h)$ will approach $\hat{U}(\Pi\mathbf{u}_h)$ and thus not influence the final result.

It remains to define the strain energy $\hat{\Psi}$. Following the approach advocated in Krysl [19], we propose the stabilization strain energy

$$\hat{U}(\mathbf{u}_h|_e) = \sum_{m=1}^{n_{int}} \int_{\Omega_m^i} \left[\frac{\hat{\lambda}}{2} (J_m - 1)^2 + \frac{\hat{\mu}}{2} (\text{tr } \mathbf{C}_m - 2 - 2 \ln J_m) \right] d\Omega. \tag{43}$$

Once an ansatz is formulated for the approximation within each triangle Ω_m^i of the inscribed mesh for the displacement

field, here denoted by \mathbf{u}_m the deformation measures J_m and \mathbf{C}_m can be easily computed in the standard way, either by using an isoparametric formulation for the three noded triangle or by direct evaluation of the ansatz functions. The latter yields with the ansatz

$$\begin{Bmatrix} u_x \\ u_y \end{Bmatrix}_m = \begin{Bmatrix} d_1 \\ d_4 \end{Bmatrix}_m + \begin{Bmatrix} d_2 \\ d_5 \end{Bmatrix}_m x_m + \begin{Bmatrix} d_3 \\ d_6 \end{Bmatrix}_m y_m \tag{44}$$

an explicit expression for the displacement gradient

$$\text{Grad } \mathbf{u}_m = \begin{bmatrix} d_2 & d_3 \\ d_5 & d_6 \end{bmatrix}_m. \tag{45}$$

The values d_2, d_3, d_5 and d_6 depend on the nodal displacements of the triangle. Their explicit form is

$$\begin{Bmatrix} d_2 \\ d_3 \end{Bmatrix}_m = \frac{1}{2\Omega_m^i} \begin{bmatrix} y_2 - y_3 & y_3 - y_1 & y_1 - y_2 \\ x_3 - x_2 & x_1 - x_3 & x_2 - x_1 \end{bmatrix}_m \begin{Bmatrix} u_{x1} \\ u_{x2} \\ u_{x3} \end{Bmatrix}_m = \mathbf{D}_m \mathbf{u}_{xm}. \tag{46}$$

Here \mathbf{u}_{xm} are the x -components of the displacements at the nodal points of the interior triangles, x_i and y_i with $i = 1, 2, 3$ are the nodal coordinates and Ω_m^i is the element area. The expression for d_5 and d_6 can now be computed in a similar way with the y -components \mathbf{u}_{ym} of the nodal displacements, to yield

$$\begin{Bmatrix} d_5 \\ d_6 \end{Bmatrix}_m = \mathbf{D}_m \mathbf{u}_{ym}. \tag{47}$$

The gradient $\text{Grad } \mathbf{u}_m$ in (45) is constant over each inscribed element. Hence the deformation gradient $\mathbf{F}_m^i = \mathbf{1} + \text{Grad } \mathbf{u}_m$ within each element Ω_m^i is constant as well. The deformation gradient is then used to compute the Jacobi determinant $J_m^i = \det \mathbf{F}_m^i$ and the right Cauchy Green tensor $\mathbf{C}_m^i = (\mathbf{F}_m^i)^T \mathbf{F}_m^i$ that are needed to evaluate (43). Now the stabilization energy can be computed using (43) which is a nonlinear function of the nodal displacements of the interior triangular mesh. All further derivations leading to the residual vector \mathbf{R}_e and the tangent matrix \mathbf{K}_{Te} were performed with the symbolic tool ACEGEN, see Korelc and Wriggers [18]. This yields for (40)

$$\mathbf{R}_e^s = \frac{\partial U_{stab}(\mathbf{u}_e)}{\partial \mathbf{u}_e} \quad \text{and} \quad \mathbf{K}_{Te}^s = \frac{\partial \mathbf{R}_e^s(\mathbf{u}_e)}{\partial \mathbf{u}_e}. \tag{48}$$

Thus the total residual and tangent matrix of the virtual element are given by the sum of expressions (37) and (48): $\mathbf{R}_e = \mathbf{R}_e^c + \mathbf{R}_e^s$ and $\mathbf{K}_{Te} = \mathbf{K}_{Te}^c + \mathbf{K}_{Te}^s$.

The values of the Lamé parameters in the strain energy (43) have to be selected in a proper way. Krysl [21] suggested for hexahedra a procedure that takes into account the element

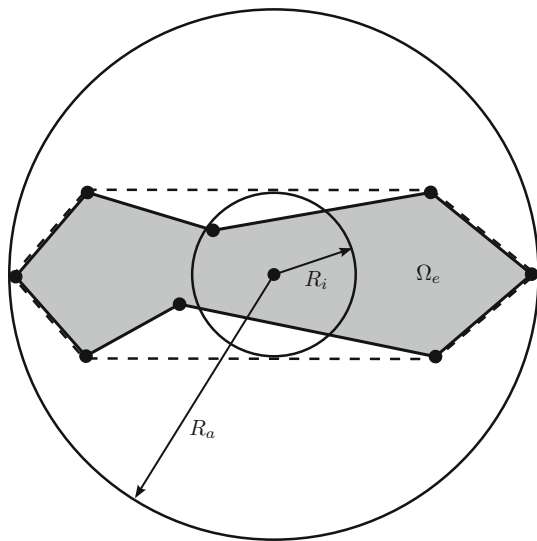


Fig. 5 Inner and outer radius of a virtual element

distortion to compute the Lamé parameters $\hat{\lambda}$ and $\hat{\mu}$. Since this procedure is not directly applicable to arbitrary virtual elements we suggest a simple and computationally efficient way to compute the parameters from the basic geometric data of the virtual element, see Fig. 5.

First the Lamé parameters, related to the problem, are converted to Young’s modulus and Poisson ratio. The resulting Young’s modulus can now be changed based on the length to height ratio of the element. This procedure was developed in Krysl [21] by comparing the bending energy of a rectangular block with that of a beam in order to enhance the bending behaviour of the element. This yields then a correction

$$\hat{E} = E \frac{\beta}{1 + \beta} \quad \hat{\nu} = 0.3. \tag{49}$$

for the Young’s modulus and the Poisson ratio of the brick element which is the same in two- and three dimensional formulations. Note that $\hat{\nu}$ is kept constants since the Poisson ratio does not influence the convergence behaviour of the element and avoids locking in the stabilization term for incompressible problems. We note that a similar procedure was used in the work of Nadler and Rubin [25] who stabilized their Cosserat brick element using 18 deformation modes for bending and torsion and matched the stiffness of these modes by comparisons with linear elastic solutions. Extensions for non-rectangular elements were provided in Loehnert et al. [22] and Boerner et al. [7]. Another strategy which used a stabilization of a mean strain hexahedron by enhanced modes was presented in Mueller-Hoeppe et al. [24] where the enhanced strain energy was formulated as in the geometrically linear theory like in Nadler and Rubin [25].

The factor β depends on the height to length ratio of the element. For a virtual element with arbitrary shape this ratio

can be computed by using the inner and outer radii, R_i^2, R_a^2 respectively, see Fig. 5, to obtain

$$\beta = 2\sqrt{2} (1 + \nu) \frac{R_i^2}{R_a^2 - R_i^2}. \tag{50}$$

The inner radius is computed by using the distance from the geometrical centre to the convex hull of the virtual element while the outer radius is defined by the maximum distance of nodes related to the virtual element. The computation of β is different from the scaling that was introduced in Krysl [19] and additionally takes into account that for a square shaped element the ratio of length to height is 1.

Now the Lamé parameters are obtained from

$$\hat{\lambda} = \frac{\hat{E} \hat{\nu}}{(1 + \hat{\nu})(1 - 2\hat{\nu})} \quad \hat{\mu} = \frac{\hat{E}}{2(1 + \hat{\nu})} \tag{51}$$

In the incompressible case the same stabilization is applied, leading to

$$U_{stab}^i(\mathbf{u}_h|_e - \Pi \mathbf{u}_h|_e) = \hat{U}^i(\mathbf{u}_h|_e) - \hat{U}^i(\Pi \mathbf{u}_h|_e). \tag{52}$$

with \hat{U}^i being given by

$$\hat{U}^i(\mathbf{u}_h|_e) = \sum_{i=1}^{n_{int}} \int_{\Omega_m^i} \frac{\hat{\mu}}{2} (\text{tr } \mathbf{C}_m - 2) d\Omega. \tag{53}$$

4 Numerical examples

In this section we will compare the new virtual element formulation with existing state-of-the-art finite elements. All examples are subjected to loads that lead to finite deformation strain states.

All computations are performed by using a Newton-Rapson algorithm with load stepping when necessary. Due to the fact that all formulations are linearized in a consistent manner, using ACEGEN, quadratic convergence is achieved.

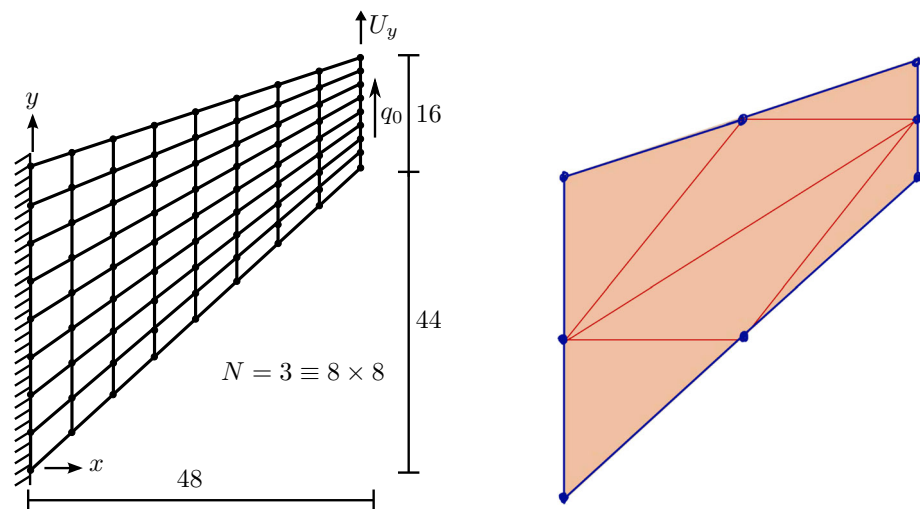
4.1 Cook’s membrane problem

The first example is Cook’s membrane problem of a tapered cantilever beam, clamped at the left end. The structure is loaded at the right end by a constant distributed vertical load q_0 , as depicted in Fig. 6.

The selected constitutive parameters for the Lamé constants are $\mu = 40$ and $\lambda = 100$. The distributed load is given as $q_0 = 4$.

The mesh used was automatically generated by the meshing tools in *Mathematica*. These had to be adapted to the different ways that are used to create meshes. In general we

Fig. 6 Initial configuration of Cook's membrane and internal mesh for a regular shaped virtual element



use different mesh types: a regular mesh, a severely distorted mesh and a Voronoi mesh, to show the performance of the new element.

The regular mesh is based on an 8 noded element that has an internal triangular mesh as depicted by the red lines in the right part of Fig. 6. The internal mesh is an arbitrary choice, it could have been constructed in a different way.

A selection of the different meshes can be found in Fig. 7 where three types of distorted meshes are shown. The left mesh is a mesh that can be obtained from the standard mesh in Fig. 6 by randomly changing the nodal placements within the standard mesh. In the middle a mesh is depicted that is generated by using again a random change of the nodal placement, but here also the side nodes are included and thus a mesh with convex and non-convex shaped elements is obtained. Finally on the right side of Fig. 7, a Voronoi mesh is shown that includes arbitrary number of nodes and element sizes. In all distorted cases a special optimization procedure provided in *Mathematica* has been used to generate the internal mesh.

Different mesh densities were employed to compute the solution for all generated types of meshes. The mesh refinement is uniform in the sense that the finer meshes are included in the coarser meshes for the regular and distorted mesh types of type d1 and d2. This enables convergence studies that will depict differences of the formulations. The number N denotes the mesh division: for example, in the left part of Fig. 6, a mesh with division of $2^3 = 8$ is depicted. The sequence of N that was used in this study is $N = 1, 2, 3, 4, 5, 6, 7, 8$. Note that the final mesh density leads to roughly 4×10^5 degrees of freedom. The Voronoi meshes are generated using randomly distributed nodes.

In order to test the robustness of the virtual element formulation the load was applied in one load step. Due to the consistent linearization of the element residual quadratic convergence is achieved in this load step. The deformation state, depicted in Fig. 8 for a mesh with subdivision $N = 3$, is computed with 7 iterations for all mesh divisions. The convergence rate of the Newton method is depicted in Table 1 for a Cook's membrane problem with 32×32 elements.

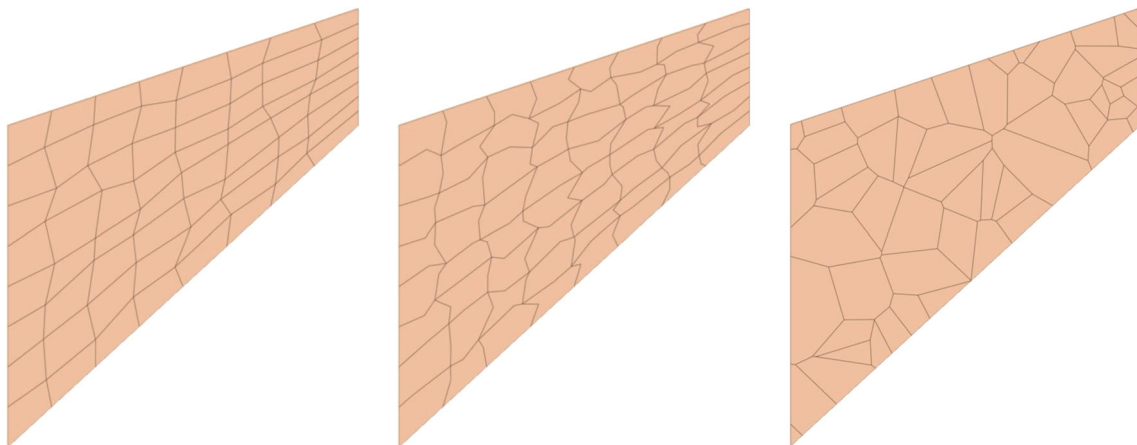


Fig. 7 Distorted mesh d1 and d2 and Voronoi mesh

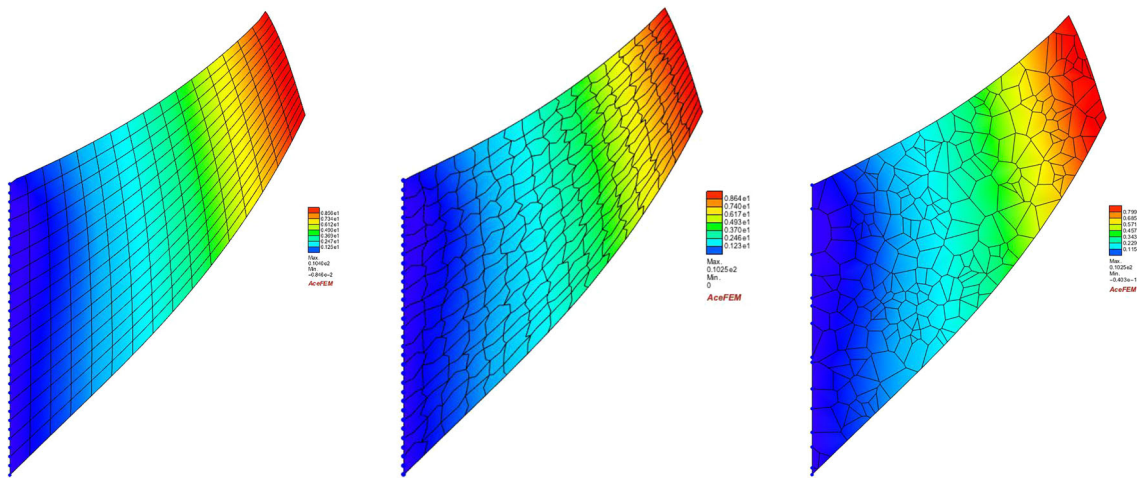


Fig. 8 Deformed configuration of the cantilever beam for a regular, a distorted and a Voronoi mesh

Table 1 Convergence rate, load applied in one load step

Iteration	$\ R(u)\ $
2	1.13×10^1
3	3.59×10^{-1}
4	$7, 21 \times 10^{-2}$
5	9.46×10^{-4}
6	3.84×10^{-7}
7	1.19×10^{-12}

Mesh convergence is investigated using the vertical displacement of the upper node, $U_y(48, 60)$ at the right end of the cantilever beam.

A mesh convergence study is performed for the new virtual element formulation, denoted here as VEM-T1. The Lamé parameters for the stabilization term in the strain energy function were computed using the formulation leading to (49) and (51), which resulted in $\hat{\lambda} = 25$ and $\hat{\mu} = 17$ for the regular mesh. The VEM-T1 element is compared with a standard biquadratic Q2 element that should have superior convergence rates.

To test the new VEM-T1 against a mesh that consists only of the internal triangles, here labeled T6, this element has been coded using the internal mesh shown in Fig. 6 and used to compute the Cook membrane problem as well. The results stemming from the T6 element are also presented in Fig. 9 that depicts the convergence of the vertical displacement at point (48, 60). Clearly the VEM-T1 element is superior to the T6 element. This is due to the fact that the stabilization term is derived from the equivalent bending energy, see (49). These results can also be observed in case of small strains.

It can be observed that all formulations converge from below in this case. Note that the displacement for the convergence of the VEM-T1 element lies between the Q2 and the T6 convergence and shows by far a superior conver-

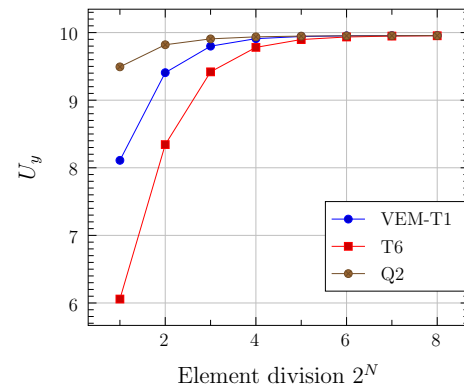


Fig. 9 Convergence study: VEM-T1, T6 and Q2 element

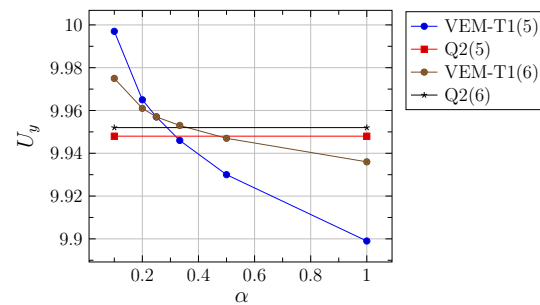


Fig. 10 Sensitivity study for a mesh with $N = 5$ and $N = 6$: VEM-T1, Q2 element

gence rate than the T6 formulation. From a mesh size with 16×16 ($N = 4$) on the VEM-T1 element performs almost as well as the Q2-element.

A sensitivity study with respect to the choice of the constitutive parameters of the stabilization energy (43) is performed next. This was conducted for 32×32 mesh ($N = 5$) and 64×64 ($N = 6$) meshes. The outcome is shown in Fig. 10 where the results of the VEM-T1 element are compared for

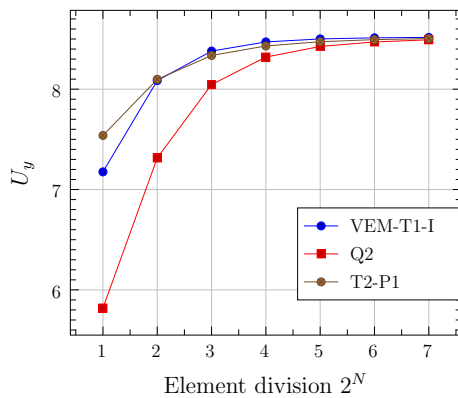


Fig. 11 Incompressible case, convergence study: VEM-T1-I-, Q2- and T2-P1-element

different values of $\alpha = \hat{\lambda} / \lambda$ with the solution obtained by the Q2 element. The value of the shear modulus $\hat{\mu}$ is given by $\hat{\mu} = \alpha \mu$ in this study. The differences are, in a range of the parameter $0, 2 \leq \alpha \leq 0.5$, very small with a maximum deviation of the solution from the Q2 element less than 0.2%. Even for the larger range $0, 1 \leq \alpha \leq 1$ the deviations are acceptable, being less than 1%. However, it should be noted that for $\alpha < 0.1$ the solution is no longer stable, showing light hour-glass modes that become distinct at $\alpha < 0.05$. Note that the parameters $\hat{\lambda}$ and $\hat{\mu}$ obtained following the suggestion of Krysl [19], see (49), lead in this case to the smallest deviation from the Q2 results and thus seem to be optimal.

In order to investigate whether the virtual element VEM-T1 also works for incompressible case the formulation discussed in Sect. 3.2.1 was applied. The constitutive parameters $\hat{\lambda}$ and $\hat{\mu}$ for the stabilization energy were kept as before. The results for different mesh densities were compared with the Q2-element and a Taylor-Hood triangular element with quadratic ansatz for the displacements and linear ansatz for the pressure (T2-P1) which fulfils the incompressibility constraint exactly. The results are depicted in Fig. 11.

Here is obvious that the virtual element VEM-T1-I can handle the incompressible case almost as well as the T2-P1 element. On the other hand it is clearly superior to the Q2 element which however, while higher order, is not designed for incompressible materials. It should be noted that the T2-P1 element yields the final result in one load step with 6 iterations independent of the mesh size. Also the VEM-T1-I element is able to compute the solution in one load step with 6 iterations, and thus has the same robustness as the T2-P1 element. It should be noted that the standard VEM-T1 formulation can also handle nearly incompressible behaviour and thus is locking free. However in comparison to the VEM-T1-P0, discussed in more detail in the next section, the formulation lacks robustness in the sense that more load steps are needed to obtain the final result. For the Cook's membrane problem this would mean that four load steps, using VEM-T1,

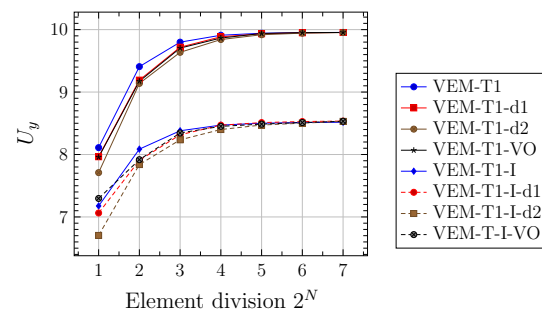


Fig. 12 Convergence study for distorted meshes

instead of one load step, using VEM-T1-I or VEM-T1-P0, are needed to obtain the results presented in Fig. 11.

Next, the virtual element formulation is applied to the Cook membrane problem with severely distorted meshes. Examples of the distorted meshes, which were randomly generated, are shown in Fig. 7 for $N = 3$.

The mesh (d1) on the left hand side comprises convex elements while the mesh d2 in the middle of the figure includes non-convex elements. Finally, on the right hand side a discretization is shown that is based on a Voronoi mesh.

Again a sequence of meshes from $N = 1, 2, 3, 4, 5, 6, 7$ is computed for the cases of the compressible and incompressible problem using the same constitutive parameters as selected above. Fig. 12 depicts the convergence for both cases. Here VEM-T1 denotes the compressible results while the element VEM-T1-I is related to the incompressible case. Even for the mesh (d2) with the non-convex shaped elements the convergence is smooth and a good coarse mesh accuracy can be observed in comparison with the regular grid, see Fig. 6. The Voronoi discretization leads to results that are very close to the mesh with the element distortion (d1).

In a final step the load was increased by a factor of 4. A Voronoi mesh was used for this application with 1017 elements. The load was applied in four steps, leading to a total number of 24 iterations to obtain the deformed state depicted in Fig. with a total computing time of less than 7 sec on a notebook with a two-core Intel processor of 3.1 GHz.

The final displacement at the point (40, 60) is $U_x = -13.23$ and $U_y = 26.94$. Figure 13 shows the distribution of the Cauchy stresses τ_{xx} and the shear stress τ_{xy} .

4.2 Punch problem

The next example is that of a punch that is driven into a solid, as shown in Fig. 14. It undergoes severe deformations and thus can be used to test the robustness of the virtual element formulation. We assume the same material properties as in the first example. One part of the upper plane is deformed by a prescribed vertical displacement. The displacements are set to zero along the bottom of the solid, the left and right

Fig. 13 Deformed state and Cauchy stresses τ_{xx} and τ_{xy} for a total load of $q_0 = 16$

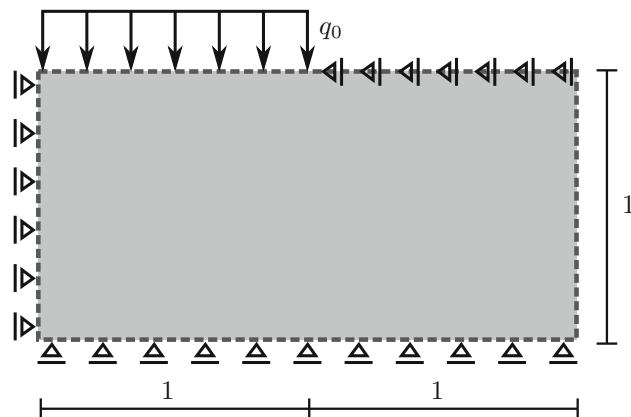
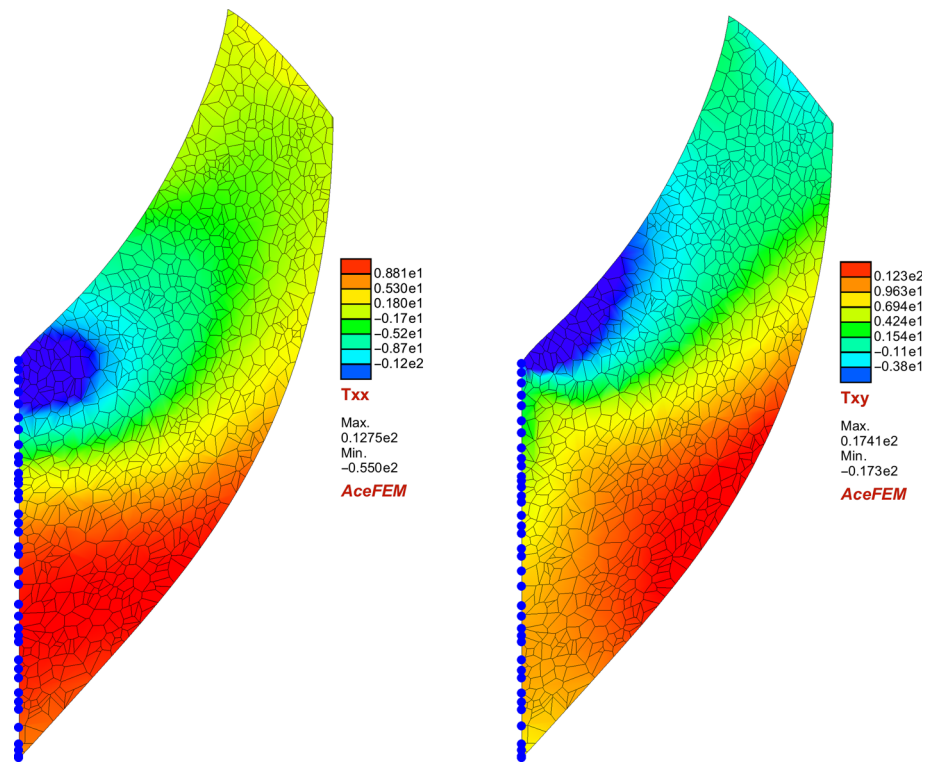


Fig. 14 Punch problem

sides of the solid are fixed in the x -direction, while the top is also fixed in the x -direction.

The deformation of the solid due to the applied vertical load $q_0 = 800$ is investigated for the compressible and incompressible case. The constitutive parameters for the compressible case are $\lambda = 138.75$ and $\mu = 92.5$. The parameters for the stabilization energy are computed using (51). The deformed mesh is depicted in Fig. 15 which shows the severe deformation related to the application of the full load (Figs. 16, 17).

For a convergence analysis the vertical displacement at the upper left point under the load is used. Again a mesh sequence is analysed, see Fig. 18. The convergence of the

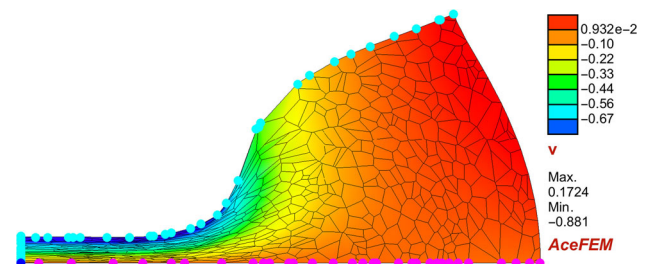


Fig. 15 Deformed configuration of the punch problem, compressible case for a mesh with $N = 4$

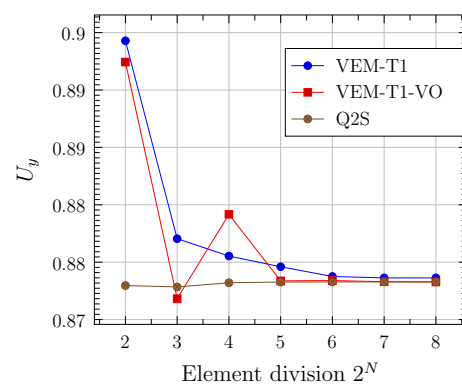


Fig. 16 Compressible deformation of the punch, convergence study: VEM-T1-, VEM-T1-VO, Q2S—element

new VEM-T1 element (normal and Voronoi mesh) is expectedly not as good as the convergence for the Q2 element for the compressible case but the result is close to the solution

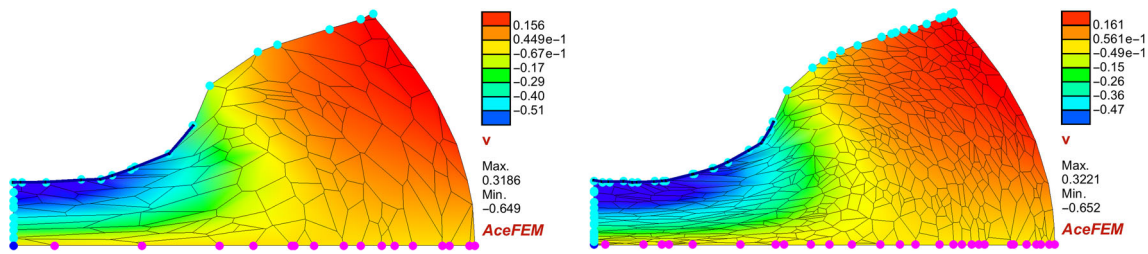


Fig. 17 Deformed configuration of the punch problem, incompressible case

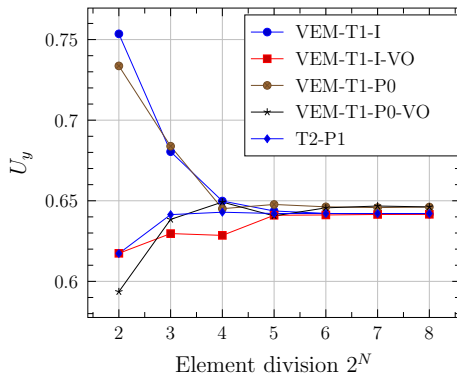


Fig. 18 Incompressible deformation of the punch, convergence study: VEM-T1-I, VEM-T1-P0 and T2-P1 - element

since for $N \geq 3$ the result deviates by less than 0.5 % from the converged solution. It should also be noted that the convergence related to the Voronoi mesh is not smooth since the meshes are generated randomly and thus a finer mesh does not have the same nodes as a coarser mesh.

In the incompressible case the Lamé parameter $\mu = 80$ was used for the incompressible VEM-T1-I element, see (29). The constitutive parameters for the stabilization energy were again computed from (51).

The final configuration for a load of $q_0 = 800$ is depicted in Fig. 17 for two Voronoi meshes with mesh densities related to $N = 3$ and $N = 5$. It is noted that finer meshes do not influence the overall convergence characteristics leading to the same number of load steps and iterations.

Again a convergence study is performed for the incompressible and nearly incompressible deformations of the punch. Here the incompressible element VEM-T1-I and the element related to the nearly incompressible formulation VEM-T1-P0, see (35), were employed. Two different meshes were selected for the VEM-T1-I and VEM-T1-P0 element. These are a regular mesh with rectangular 8 node virtual elements here denoted by VEM-T1-I and a Voronoi mesh denoted by VEM-T1-I-VO and VEM-T1-P0 and VEM-T1-P0-VO, respectively. The nearly incompressible solution is based on a choice of $\nu = 0.49999$. The results are compared to a solution obtained with a classical mixed T2-P1 element.

It can be seen that the convergence of the virtual element using the Voronoi mesh is not completely smooth which is

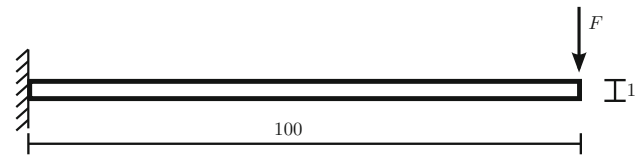


Fig. 19 Beam under a point load F

due to the random nature of the Voronoi mesh. However all results of the VEM-T1-I element converge to the solution that was generated with the T2-P1 element. Even for a coarse mesh of 16×8 elements ($N = 4$) the solution deviates less than 2% from the converged solution for the VEM-T1 with a regular mesh and about 2.5 % from the solution using the Voronoi mesh. Thus the incompressible version of the virtual element can handle incompressible deformations while fulfilling the constraint $J = 1$ at element level.

In the nearly incompressible case both meshes converge to a solution that deviates by less than 1 % from the solution of the incompressible case which is due to the fact that the incompressibility constraint is not exactly fulfilled. However the results of the VEM-P0 element converge as well as before. Also the number of load steps needed to reach the final deformation state are the same as for the VEM-T1-I element.

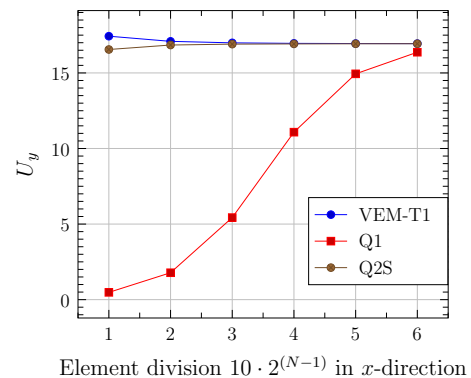


Fig. 20 Convergence study for the beam: VEM-T1-, Q1-, Q2S- element

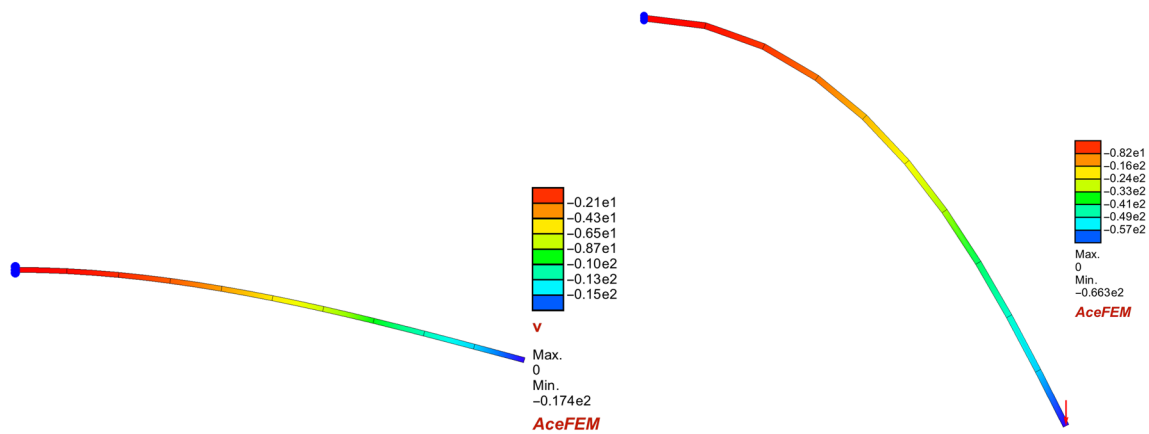


Fig. 21 Deflection of the beam for the coarse discretization

4.3 Thin beam

In a final example the behaviour of the new virtual element will be tested in bending dominated structural response. For this bending of a beam undergoing large deflections is considered. The beam is depicted in Fig. 19a. The beam is thin, having a height to length ratio of $H/L = 1/100$. It is loaded at the right end by a point load F and clamped at the left end. The constitutive data are given by $E = 10^7$ for the Young’s modulus and $\nu = 0.3$ for the Poisson ratio. This is equivalent to the Lamé constants $\lambda = 5769231$ and $\mu = 384615$. The applied load has the magnitude $F = 48$.

The analysis is performed with the VEM-T1 element using a mesh with rectangular shaped elements. Here the 8-node virtual element which has been already applied in Cook’s membrane problem is selected. The results are compared to the solution of a 4-node Q1 element and an 8-node Q2 serendipity element. The latter uses a quadratic interpolation for the displacement field and thus has superior convergence rates. All meshes are selected with elements that have a length to height ratio larger than 10 in order to test the applicability of the new virtual element formulation for more severe bending situations.

Figure 20 reports a mesh convergence study which is performed using a series of meshes with 10×1 , 20×2 , 40×4 , 80×6 , 160×8 and 320×10 elements.

The graphs in Fig. 20 show that the convergence of the linear 8-node VEM-T1 element is nearly as good as the convergence behaviour of the quadratic Q2S element despite that it only uses linear ansatz functions. Even for the coarse mesh with only 10 elements the VEM-T1 element deviates only by 3 % from the converged solution. The Q1 element depicts the classical bending locking. Only for a very fine mesh, with 1280×14 elements, it converges to the solution.

For the given load the total deflection is computed within one load step and 7 iterations in total, leading to the state depicted on the left side of Fig. 21.

On the right side side of Fig. 21 the deflection of the beam is shown for a load that is 10 times larger. In that case 3 load steps with total 22 iterations are needed to obtain the final state. The vertical displacement at the end deviates for a coarse mesh with $u_Y^{10 \times 1} = 66.3$ about 10 % from the converged solution of $u_Y^{conv} = 72.17$ while a solution with a 20×2 mesh is only 2 % away from the converged solution ($u_Y^{20 \times 2} = 70.66$).

5 Summary and conclusions

We have presented a virtual element method for nonlinear elasticity in which the key novel feature is a stabilization term involving a modified positive definite strain energy, together with a simple scheme for carrying out quadrature. The formulation is simple and has been presented for both compressible and incompressible materials. In the latter case a mixed formulation is used with the pressure as a Lagrange multiplier to enforce the incompressibility constraint. Locking-free behaviour is observed in the nearly incompressible and incompressible case and for bending dominated problems. The performance of the method is superior when compared with various other conventional or virtual element schemes.

The method proposed here is amenable to extensions of various kinds: for example to higher-order VEM formulations, to problems in three dimensions, and to other nonlinear problems such as those involving inelastic material behaviour. First results indicate that the same stabilization procedure can be employed in the three-dimensional case however these topics await investigation.

Acknowledgements The first author gratefully acknowledges support by the Deutsche Forschungsgemeinschaft in the Priority Program 1748 ‘Reliable simulation techniques in solid mechanics: Development of non- standard discretization methods, mechanical and mathematical analysis’ under the Project WR 19/50-1. The second author acknowledges the support of the National Research Foundation of South Africa,

and the Alexander von Humboldt Foundation through a Georg Forster Research Fellowship.

References

- Bathe KJ (1996) Finite element procedures. Prentice-Hall, Englewood Cliffs
- Belytschko T, Bindeman LP (1991) Assumed strain stabilization of the 4-node quadrilateral with 1-point quadrature for nonlinear problems. *Comput Methods Appl Mech Eng* 88(3):311–340
- Belytschko T, Liu WK, Moran B (2000) Nonlinear finite elements for continua and structures. Wiley, Chichester
- Belytschko T, Ong JSJ, Liu WK, Kennedy JM (1984) Hourglass control in linear and nonlinear problems. *Comput Methods Appl Mech Eng* 43:251–276
- Biabanaki S, Khoei A (2012) A polygonal finite element method for modeling arbitrary interfaces in large deformation problems. *Comput Mech* 50:19–33
- Biabanaki SOR, Khoei AR, Wriggers P (2014) Polygonal finite element methods for contact-impact problems on non-conformal meshes. *Comput Methods Appl Mech Eng* 269:198–221
- Boerner E, Loehnert S, Wriggers P (2007) A new finite element based on the theory of a Cosserat point—extension to initially distorted elements for 2D plane strains. *Int J Numer Methods Eng* 71:454–472
- Cangiani A, Manzini G, Russo A, Sukumar N (2015) Hourglass stabilization and the virtual element method. *Int J Numer Methods Eng* 102:404–436
- Chi H, da Veiga LB, Paulino G (2016) Some basic formulations of the virtual element method (VEM) for finite deformations. *Comput Methods Appl Mech Eng*. doi:10.1016/j.cma.2016.12.020
- Chi H, Talischi C, Lopez-Pamies O, Paulino HG (2015) Polygonal finite elements for finite elasticity. *Int J Numer Methods Eng* 101(4):305–328
- da Beirão LV, Brezzi F, Cangiani A, Manzini G, Marini L, Russo A (2013) Basic principles of virtual element methods. *Math Models Methods Appl Sci* 23(01):199–214
- da Beirão LV, Brezzi F, Marini L (2013) Virtual elements for linear elasticity problems. *SIAM J Numer Anal* 51(2):794–812
- da Beirão LV, Brezzi F, Marini LD, Russo A (2014) The Hitchhiker's guide to the virtual element method. *Math Models Methods Appl Sci* 24(08):1541–1573
- da Beirão LV, Lovadina C, Mora D (2015) A virtual element method for elastic and inelastic problems on polytope meshes. *Comput Methods Appl Mech Eng* 295:327–346
- Flanagan D, Belytschko T (1981) A uniform strain hexahedron and quadrilateral with orthogonal hour-glass control. *Int J Numer Methods Eng* 17:679–706
- Gain AL, Talischi C, Paulino GH (2014) On the virtual element method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes. *Comput Methods Appl Mech Eng* 282:132–160
- Korelc J, Solinc U, Wriggers P (2010) An improved EAS brick element for finite deformation. *Comput Mech* 46:641–659
- Korelc J, Wriggers P (2016) Automation of finite element methods. Springer, Berlin
- Krysl P (2015a) Mean-strain eight-node hexahedron with optimized energy-sampling stabilization for large-strain deformation. *Int J Numer Methods Eng* 103:650–670
- Krysl P (2015b) Mean-strain eight-node hexahedron with stabilization by energy sampling stabilization. *Int J Numer Methods Eng* 103:437–449
- Krysl P (2016) Mean-strain 8-node hexahedron with optimized energy-sampling stabilization. *Finite Elem Anal Des* 108:41–53
- Loehnert S, Boerner E, Rubin M, Wriggers P (2005) Response of a nonlinear elastic general Cosserat brick element in simulations typically exhibiting locking and hourglassing. *Comput Mech* 36:255–265
- Lovadina C, Auricchio F (2003) On the enhanced strain technique for elasticity problems. *Comput Struct* 81:777–787
- Mueller-Hoeppe DS, Loehnert S, Wriggers P (2009) A finite deformation brick element with inhomogeneous mode enhancement. *Int J Numer Methods Eng* 78:1164–1187
- Nadler B, Rubin M (2003) A new 3-d finite element for nonlinear elasticity using the theory of a cosserat point. *Int J Solids Struct* 40:4585–4614
- Noels L, Radovitzky R (2006) A general discontinuous galerkin method for finite hyperelasticity. formulation and numerical applications. *Int J Numer Methods Eng* 68:64–97
- Reddy BD, Simo JC (1995) Stability and convergence of a class of enhanced strain methods. *SIAM J Numer Anal* 32:1705–1728
- Reese S (2003) On a consistent hourglass stabilization technique to treat large inelastic deformations and thermo-mechanical coupling in plane strain problems. *Int J Numer Methods Eng* 57:1095–1127
- Reese S, Kuessner M, Reddy BD (1999) A new stabilization technique to avoid hourglassing in finite elasticity. *Int J Numer Methods Eng* 44:1617–1652
- Reese S, Wriggers P (2000) A new stabilization concept for finite elements in large deformation problems. *Int J Numer Methods Eng* 48:79–110
- Simo JC, Armero F (1992) Geometrically non-linear enhanced strain mixed methods and the method of incompatible modes. *Int J Numer Methods Eng* 33:1413–1449
- Simo JC, Rifai MS (1990) A class of assumed strain methods and the method of incompatible modes. *Int J Numer Methods Eng* 29:1595–1638
- Simo JC, Taylor RL, Pister KS (1985) Variational and projection methods for the volume constraint in finite deformation elastoplasticity. *Comput Methods Appl Mech Eng* 51:177–208
- Sukumar N (2004) Construction of polygonal interpolants: a maximum entropy approach. *Int J Numer Methods Eng* 61:2159–2181
- Sukumar N, Malsch EA (2006) Recent advances in the construction of polygonal finite element interpolants. *Arch Comput Methods Eng* 13:129–163
- Taylor RL (2000) A mixed-enhanced formulation for tetrahedral finite elements. *Int J Numer Methods Eng* 47:205–227
- ten Eyck A, Lew A (2006) Discontinuous Galerkin methods for non-linear elasticity. *Int J Numer Methods Eng* 67:1204–1243
- Wriggers P (2008) Nonlinear finite elements. Springer, Berlin, Heidelberg, New York
- Wriggers P, Rust W, Reddy BD (2016) A virtual element method for contact. *Comput Mech* 58:1039–1050